

MELSEC

Serie A/Q

Manuale di programmazione

**Manuale di programmazione per
MELSEC serie A e Q e MELSEC System Q
Numero catalogo: 212576**

Revisione			Modifiche / Aggiunte / Correzioni
A	09/1998	pdp	Manuale di programmazione per le serie MELSEC A e Q sulla base di Melsec Medoc <i>plus</i>
B	04/2001	pdp	<p>Cap. 5.1.2: Modificata bitmap (schema a contatti in GX IEC Developer) per l'istruzione ORP Modificata bitmap per istruzione LDP</p> <p>Cap. 7.6.3: Nota per l'uso dell'istruzione CALL</p> <p>Cap. 7.14: Aggiunta delle istruzioni RSET_K_MD e RSET_K_P_MD</p> <p>Cap. 7.11.13: Nota per l'uso dell'istruzione ASC(P)</p> <p>Ulteriori informazioni per le CPU del System Q (Q02, Q02H, Q06H, Q12H e Q25H).</p>
C	08/2002	pdp-dk	<p>Ulteriori informazioni per Q00JCPU, Q00CPU e Q01CPU del System Q.Q25H)</p> <p>Nuove istruzioni S.TO e FROM per uso in sistemi multi-CPU.</p> <p>Nel capitolo 9 compare ora il formato di rappresentazione dell'istruzione in GX IEC Developer.</p> <p>Relé e registri speciali aggiuntivi per le CPU del System Q con versione funzionale B o successiva.</p> <p>Tabelle separate per i tempi di elaborazione della serie A e serie Q/System Q.</p> <p>Aggiunta di un esempio nel cap. 7.6.10, che mostra la modifica di programma.</p> <p>Correzioni:</p> <p>Cap. 6.5.1: Errori di esecuzione</p> <p>Cap. 6.5.2: Errori di esecuzione</p> <p>Cap. 6.7.3: Informazioni aggiuntive per l'istruzione COM se usata in un sistema multi-CPU</p> <p>Cap. 6.8.9: Valore tempi per n1</p> <p>Cap. 7.1.1: Operandi MELSEC Q</p> <p>Cap. 7.1.3: Operandi MELSEC Q</p> <p>Cap. 7.1.5: Operandi MELSEC Q</p> <p>Cap. 7.1.7: Operandi MELSEC Q</p> <p>Cap. 7.5.12: Errori di esecuzione</p> <p>Cap. 9.5.1: Tempi di elaborazione delle istruzioni RBMOV e BMOV</p>
D	09/2004	pdp-dk	<p>Nuovo capitolo 10: Istruzioni per Q4ARCPU</p> <p>Nuovo capitolo 11: Istruzioni dedicate per moduli funzione speciali</p> <p>Cap. 2.8: Riassunto delle istruzioni per Q4ARCPU</p> <p>Cap. 2.9: Riassunto delle istruzioni dedicate</p> <p>Nuovi moduli CPU Q12PHCPU e Q25PHCPU</p>

Note sul manuale

I testi, illustrazioni, schemi ed esempi contenuti in questo manuale vanno intesi esclusivamente come materiale di supporto per la spiegazione, manipolazione, programmazione e funzionamento dei controllori logici programmabili MELSEC serie A e Q e del System Q MELSEC.

In caso di domande relative alla programmazione ed al funzionamento della apparecchiature descritte in questo manuale, si prega di contattare l'agenzia o il reparto di vendita più vicino (vedi ultima di copertina).

Informazioni aggiornate e risposte alle domande più frequenti sono anche disponibili su Internet (www.mitsubishi-automation.com)

MITSUBISHI ELECTRIC EUROPE B.V. si riserva il diritto di apportare modifiche al presente manuale senza preavviso.

© 09/2004

Contenuto

1	Introduzione	
1.1	Ulteriori manuali	1-1
1.2	Tipi di CPU	1-2
1.3	Software	1-2
1.4	Ricerca di una istruzione	1-3
1.5	Parametri PLC	1-3
1.6	Confronto fra GX IEC Developer e GX Developer	1-4
2	Tabelle istruzioni	
2.1	Suddivisione delle istruzioni	2-1
2.2	Panoramica delle istruzioni	2-5
2.2.1	Descrizione delle tabelle di panoramica	2-5
2.3	Istruzioni di sequenza	2-7
2.3.1	Istruzioni di ingresso	2-7
2.3.2	Istruzioni di collegamento	2-8
2.3.3	Istruzioni di uscita	2-9
2.3.4	Istruzioni di scorrimento	2-9
2.3.5	Istruzioni di controllo primario	2-10
2.3.6	Istruzioni di terminazione programma	2-10
2.3.7	Istruzioni miscellanee	2-10
2.4	Istruzioni applicative, parte 1	2-11
2.4.1	Istruzioni per operazioni di confronto	2-11
2.4.2	Istruzioni per operazioni aritmetiche	2-16
2.4.3	Istruzioni di conversione dati	2-23
2.4.4	Istruzioni di trasferimento dati	2-26
2.4.5	Istruzioni di salto di programma	2-28
2.4.6	Istruzioni di controllo per interruzione esecuzione programma	2-28
2.4.7	Istruzioni per rinfresco dati	2-29
2.4.8	Altre istruzioni utili	2-30
2.5	Istruzioni applicative, parte 2	2-32
2.5.1	Istruzioni per operazioni logiche	2-32
2.5.2	Istruzioni di rotazione	2-36

2.5.3	Istruzioni di scorrimento	2-37
2.5.4	Istruzioni di elaborazione bit	2-38
2.5.5	Istruzioni per elaborazione dati	2-39
2.5.6	Istruzioni per programmi strutturati	2-42
2.5.7	Istruzioni per operazioni su tabelle dati	2-44
2.5.8	Istruzioni di accesso a buffer di memoria	2-45
2.5.9	Istruzioni di visualizzazione	2-46
2.5.10	Istruzioni per debug e diagnosi guasti	2-47
2.5.11	Istruzioni di elaborazione stringhe di caratteri	2-48
2.5.12	Istruzioni per funzioni speciali	2-52
2.5.13	Istruzioni per controllo dati	2-55
2.5.14	Istruzioni di commutazione registro file	2-56
2.5.15	Istruzioni per orologio	2-57
2.5.16	Istruzioni per dispositivi periferici	2-58
2.5.17	Istruzioni di programma	2-58
2.5.18	Altre istruzioni	2-59
2.6	Istruzioni per collegamento dati	2-61
2.6.1	Istruzioni rinfresco rete	2-61
2.6.2	Istruzioni dedicate per collegamento dati	2-61
2.6.3	Istruzioni per collegamento dati compatibili con serie A	2-62
2.6.4	Letture/scrittura dati di instradamento	2-62
2.7	Istruzioni per CPU System Q	2-63
2.7.1	Letture informazioni dei moduli	2-63
2.7.2	Istruzioni per debug e diagnosi guasti	2-63
2.7.3	Scrittura e lettura da file	2-64
2.7.4	Istruzioni di programma	2-64
2.7.5	Istruzioni per trasferimento dati	2-65
2.7.6	Istruzioni per scambio dati in un sistema multi-CPU	2-65
2.8	Istruzioni dedicate per Q4ARCPU	2-66
2.8.1	Istruzioni per impostazione del modo	2-66
2.8.2	Istruzioni per trasferimento dati	2-66
2.9	Istruzioni per moduli funzione speciali	2-67
2.9.1	Istruzioni per moduli di comunicazione seriale	2-67
2.9.2	Istruzioni per modulo interfaccia PROFIBUS/DP	2-68
2.9.3	Istruzioni per moduli interfaccia ETHERNET	2-69
2.9.4	Istruzioni per MELSECNET/10	2-69
2.9.5	Istruzioni per CC-Link	2-70

3	Configurazione delle istruzioni	
3.1	La struttura di una istruzione	3-1
3.1.1	Sorgente dei dati (s)	3-1
3.1.2	Destinazione dei dati (d)	3-2
3.1.3	Numero (n)	3-2
3.2	Notazione delle istruzioni	3-3
3.2.1	16/ 32-bit e impulso	3-3
3.2.2	MELSEC e IEC	3-3
3.2.3	Ulteriori caratteristiche della notazione istruzioni	3-5
3.2.4	Specifiche della notazione	3-5
3.3	Programmazione di istruzioni dedicate	3-6
3.4	Programmazione di variabili	3-7
3.4.1	Programmazione con GX IEC Developer	3-7
3.4.2	Programmazione con GX Developer	3-8
3.5	Tipi di dati	3-9
3.5.1	Elaborazione dati	3-11
3.5.2	Indirizzamento di array e registri con GX IEC Developer	3-19
3.5.3	Uso di dati a stringhe di caratteri (STRING)	3-22
3.6	Indicizzazione	3-24
3.6.1	Indicizzazione	3-24
3.6.2	Caratteristiche speciali delle CPU System Q e QnA	3-26
3.6.3	Caratteristiche speciali delle CPU AnA, AnAS e AnU	3-28
3.7	Indirizzamento indiretto (solo GX Developer)	3-29
3.8	Errore di esecuzione	3-31
3.8.1	Verifica del campo operando	3-31
3.8.2	Verifica del dato operando	3-33
3.9	Condizioni di esecuzione delle istruzioni	3-34
3.9.1	Condizione di esecuzione	3-34
3.9.2	Ingresso EN e uscita ENO	3-35
3.10	Numero passi di programma	3-37
3.10.1	Per una CPU System Q e QnA	3-37
3.10.2	Per una CPU AnA, AnAS e AnU	3-38

4	Organizzazione e struttura dei capitoli	
4.1	Panoramica delle istruzioni	4-2
4.2	La tabella CPU	4-2
4.3	Operandi MELSEC A	4-3
4.4	Operandi MELSEC Q	4-4
4.5	Rappresentazione del formato di una istruzione	4-4
4.5.1	Rappresentazione in GX IEC Developer	4-4
4.5.2	Rappresentazione in GX Developer	4-5
4.6	Variabili	4-5
4.7	Funzioni	4-6
4.8	Note	4-6
4.9	Errori di esecuzione	4-6
4.10	Esempi di programma	4-7
5	Istruzioni di sequenza	
5.1	Istruzioni di ingresso	5-4
5.1.1	LD, LDI, AND, ANI, OR, ORI	5-4
5.1.2	LDP, LDF, ANDP, ANDF, ORP, ORF	5-8
5.2	Istruzioni di collegamento	5-11
5.2.1	ANB, ORB	5-11
5.2.2	MPS, MRD, MPP	5-14
5.2.3	INV	5-17
5.2.4	MEP, MEF	5-19
5.2.5	EGP, EGF	5-21

5.3	Istruzioni di uscita	5-23
5.3.1	OUT	5-23
5.3.2	OUT T, OUTH T	5-25
5.3.3	OUT C	5-29
5.3.4	OUT F	5-32
5.3.5	SET	5-35
5.3.6	RST	5-37
5.3.7	SET F, RST F	5-40
5.3.8	PLS, PLF	5-43
5.3.9	FF	5-47
5.3.10	CHK	5-49
5.3.11	DELTA, DELTAP	5-51
5.4	Istruzioni di scorrimento	5-53
5.4.1	SFT, SFTP	5-53
5.5	Istruzioni di controllo primario	5-56
5.5.1	MC, MCR	5-56
5.6	Istruzioni di terminazione	5-62
5.6.1	FEND	5-62
5.6.2	END	5-65
5.7	Istruzioni miscellanee	5-68
5.7.1	STOP	5-68
5.7.2	NOP	5-71

6 Istruzioni applicative, parte 1

6.1	Istruzioni per operazioni di confronto	6-2
6.1.1	=, <>, >, <=, <, >=	6-5
6.1.2	D=, D<>, D>, D<=, D<, D>=	6-8
6.1.3	E=, E<>, E>, E<=, E<, E>=	6-11
6.1.4	\$ =, \$ <>, \$ >, \$ <=, \$ <, \$ >=	6-15
6.1.5	BKCMP, BKCMPP	6-20
6.2	Istruzioni per operazioni aritmetiche	6-25
6.2.1	+, +P, -, -P	6-28
6.2.2	D+, D+P, D-, D-P	6-32
6.2.3	x, xP, /, /P	6-36
6.2.4	Dx, DxP, D/, D/P	6-40
6.2.5	B+, B+P, B-, B-P	6-43
6.2.6	DB+, DB+P, DB-, DB-P	6-48
6.2.7	Bx, BxP, B/, B/P	6-53
6.2.8	DBx, DBxP, DB/, DB/P	6-57
6.2.9	E+, E+P, E-, E-P	6-61
6.2.10	Ex, ExP, E/, E/P	6-66
6.2.11	BK+, BK+P, BK-, BK-P	6-69
6.2.12	\$+, \$+P	6-73
6.2.13	INC, INCP, DEC, DECP	6-77
6.2.14	DINC, DINCP, DDEC, DDECP	6-80
6.3	Istruzioni di conversione dati	6-81
6.3.1	BCD, BCDP, DBCD, DBCDP	6-82
6.3.2	BIN, BINP, DBIN, DBINP	6-85
6.3.3	FLT, FLTP, DFLT, DFLTP	6-89
6.3.4	INT, INTP, DINT, DINTP	6-92
6.3.5	DBL, DBLP	6-95
6.3.6	WORD, WORDP	6-97
6.3.7	GRY, GRYP, DGRY, DGRYP	6-99
6.3.8	GBIN, GBINP, DGBIN, DGBINP	6-102
6.3.9	NEG, NEGP, DNEG, DNEGP	6-105
6.3.10	ENEG, ENEGP	6-108
6.3.11	BKBCD, BKBCDP	6-110
6.3.12	BKBIN, BKBINP	6-113

6.4	Istruzioni per trasferimento dati	6-116
6.4.1	MOV, MOVP, DMOV, DMOVP	6-117
6.4.2	EMOV, EMOVP	6-120
6.4.3	\$MOV, \$MOVP	6-123
6.4.4	CML, CMLP, DCML, DCMLP	6-126
6.4.5	BMOV, BMOVP	6-131
6.4.6	FMOV, FMOVP	6-134
6.4.7	XCH, XCHP, DXCH, DXCHP	6-137
6.4.8	BXCH, BXCHP	6-140
6.4.9	SWAP, SWAPP	6-143
6.5	Istruzioni di salto di programma	6-147
6.5.1	CJ, SCJ, JMP	6-148
6.5.2	GOEND	6-153
6.6	Istruzioni per controllo esecuzione del programma	6-155
6.6.1	DI, EI, IMASK	6-156
6.6.2	IRET	6-163
6.7	Istruzioni per rinfresco collegamento	6-165
6.7.1	RFS, RFSP	6-166
6.7.2	SEG	6-168
6.7.3	COM	6-172
6.7.4	EI, DI	6-175
6.8	Altre istruzioni utili	6-178
6.8.1	UDCNT1	6-179
6.8.2	UDCNT2	6-182
6.8.3	TTMR	6-185
6.8.4	STM, STMH	6-187
6.8.5	ROTC	6-191
6.8.6	RAMP	6-195
6.8.7	SPD	6-197
6.8.8	PLSY	6-199
6.8.9	PWM	6-201
6.8.10	MTR	6-203

7	Istruzioni applicative, parte 2	
7.1	Istruzioni per operazioni logiche	7-2
7.1.1	WAND, WANDP, DAND, DANDP	7-4
7.1.2	BKAND, BKANDP	7-11
7.1.3	WOR, WORP, DOR, DORP	7-14
7.1.4	BKOR, BKORP	7-20
7.1.5	WXOR, WXORP, DXOR, DXORP	7-23
7.1.6	BKXOR, BKXORP	7-29
7.1.7	WXNR, WXNRP, DXNR, DXNRP	7-32
7.1.8	BKXNR, BKXNRP	7-39
7.2	Istruzioni per rotazione dati	7-42
7.2.1	ROR, RORP, RCR, RCRP	7-43
7.2.2	ROL, ROLP, RCL, RCLP	7-46
7.2.3	DROR, DRORP, DROR, DRORP	7-49
7.2.4	DROR, DRORP, DROR, DRORP	7-52
7.3	Istruzioni di scorrimento	7-55
7.3.1	SFR, SFRP, SFL, SFLP	7-56
7.3.2	BSFR, BSFRP, BSFL, BSFLP	7-59
7.3.3	DSFR, DSFRP, DSFL, DSFLP	7-62
7.4	Istruzioni di elaborazione bit	7-65
7.4.1	BSET, BSETP, BRST, BRSTP	7-66
7.4.2	TEST, TESTP, DTEST, DTESTP	7-69
7.4.3	BKRST, BKRSTP	7-73
7.5	Istruzioni per elaborazione dati	7-76
7.5.1	SER, SERP, DSER, DSERP	7-78
7.5.2	SUM, SUMP, DSUM, DSUMP	7-84
7.5.3	DECO, DECOP	7-87
7.5.4	ENCO, ENCOP	7-89
7.5.5	SEG, SEGP	7-91
7.5.6	DIS, DISP	7-95
7.5.7	UNI, UNIP	7-98
7.5.8	NDIS, NDISP, NUNI, NUNIP	7-101
7.5.9	WTOB, WTOBP, BTOW, BTOWP	7-106
7.5.10	MAX, MAXP, DMAX, DMAXP	7-111
7.5.11	MIN, MINP, DMIN, DMINP	7-114
7.5.12	SORT, SORTP, DSORT, DSORTP	7-117
7.5.13	WSUM, WSUMP	7-121
7.5.14	DWSUM, DWSUMP	7-123

7.6	Istruzioni per programmi strutturati	7-125
7.6.1	FOR, NEXT	7-126
7.6.2	BREAK, BREAKP	7-129
7.6.3	CALL, CALLP	7-132
7.6.4	RET	7-135
7.6.5	FCALL, FCALLP	7-137
7.6.6	ECALL, ECALLP	7-141
7.6.7	EFCALL, EFCALLP	7-144
7.6.8	CHG	7-147
7.6.9	SUB, SUBP	7-156
7.6.10	IX, IXEND	7-159
7.6.11	IXDEV, IXSET	7-164
7.7	Istruzioni per operazioni su tabelle dati	7-167
7.7.1	FIFW, FIFWP	7-168
7.7.2	FIFR, FIFRP	7-172
7.7.3	FPOP, FPOPP	7-176
7.7.4	FDEL, FDELP, FINS, FINSP	7-180
7.8	Istruzioni di accesso a buffer di memoria	7-185
7.8.1	FROM, DFRO	7-186
7.8.2	TO, DTO, DTO, DTO	7-190
7.9	Istruzioni di visualizzazione	7-194
7.9.1	PR	7-196
7.9.2	PRC	7-201
7.9.3	LED	7-205
7.9.4	LEDC	7-208
7.9.5	LEDA, LEDB	7-211
7.9.6	LEDR	7-213
7.10	Diagnostica guasti e debug	7-217
7.10.1	CHKST, CHK (Solo serie Q e System Q):	7-218
7.10.2	CHK (solo serie A)	7-226
7.10.3	CHKCIR, CHKEND	7-234
7.10.4	SLT, SLTR	7-239
7.10.5	STRA, STRAR	7-241
7.10.6	PTRA, PTRAR, PTRAEXE, PTRAEXEP	7-243
7.11	Istruzioni di elaborazione stringhe di caratteri	7-245
7.11.1	BINDA, BINDAP, DBINDA, DBINDAP	7-248
7.11.2	BINHA, BINHAP, DBINHA, DBINHAP	7-253

7.11.3	BCDDA, BCDDAP, DBCDDA, DBCDDAP	7-258
7.11.4	DABIN, DABINP, DDABIN, DDABINP	7-263
7.11.5	HABIN, HABINP, DHABIN, DHABINP	7-268
7.11.6	DABCD, DABCDP, DDABCD, DDABCDP	7-272
7.11.7	COMRD, COMRDP	7-277
7.11.8	LEN, LENP	7-281
7.11.9	STR, STRP, DSTR, DSTRP	7-284
7.11.10	VAL, VALP, DVAL, DVALP	7-292
7.11.11	ESTR, ESTRP	7-298
7.11.12	EVAL, EVALP	7-307
7.11.13	ASC, ASCP (serie Q e System Q)	7-313
7.11.14	ASC (serie A)	7-316
7.11.15	HEX, HEXP	7-318
7.11.16	RIGHT, RIGHTP, LEFT, LEFTP	7-322
7.11.17	MIDR, MIDRP, MIDW, MIDWP	7-326
7.11.18	INSTR, INSTRP	7-332
7.11.19	EMOD, EMODP	7-336
7.11.20	EREXP, EREXPP	7-339
7.12	Funzioni speciali	7-342
7.12.1	SIN, SINP	7-344
7.12.2	COS, COSP	7-347
7.12.3	TAN, TANP	7-350
7.12.4	ASIN, ASINP	7-353
7.12.5	ACOS, ACOSP	7-356
7.12.6	ATAN, ATANP	7-359
7.12.7	RAD, RADP	7-362
7.12.8	DEG, DEGP	7-365
7.12.9	SQR, SQRP	7-368
7.12.10	EXP, EXPP	7-371
7.12.11	LOG, LOGP	7-374
7.12.12	RND, RNDP, SRND, SRNDP	7-377
7.12.13	BSQR, BSQRP, BDSQR, BDSQRP	7-379
7.12.14	BSIN, BSINP	7-383
7.12.15	BCOS, BCOSP	7-386
7.12.16	BTAN, BTANP	7-389
7.12.17	BASIN, BASINP	7-393
7.12.18	BACOS, BACOSP	7-396
7.12.19	BATAN, BATANP	7-399

7.13	Istruzioni per controllo dati	7-401
7.13.1	LIMIT, LIMITP, DLIMIT, DLIMITP	7-402
7.13.2	BAND, BANDP, DBAND, DBANDP	7-406
7.13.3	ZONE, ZONEP, DZONE, DZONEP	7-410
7.14	Istruzioni di commutazione file registri	7-414
7.14.1	RSET, RSETP	7-415
7.14.2	QDRSET, QDRSETP	7-418
7.14.3	QCDSET, QCDSETP	7-421
7.15	Istruzioni per orologio	7-424
7.15.1	DATERD, DATERDP	7-425
7.15.2	DATEWR, DATEWRP	7-429
7.15.3	DATE+, DATE+P	7-433
7.15.4	DATE-, DATE-P	7-438
7.15.5	SECOND, SECONDP, HOUR, HOURP	7-444
7.16	Istruzioni per dispositivi periferici	7-450
7.16.1	MSG	7-451
7.16.2	PKEY	7-454
7.17	Istruzioni controllo di programma	7-457
7.17.1	PSTOP, PSTOPP	7-458
7.17.2	POFF, POFFP	7-460
7.17.3	PSCAN, PSCANP	7-462
7.17.4	PLOW, PLOWP	7-464
7.18	Altre istruzioni utili	7-466
7.18.1	WDT, WDTP	7-467
7.18.2	STC, CLC	7-469
7.18.3	DUTY	7-471
7.18.4	ZRRDB, ZRRDBP	7-474
7.18.5	ZRWRB, ZRWRBP	7-478
7.18.6	ADRESET, ADRSETP	7-482
7.18.7	KEY	7-483
7.18.8	ZPUSH, ZPUSHP, ZPOP, ZPOPP	7-489
7.18.9	EROMWR, EROMWRP	7-492

8	Istruzioni per comunicazione dati	
8.1	Fondamenti	8-1
8.2	Categorie di istruzioni	8-1
8.3	Campi di lettura e scrittura dei dati	8-3
8.3.1	MELSECNET/10	8-3
8.3.2	MELSECNET	8-4
8.4	Istruzioni dedicate per collegamenti dati	8-4
8.4.1	Esecuzione simultanea	8-4
8.4.2	Completamento della trasmissione	8-4
8.5	Istruzioni per rinfresco dati	8-6
8.5.1	ZCOM	8-7
8.6	Istruzioni di comunicazione dedicate per la serie QnA	8-11
8.6.1	READ	8-12
8.6.2	SREAD	8-18
8.6.3	WRITE	8-25
8.6.4	SWRITE	8-32
8.6.5	SEND	8-39
8.6.6	RECV	8-47
8.6.7	REQ	8-52
8.6.8	ZNFR	8-63
8.6.9	ZNTO	8-69
8.7	Istruzioni di comunicazione dati compatibili con serie A	8-76
8.7.1	ZNRD	8-77
8.7.2	ZNWR	8-81
8.7.3	LRDP	8-85
8.7.4	LWTP	8-89
8.7.5	RFRP	8-93
8.7.6	RTOP	8-99
8.8	Lettura e scrittura di informazioni di instradamento	8-105
8.8.1	RTREAD	8-106
8.8.2	RTWRITE	8-108

9	Istruzioni per CPU System Q	
9.1	Letture informazioni modulo	9-2
9.1.1	UNIRD, UNIRD	9-2
9.2	Istruzioni per debug e diagnosi guasti	9-7
9.2.1	TRACE, TRACER	9-7
9.3	Scrittura e lettura da file	9-9
9.3.1	FWRITE	9-9
9.3.2	FREAD	9-20
9.4	Istruzioni di programma	9-33
9.4.1	PLOADP	9-33
9.4.2	PUNLOADP	9-36
9.4.3	PSWAPP	9-38
9.5	Istruzioni per trasferimento dati	9-41
9.5.1	RBMOV, RBMOV	9-41
9.6	Istruzioni per uso in un sistema multi CPU	9-46
9.6.1	S.TO, SP.TO	9-46
9.6.2	FROM, FROMP	9-49
10	Istruzioni per Q4ARCPU	
10.1	Istruzioni di impostazione di modo	10-2
10.1.1	STMODE	10-2
10.1.2	CGMODE	10-4
10.2	Istruzioni per trasferimento dati	10-6
10.2.1	TRUCK	10-6
10.2.2	SPREF	10-11

11 Istruzioni per moduli funzione speciali

11.1	Istruzioni per moduli di comunicazione seriale	11-2
11.1.1	BUFRCVS	11-3
11.1.2	GETE, GETEP	11-6
11.1.3	PUTE, PUTEP	11-11
11.1.4	PRR, PRRP	11-18
11.2	Istruzioni per modulo interfaccia PROFIBUS/DP	11-26
11.2.1	BBLKRD, BBLKRDP	11-27
11.2.2	BBLKWR, BBLKWRP	11-30
11.3	Istruzioni per moduli interfaccia ETHERNET	11-33
11.3.1	BUFRCV	11-34
11.3.2	BUFRCVS	11-39
11.3.3	BUFSND	11-42
11.3.4	OPEN	11-47
11.3.5	CLOSE	11-56
11.3.6	ERRCLR	11-61
11.3.7	ERRRD	11-67
11.3.8	UINI	11-72
11.4	Istruzioni per MELSECNET/10	11-78
11.4.1	PAIRSET	11-79
11.5	Istruzioni per CC-Link	11-82
11.5.1	RLPA (serie A)	11-83
11.5.2	RLPASET(System Q)	11-89
11.5.3	RRPA (serie A)	11-101
11.5.4	RIRD (serie A)	11-109
11.5.5	RIRD (serie QnA, System Q)	11-115
11.5.6	RIWT (serie A)	11-123
11.5.7	RIWT (serie QnA, System Q)	11-130
11.5.8	RIRCV (serie A)	11-138
11.5.9	RIRCV (serie QnA, System Q)	11-144
11.5.10	RISEND (serie A)	11-150
11.5.11	RISEND (serie QnA e System Q)	11-156
11.5.12	RITO (serie A)	11-162
11.5.13	RITO (serie QnA e System Q)	11-166
11.5.14	RIFR (serie A)	11-170
11.5.15	RIFR (serie QnA e System Q)	11-174

12	Modalità Microcomputer (AnN(S))	
12.1	Capacità di memorizzazione ed aree di memoria	12-1
12.2	Utilizzo di programmi microcomputer creati dall'utente	12-2
12.2.1	Mappa di memoria	12-3
12.2.2	Configurazione indirizzi dell'area memorizzazione dati	12-3
12.2.3	Configurazione area memoria dati	12-4
13	Codici di errore	
13.1	Tabella dei codici di errore; CPU Q00J, Q00 e Q01	13-2
13.2	Tabella dei codici di errore; CPU QnA e System Q	13-12
13.3	Tabella di codici di errore; serie A (tranne AnA e AnAS)	13-39
13.4	Tabella di codici di errore; CPU AnA e AnAS	13-43
A	Appendice A	
A.1	Definizione dei tempi di elaborazione	A-1
A.2	Tempi di elaborazione	A-2
A.2.1	Tabella dei tempi di elaborazione (serie QnA e System Q)	A-3
A.2.2	Tempi di elaborazione delle CPU serie A	A-23
A.3	Confronto fra CPU	A-32
A.3.1	Operandi disponibili	A-32
A.3.2	Modi di controllo I/O	A-34
A.3.3	Tipi di dati	A-34
A.3.4	Confronto timer	A-35
A.3.5	Confronto dei contatori	A-38
A.3.6	Confronto istruzioni di visualizzazione	A-39
A.3.7	Istruzioni serie Q e System Q equivalenti alle istruzioni per serie A	A-40
A.3.8	Confronto fra CPU QnA/Q2AS e CPU MELSEC System Q	A-41
A.4	Panoramica dei relé speciali	A-43
A.4.1	Tabella dei bit speciali diagnostici (MELSEC serie Q e System Q)	A-43
A.4.2	Tabella bit speciali (M) (serie A)	A-64
A.4.3	Tabella bit di comunicazione (solo serie A)	A-71
A.5	Tabella dei registri speciali	A-74
A.5.1	Tabella dei bit speciali (MELSEC serie Q e System Q)	A-74
A.5.2	Tabella registri speciali (D) (solo serie A)	A-111
A.5.3	Tabella registri di comunicazione (solo serie A)	A-120

1 Introduzione

Questo manuale descrive la programmazione e l'elaborazione delle istruzioni per sequenze e applicative, contenute nelle CPU delle serie MELSEC A e Q.

1.1 Ulteriori manuali

Manuale di programmazione QCPU/QnACPU (Istruzioni PID)

- Descrizione delle istruzioni del regolatore PID

Manuale di programmazione QnPHCPU (Istruzioni per controllo di processo)

- Descrizione delle istruzioni del regolatore PID

Manuale di programmazione (AD57/58)

- Descrizione delle istruzioni specifiche per i moduli funzione speciali AD57/58

Manuale di programmazione QCPU/QnACPU (SFC)

- Descrizione delle istruzioni per diagrammi funzionali sequenziali

Manuali operativi GX Developer

- Fondamenti di programmazione di GX Developer

GX IEC Developer Manuale introduttivo

- Fondamenti di programmazione di GX IEC Developer

GX IEC Developer Manuale di riferimento

- Descrizione dettagliata della programmazione con GX IEC Developer

- Descrizione delle istruzioni IEC (libreria standard IEC)

NOTA

Tutti i manuali sono elencati nel nostro listino PLC attuale.

I manuali possono anche essere scaricati in formato PDF dalla homepage MITSUBISHI ELECTRIC (www.mitsubishi-automation.com).

1.2 Tipi di CPU

Le funzioni descritte in questo manuale possono essere impiegate in tutti i tipi di CPU dalle attuali versioni di GX Developer e GX IEC Developer se la CPU desiderata supporta le istruzioni.

I diversi tipi di PLC e le relative CPU specifiche sono elencate in dettaglio di seguito:

Tipo PLC		Tipo CPU
Serie A	AnA/AnU	A2A, A2A-S1, A2U, A2U-S1, A3A, A3U
	AnAS/AnUS	A2AS, A2AS-S1, A2AS-S30, A2AS-S60, A2US, A2US-S1
	AnN	A1, A2, A2C A3M, A3N
	AnS	A1S, A1S-S1, A2S, A2S-S1
Serie Q	QnA	Q2A/Q2AS, Q2A-S1/Q2AS-S Q3A Q4A, Q4AR
System Q	Q (tipi con singolo processore)	Q00J
	Q (tipi multi processore)	Q00, Q01 (uso limitato a un sistema multi-CPU) Q02, Q02H, Q06H, Q12H, Q12PH, Q25H, Q25PH Moduli PC-CPU: PPC-CPU686(MS)-64 PPC-CPU686(MS)-128 In un sistema multi-CPU si possono usare fino a 4 tipi di CPU PLC, per condividere compiti di controllo e comunicazione.

Quando nelle tabelle sono menzionati i tipi A e Q, vengono inclusi tutti i tipi di CPU della serie e della serie Q/System Q. Le eccezioni vengono evidenziate separatamente.

1.3 Software

Tutte le istruzioni descritte, con poche eccezioni, possono essere applicate con i pacchetti software disponibili:

- GX Developer
- GX IEC Developer

Gli esempi di programma contenuti in questo manuale sono stati creati con GX IEC Developer. La rappresentazione a lista istruzioni MELSEC (IL) corrisponde generalmente con quella di GX Developer.

Tutte le istruzioni descritte in questo manuale sono incluse nella libreria standard di GX IEC Developer.

A seconda della CPU selezionata, le istruzioni disponibili nelle finestre di dialogo di GX IEC Developer sono solo quelle che possono essere processate dalla CPU.

1.4 Ricerca di una istruzione

Avanzata

Se si è già familiarità con la programmazione delle istruzioni delle serie A e Q MELSEC e con il System Q, guardare direttamente alle istruzioni dei capitoli da 5 a 9. La riga di intestazione contiene il nome dell'istruzione come viene riferita da GX Developer e dall'editor MELSEC di GX IEC Developer.

Principianti

Se non si possiede familiarità nel adoperare le istruzioni, procedere come segue:

- Leggere tutto il capitolo 3, relativo alla diversa rappresentazione delle istruzioni negli editor MELSEC e IEC.
- Leggere tutto il capitolo 4 che contiene un layout consistente e strutturato della descrizione di ciascuna istruzione.
- Usare
 - la panoramica tabellare delle categorie di istruzioni, complete di una breve descrizione, del capitolo 2
 - l'indice che comprende tutte le istruzioni

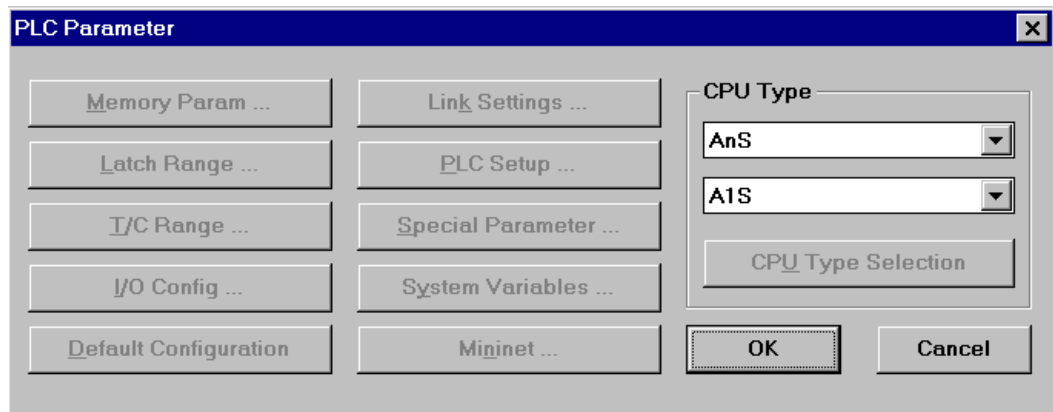
NOTA

Tutte le istruzioni contenute in questo manuale sono anche incluse nella guida in linea di GX IEC Developer con lo stesso livello di dettaglio.

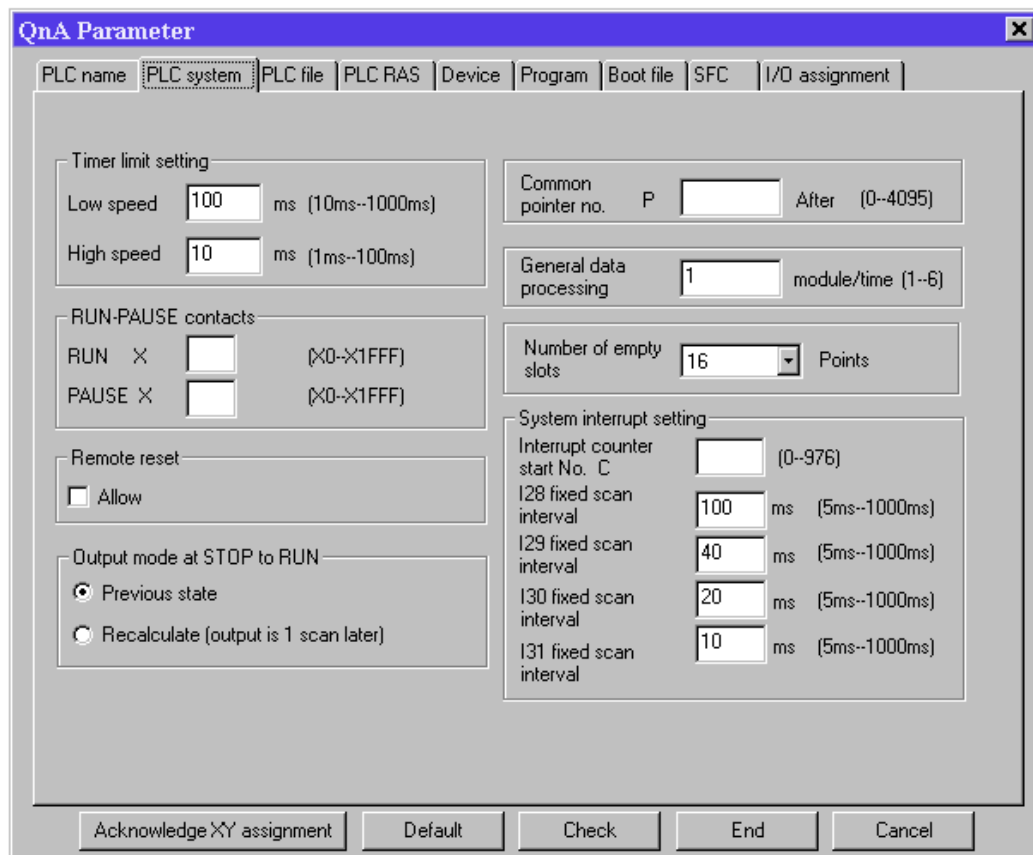
1.5 Parametri PLC

I parametri consentono di configurare diverse funzioni, campi di dispositivi, ecc. Per la programmazione delle funzioni descritte in questo manuale, l'impostazione dei parametri può rimanere quella di default, oppure personalizzata per le necessità dell'utente. Fare riferimento ai manuali hardware appropriati delle CPU ed ai manuali di programmazione, per una descrizione dettagliata dell'impostazione dei parametri PLC.

Esempio: GX IEC Developer



Esempio: GX Developer, GX IEC Developer 6.0



1.6 Confronto fra GX IEC Developer e GX Developer

Le caratteristiche più importanti di GX IEC Developer e di GX Developer sono elencate nella tabella seguente:

GX IEC Developer	GX Developer
Uso strutturato	Semplice da usare
Programmazione conforme a IEC 1131	—
Editor: Lista istruzioni, schema a contatti, testo strutturato, SFC, FUB	Editor: Lista istruzioni, schema a contatti, SFC
Funzioni e blocchi funzione	Blocchi funzione (V7.0 o successiva)
Modifiche online al programma	Modifiche online al programma
Funzioni diagnostiche per il PLC	Funzioni diagnostiche per il PLC
Funzioni diagnostiche per i sistemi di rete	Funzioni diagnostiche per i sistemi di rete

2 Tabelle istruzioni

2.1 Suddivisione delle istruzioni

Le istruzioni sono suddivise in quattro categorie principali:

- Istruzioni di sequenza
- Istruzioni applicative, parte 1
- Istruzioni applicative, parte 2
- Istruzioni per collegamento dati

Le categorie di istruzioni sono descritte in maggiore dettaglio nella tabella seguente:

Categoria dell'istruzione		Descrizione	Riferimento
Istruzioni di sequenza	Istruzione di ingresso	Inizio operazione, collegamento contatti in serie e parallelo.	Cap. 5.1
	Istruzione di collegamento	Collegamento blocchi in serie e parallelo, memorizzazione ed elaborazione risultati operazione, inversione risultato operazione, conversione risultato operazione in impulsi, comando di relé comandati su fronte.	Cap. 5.2
	Istruzioni di uscita	Operandi a bit, contatti di contatori e temporizzatori, uscita, set, e reset di spie di errore, set e reset di operandi, uscita su fronte di salita e discesa, inversione uscita operandi a bit, generazione di impulsi.	Cap. 5.3
	Istruzioni di scorrimento	Scorrimento di operandi a bit	Cap. 5.4
	Istruzioni di controllo principale	Set e reset di singole parti di un programma.	Cap. 5.5
	Istruzione di terminazione	Fine di una parte di programma, fine sequenza e routine di programma.	Cap. 5.6
	Istruzioni miscellanee	Arresto sequenza di programma, nessuna operazione.	Cap. 5.7
Istruzioni di applicazione Parte 1	Istruzioni per operazioni di confronto	Confronto di dati fra loro (ad es. =, >, ≥)	Cap. 6.1
	Istruzioni per operazioni aritmetiche	Somma, sottrazione, moltiplicazione divisione, incremento e decremento di dati BIN e BCD, dati in virgola mobile e blocchi dati BIN, collegamento di stringhe di caratteri	Cap. 6.2
	Istruzioni di conversione dati	Conversione fra tipi di dati, ad es. BCD → BIN, BIN → BCD	Cap. 6.3
	Istruzioni per trasferimento dati	Trasmette un insieme di dati definiti	Cap. 6.4
	Istruzioni di salto di programma	Comandi di salto di programma	Cap. 6.5
	Istruzioni per controllo esecuzione del programma	Abilita e disabilita gli interrupt di programma	Cap. 6.6
	Istruzione di rinfresco	Rinfresca operandi a bit, collegamenti e interfacce di I/O	Cap. 6.7
	Altre istruzioni utili	Conteggio avanti o indietro ingressi a 1 o 2 fasi, timer apprendimento, timer per funzioni speciali, controllo percorso più breve tavola rotante, segnale rampa, misura densità di impulso, uscita impulsiva a ciclo fisso, modulazione larghezza impulso, ingresso matrice	Cap. 6.8

Categoria dell'istruzione		Descrizione	Riferimento
Istruzioni di applicazione Parte 2	Istruzioni per operazioni logiche	AND / OR logico, OR / NOR esclusivo logico,	Cap. 7.1
	Istruzioni di rotazione	Rotazione destra / sinistra dati a 16 e 32 bit	Cap. 7.2
	Istruzioni di scorrimento	Scorrimento dati di bit o parole	Cap. 7.3
	Istruzioni di elaborazione bit	Set, reset, e test di bit	Cap. 7.4
	Istruzioni per elaborazione dati	Ricerca, codifica e decodifica dati di operandi specificati, disgrega e aggrega dati	Cap. 7.5
	Istruzioni per programmi strutturati	Operazione ripetuta, chiamata subroutine di programma, richiamo subroutine fra file di programma, commutazione fra programma principale e sottoprogramma, chiamate programmi microcomputer, indicizzazione schema a contatti, memorizzazione valori di indicizzazione in tabelle dati	Cap. 7.6
	Istruzioni per operazioni su tabelle dati	Scrittura e lettura dati da una tabella, cancellazione e inserimento di blocchi dati in una tabella	Cap. 7.7
	Istruzioni di accesso a buffer di memoria	Accesso a buffer di memoria da parte di moduli funzione speciali o moduli remoti	Cap. 7.8
	Istruzioni di visualizzazione	Invio di caratteri ASCII sull'uscita di un modulo o su un display a LED	Cap. 7.9
	Istruzioni per debug e diagnosi guasti	Controllo guasti, set e reset di memorie di stato, traccia campionamento, traccia programma	Cap. 7.10
	Istruzioni di elaborazione stringhe di caratteri	Elaborazione stringhe di caratteri (codice ASCII)	Cap. 7.11
	Istruzioni per funzioni speciali	Funzioni trigonometriche, radice quadrata e calcolo esponenziale con dati BCD e dati in virgola mobile	Cap. 7.12
	Istruzioni per controllo dati	Controllo limite inferiore e superiore e memorizzazione dei dati controllati	Cap. 7.13
	Istruzioni di commutazione file registri	Commutazione fra blocchi di registri e file registri	Cap. 7.14
	Istruzioni per orologio	Scrittura e lettura dei dati orologio	Cap. 7.15
	Istruzioni per dispositivi periferici	Invio messaggi e acquisizione tasti da unità periferiche	Cap. 7.16
	Istruzioni di programma	Seleziona diversi modi di esecuzione del programma	Cap. 7.17
Altre istruzioni	Reset timer di watchdog (WDT), set e reset riporto, generazione impulso, lettura diretta da registri file ad accesso indiretto, ingresso tasti numerici da tastiera, salvataggio o ripristino in blocco di registri indice, scrittura dei file registri su EEPROM	Cap. 7.18	
Istruzioni per collegamento dati	Istruzioni per rinfresco rete	Istruzioni per operazioni di rinfresco dati sui moduli di rete.	Cap. 8.5
	Istruzioni dedicate per collegamenti dati	Letture e scritture dati da e per stazioni oggetto in reti oggetto; invio dati su moduli di rete in stazioni oggetto di reti oggetto; ricezione dati inviati con istruzione SEND; richieste dati da stazioni diverse (operazioni di scrittura/lettura con dati orologio, operazioni RUN/STOP); lettura e scrittura dati da e per moduli funzione speciali in stazioni di I/O remote,	Cap. 8.6
	Istruzioni per collegamento dati compatibili con serie A	Letture e scritture dati da e per stazioni oggetto di reti diverse, lettura e scrittura dati da e per stazioni locali (solo su stazione master), lettura e scrittura dati da e per moduli funzione speciali in stazioni I/O remoto.	Cap. 8.7
	Letture/scritture dati di instradamento	Letture e scritture di parametri di routing (numero di rete e numero stazione della stazione di smistamento, numero di stazione della stazione di routing)	Cap. 8.8

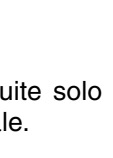

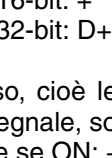

Categoria dell'istruzione		Descrizione	Riferimento
Istruzioni per una CPU del System Q	Lettura informazioni modulo	Legge le informazioni del modulo dal numero designato di intestazione di I/O	Cap. 9.1
	Set / reset traccia	Memorizza i dati della traccia nel file di traccia nella memory card	Cap. 9.2
	Scrittura e lettura da file	Scrive dati nel file designato. Legge dati dal file designato.	Cap. 9.3
	Istruzioni di programma	Carica, scarica, carica + scarica programma da memory card	Cap. 9.4
	Trasferimento dati	Trasferimento blocchi di file registri ad alta velocità	Cap. 9.5
	Scambio dati in un sistema multi-CPU	Scrive nella memoria condivisa della CPU Lettura dalla memoria condivisa di una diversa CPU	Cap. 9.6
Istruzioni per Q4ARCPU	Impostazione modo	Impostazione modo operativo per avviamento CPU e per commutazione fra sistema attivo e sistema in stand-by	Cap. 10.1
	Trasferimento dati	Trasferimento dati dalla PU del sistema attivo alla CPU del sistema in stand-by Trasferimento in blocco di dati dalla memoria buffer dei moduli funzione speciali	Cap. 10.2

2.2 Panoramica delle istruzioni

2.2.1 Descrizione delle tabelle di panoramica

Le sezioni che seguono, da 2.3 a 2.6 comprendono una panoramica di tutte le istruzioni descritte in questo manuale.

Di seguito viene descritto in dettaglio il layout della tabella riassuntiva:

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Somma e sottrazione di dati binari a 16 bit	+	s, d	(d)+(s) → (d)		3	5	6.2.1
	+P						
	+	s1, s2, d1	(s1)+(s2) → (d1)		4	7	6.2.1
	+P						

↑
↑
↑
↑
↑
↑
↑
↑

(1) (2) (3) (4) (5) (6) (7) (8)

Significato delle diverse colonne:

(1) Categoria dell'istruzione

(2) Specifica del nome istruzione ("comando") per la programmazione

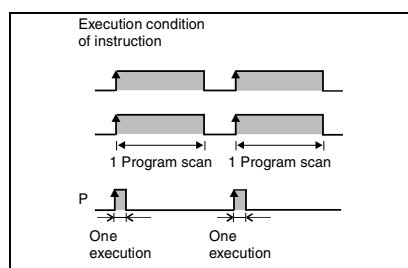
I nomi delle istruzioni sono rappresentati in notazione MELSEC (fare riferimento alla sezione 3.2 per una spiegazione della notazione).

In generale, sono rappresentate le istruzioni a 16 bit. Tutte le istruzioni a 32 bit sono indicate da una "D" iniziale.

Esempio: - istruzione 16-bit: +
 - istruzione 32-bit: D+

Le istruzioni a impulso, cioè le istruzioni che sono eseguite solo in corrispondenza del fronte positivo di un segnale, sono indicate da una "P" finale.

Esempio: - esecuzione se ON: +
 - esecuzione su fronte di salita: +P



Le istruzioni e l'elaborazione di stringhe di caratteri sono indicate da un "\$" iniziale

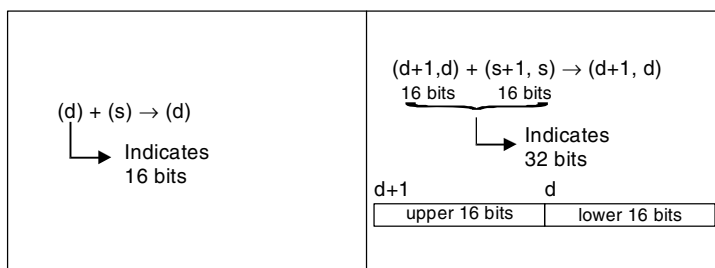
- Esempio: - istruzione standard: +
- istruzione per stringa di caratteri: \$+

(3) Specifica delle variabili

Qui vengono specificate le variabili da usare. La sorgente dati è indicata da una "s", mentre la destinazione è rappresentata da una "d".

- Esempio: s = se esiste solo una sorgente dati
- s1, s2 = se esistono più sorgenti dati
- s+0, s+1, (s1)+0, (s1)+1 = per istruzioni a 32 bit
cioè s1 = registro dati D0, (s1)+1 = registro dati D1
- s+0, s+1, s+2, s+3 = 4 operandi in sequenza, ad es. una matrice

(4) Significato ed elaborazione dell'intera istruzione



(5) Indicazione della condizione di esecuzione, secondo la tabella che segue

Simbolo	Condizione di esecuzione
nessuna indicazione	L'istruzione viene eseguita continuamente e indipendentemente dalla condizione di esecuzione precedente. Se la precondizione non è vera, l'istruzione non viene eseguita.
	L'istruzione viene eseguita se la precondizione è ON. Se la precondizione è OFF, l'istruzione non viene eseguita e non viene effettuata alcuna elaborazione.
	Questa istruzione è una istruzione impulsiva. Viene eseguita una sola volta in corrispondenza del fronte di salita del segnale d'ingresso (cioè se la precondizione passa da OFF a ON). Successivamente, l'istruzione non viene più eseguita anche se il segnale d'ingresso rimane ON.
	Anche questa istruzione è una istruzione impulsiva. Viene eseguita una sola volta in corrispondenza del fronte di discesa del segnale d'ingresso (cioè se la precondizione passa da ON a OFF). Successivamente, l'istruzione non viene più eseguita anche se il segnale d'ingresso rimane ON.

(6+7) Indicazione del numero di passi di programma

Indica il numero di passi necessari per l'esecuzione completa dell'istruzione. Esiste una differenza fra le serie A e Q MELSEC ed il System Q. Fare riferimento alla sezione 3.9 per ulteriori dettagli.

(8) Indicazione del capitolo di riferimento

Indica il capitolo e la sezione di questo manuale che contiene una descrizione dettagliata dell'istruzione.

2.3 Istruzioni di sequenza

2.3.1 Istruzioni di ingresso

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Istruzione d'ingresso	LD		Inizio operazione (Carica (contatto normalmente aperto))		*1	1	5.1.1
	LDI		Inizio operazione (Carica (contatto normalmente chiuso))				
	AND		Collegamento in serie (di contatti norm. aperti)				
	ANI		Collegamento in serie (di contatti norm. chiusi)				
	OR		Collegamento parallelo (di contatti norm. aperti)				
	ORI		Collegamento parallelo (di contatti norm. chiusi)				
	LDP		Inizio operazione impulsiva (fronte di salita)		*1	2	5.1.2
	LDF		Inizio operazione impulsiva (fronte di discesa)				
	ANDP	s	Collegamento impulso in serie (fronte di salita)				
	ANDF	s	Collegamento impulso in serie (fronte di discesa)				
	ORP	s	Collegamento impulso in parallelo (fronte di salita)				
	ORF	s	Collegamento impulso in parallelo (fronte di discesa)				

*: Il numero di passi di programma dipende dagli operandi usati.

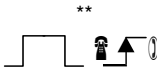
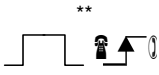





- Usando operandi interni o file registri (da R0 a R32767) : 1
- Usando ingressi ad accesso diretto (DX) : 2
- Usando altri operandi : 3

NOTA: Il numero di passi di programma può raddoppiare se viene utilizzato il file registri 2R su una memory card.


2.3.2 Istruzioni di collegamento

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Istruzione di collegamento	ANB	—	Collegamento in serie di un blocco (collegamento in serie di blocco a contatti)		1	1	5.2.1
	ORB		Collegamento in parallelo di un blocco (collegamento in parallelo di blocco a contatti)				
	MPS	—	Elaborazione risultato operazione (memorizza risultato operazione (inserimento in memoria))		1	1	5.2.2
	MRD		Elaborazione risultato operazione (lettura risultato operazione (lettura da memoria))				
	MPP		Elaborazione risultato operazione (lettura e cancellazione risultato operazione (estrazione da memoria))				
	INV	—	Inversione risultato operazione (istruzione di inversione)		1		5.2.3
	MEP	—	Conversione risultato operazione in impulso (generazione impulso su fronte di salita del risultato operazione)		1		5.2.4
	MEF		Conversione risultato operazione in impulso (generazione impulso su fronte di discesa del risultato operazione)				
	EGP	d	Eccitazione di relé comandi su fronte. (eccitazione di un relé con il fronte di salita di un risultato operazione)		1		5.2.5
	EGF		Eccitazione di relé comandi su fronte. (eccitazione di un relé con il fronte di discesa di un risultato operazione)				



2.3.3 Istruzioni di uscita

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Istruzione di uscita	OUT	d	Istruzioni di impostazione uscite		*1	*1	5.3.1
	SET	d	Attivazione operandi		*1		5.3.5
	RST	d	Disattivazione operandi		2	*1	5.3.6
	PLS	d	Eccitazione uscita su fronte di salita		2	*3	5.3.8
	PLF		Eccitazione uscita su fronte di discesa				
	FF	s	Inversione di operandi di uscita a bit		2		5.3.9
	DELTA	d	Generazione di impulsi su uscite ad accesso diretto		2		5.3.11
	DELTAP						

*: Il numero di passi di programma dipende dagli operandi usati.
Per il numero di passi esatto, fare riferimento al capitolo indicato.

** : Questa  condizione di esecuzione è valida solo se viene usata la spia di errore (F).

2.3.4 Istruzioni di scorrimento

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Istruzioni di scorrimento	SFT	d	Scorrimento di operandi a bit		2	*3	5.4.1
	SFTP						

*: Fare riferimento al capitolo 3.9.2 "Per una CPU AnA, AnAS, e AnU" per il relativo numero di passi.

2.3.5 Istruzioni di controllo primario

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Istruzioni di controllo primario	MC	n, d	Attivazione di parti definite del programma		2	*3/5	5.5.1
	MCR	n	Disattivazione di parti definite del programma		1		

*: Il numero di passi è 5 per tutte le istruzioni MC e 3 per l'istruzione MCR.
Fare riferimento al capitolo 3.9.2 "Per una CPU AnA, AnAS, e AnU" per il relativo numero di passi.

2.3.6 Istruzioni di terminazione programma

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Istruzione di terminazione	FEND	—	Fine diramazione di programma		1		5.6.1
	END		Terminazione del programma sequenza				5.6.2

2.3.7 Istruzioni miscelanee

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Istruzioni miscelanee	STOP	—	Istruzione di arresto		1		5.7.1
	NOP	—	Nessun passo di programma				5.7.2

2.4 Istruzioni applicative, parte 1

2.4.1 Istruzioni per operazioni di confronto

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Confronto dati BIN a 16-bit	LD=	s1, s2	Attiva l'uscita se s1 = s2		3	* 5/7	6.1.1
	AND=						
	OR=						
	LD<>	s1, s2	Attiva l'uscita se s1 ≠ s2		3	* 5/7	6.1.1
	AND<>						
	OR<>						
	LD>	s1, s2	Attiva l'uscita se s1 > s2		3	* 5/7	6.1.1
	AND>						
	OR>						
	LD<=	s1, s2	Attiva l'uscita se s1 ≤ s2		3	* 5/7	6.1.1
	AND<=						
	OR<=						
	LD<	s1, s2	Attiva l'uscita se s1 < s2		3	* 5/7	6.1.1
	AND<						
	OR<						
	LD>=	s1, s2	Attiva l'uscita se s1 ≥ s2		3	* 5/7	6.1.1
AND>=							
OR>=							

*: Il numero di passi di programma dipende dagli operandi usati.
Per il numero di passi esatto, fare riferimento al capitolo indicato.

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Confronto dati BIN a 32-bit	LDD=	s1, s2	Attiva l'uscita se s1 = s2		*3	**	6.1.2
	ANDD=					11	
	ORD=						
	LDD<>	s1, s2	Attiva l'uscita se s1 ≠ s2		*3	**	6.1.2
	ANDD<>					11	
	ORD<>						
	LDD>	s1, s2	Attiva l'uscita se s1 > s2		*3	**	6.1.2
	ANDD>					11	
	ORD>						
	LDD<=	s1, s2	Attiva l'uscita se s1 ≤ s2		*3	**	6.1.2
	ANDD<=					11	
	ORD<=						
	LDD<	s1, s2	Attiva l'uscita se s1 < s2		*3	**	6.1.2
	ANDD<					11	
	ORD<						
LDD>=	s1, s2	Attiva l'uscita se s1 ≥ s2		*3	**	6.1.2	
ANDD>=					11		
ORD>=							

*: Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.

- Se viene usata una CPU QnA oppure una CPU a processore singolo del System Q: 3
- Se viene usata una CPU multiprocessore del System Q con operandi interni a word (tranne file registri ZR): 5
costanti: 5
Operandi a bit con numero operando multiplo di 16, la cui sigla di designazione è K8, e che non usa indicizzazione: 5
- Se viene usata una CPU multiprocessore del System Q con operandi diversi da quelli menzionati: 5

La velocità di elaborazione è maggiore con CPU del System Q, anche se il numero dei passi è superiore.



** : Il numero di passi di programma dipende dagli operandi usati.
Per il numero di passi esatto, fare riferimento al capitolo indicato.

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Confronto dati in virgola mobile	LDE=	s1, s2	Attiva l'uscita se $s1 = s2$		3		6.1.3
	ANDE=						
	ORE=						
	LDE<>	s1, s2	Attiva l'uscita se $s1 \neq s2$		3		6.1.3
	ANDE<>						
	ORE<>						
	LDE>	s1, s2	Attiva l'uscita se $s1 > s2$		3		6.1.3
	ANDE>						
	ORE>						
	LDE<=	s1, s2	Attiva l'uscita se $s1 \leq s2$		3		6.1.3
	ANDE<=						
	ORE<=						
	LDE<	s1, s2	Attiva l'uscita se $s1 < s2$		3		6.1.3
	ANDE<						
	ORE<						
LDE>=	s1, s2	Attiva l'uscita se $s1 \geq s2$		3		6.1.3	
ANDE>=							
ORE>=							

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Confronto dati fra stringhe di caratteri	LD\$=	s1, s2	*confronta le stringhe di caratteri in s1 e s2 carattere per carattere. Attiva l'uscita se s1 = s2		3		6.1.4
	AND\$=						
	OR\$=						
	LD\$<>	s1, s2	*confronta le stringhe di caratteri in s1 e s2 carattere per carattere. Attiva l'uscita se s1 ≠ s2		3		6.1.4
	AND\$<>						
	OR\$<>						
	LD\$>	s1, s2	*confronta le stringhe di caratteri in s1 e s2 carattere per carattere. Attiva l'uscita se s1 > s2		3		6.1.4
	AND\$>						
	OR\$>						
	LD\$<=	s1, s2	*confronta le stringhe di caratteri in s1 e s2 carattere per carattere. Attiva l'uscita se s1 <= s2		3		6.1.4
	AND\$<=						
	OR\$<=						
	LD\$<	s1, s2	*confronta le stringhe di caratteri in s1 e s2 carattere per carattere. Attiva l'uscita se s1 < s2		3		6.1.4
	AND\$<						
	OR\$<						
LD\$>=	s1, s2	*confronta le stringhe di caratteri in s1 e s2 carattere per carattere. Attiva l'uscita se s1 >= s2		3		6.1.4	
AND\$>=							
OR\$>=							

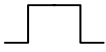

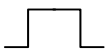

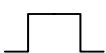

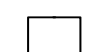
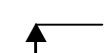
*: Condizioni che regolano il confronto di stringhe di caratteri:

- Corrispondenza: Tutti i caratteri delle stringhe devono corrispondere.
- Stringa maggiore: Se le stringhe sono diverse, individua la stringa più grande.
- Stringa minore: Se le stringhe sono diverse, individua la stringa più piccola.

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Confronto blocchi dati BIN	BKCMP=	s1, s2, n, d1	<p>Confronta l'ennesimo blocco BIN a 16 bit in s1 con l'ennesimo blocco BIN a 16 bit in s2, iniziando con il primo numero operando.</p> <p>Il risultato del confronto è memorizzato a partire da d1 in avanti.</p>		5		6.1.5
	BKCMP<>	s1, s2, n, d1					
	BKCMP>	s1, s2, n, d1					
	BKCMP<=	s1, s2, n, d1					
	BKCMP<	s1, s2, n, d1					
	BKCMP>=	s1, s2, n, d1					
	BKCMP=P	s1, s2, n, d1					
	BKCMP<>P	s1, s2, n, d1					
	BKCMP>P	s1, s2, n, d1					
	BKCMP<=P	s1, s2, n, d1					
	BKCMP<P	s1, s2, n, d1					
	BKCMP>=P	s1, s2, n, d1					

2.4.2 Istruzioni per operazioni aritmetiche

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Operazioni di somma e sottrazione dati BIN a 16 bit	+	s, d	$(d)+(s) \rightarrow (d)$		3	5	6.2.1
	+P						6.2.1
	+	s1, s2, d1	$(s1)+(s2) \rightarrow (d1)$		4	7	6.2.1
	+P						6.2.1
	-	s, d	$(d)-(s) \rightarrow (d)$		3	5	6.2.1
	-P						6.2.1
	-	s1, s2, d1	$(s1)-(s2) \rightarrow (d1)$		4	7	6.2.1
	-P						6.2.1

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Operazioni di somma e sottrazione dati BIN a 32 bit	D+	s, d	$(d+1, d)+(s+1, s)$ $\rightarrow (d+1, d)$		*3	9	6.2.2
	D+P						6.2.2
	D+	s1, s2, d1	$((s1)+1, s1)+((s2)+1, s2)$ $\rightarrow ((d1)+1, d1)$		**4	11	6.2.2
	D+P						6.2.2
	D-	s, d	$(d+1, d)-(s+1, s)$ $\rightarrow (d+1, d)$		*3	9	6.2.2
	D-P						6.2.2
	D-	s1, s2, d1	$((s1)+1, s1)-((s2)+1, s2)$ $\rightarrow ((d1)+1, d1)$		**4	11	6.2.2
	D-P						6.2.2

*: Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.

- Se viene usata una CPU QnA oppure una CPU a processore singolo del System Q: 3
- Se viene usata una CPU multiprocessore del System Q con operandi interni a word (tranne file registri ZR): 5
costanti: 5
Operandi a bit con numero operando multiplo di 16,
la cui sigla di designazione è K8, e che non usa indicizzazione: 5
- Se viene usata una CPU multiprocessore del System Q con operandi diversi da quelli menzionati: 3

** : Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.

- Se viene usata una CPU QnA oppure una CPU a processore singolo del System Q: 4
- Se viene usata una CPU multiprocessore del System Q con operandi interni a word (tranne file registri ZR): 6
costanti: 6
Operandi a bit con numero operando multiplo di 16,
la cui sigla di designazione è K8, e che non usa indicizzazione: 6
- Se viene usata una CPU multiprocessore del System Q con operandi diversi da quelli menzionati: 4

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Moltiplicazione e divisione dati BIN a 16 bit	x	s1, s2, d1	$(s1) \times (s2) \rightarrow ((d1)+1, d1)$		*4	**7	6.2.4
	xP						6.2.4
	/	s1, s2, d1	$(s1)/(s2) \rightarrow$ Quoziente (d1), resto ((d1)+1)		*4	**7	6.2.4
	/P						6.2.4
Moltiplicazione e divisione dati BIN a 32 bit	Dx	s1, s2, d1	$((s1)+1, s1) \times ((s2)+1, s2) \rightarrow$ $((d1)+3, (d1)+2, (d1)+1, d1)$		*4	**11	6.2.4
	DxP						6.2.4
	D/	s1, s2, d1	$((s1)+1, s1) / ((s2)+1, s2) \rightarrow$ Quoziente ((d1)+1, d1), resto ((d1)+3, (d1)+2)		*4	**11	6.2.4
	D/P						6.2.4

*: Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.

- Se viene usata una CPU QnA oppure una CPU a processore singolo del System Q: 4
- Se viene usata una CPU multiprocessore del System Q con operandi interni a word (tranne file registri ZR): 3
costanti: 3
Operandi a bit con numero operando multiplo di 16, la cui sigla di designazione è K8, e che non usa indicizzazione: 3
- Se viene usata una CPU multiprocessore del System Q con operandi diversi da quelli menzionati: 4

** : Il numero di passi di programma dipende dagli operandi usati.
Per il numero di passi esatto, fare riferimento al capitolo indicato.

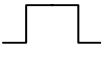
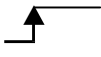
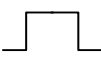

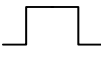
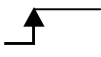
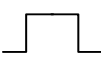

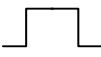

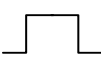

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Operazioni di somma e sottrazione dati BCD a 4 cifre	B+	s, d	$(d)+(s) \rightarrow (d)$		3	*7	6.2.5
	B+P						6.2.5
	B+	s1, s2, d1	$(s1)+(s2) \rightarrow (d1)$		4	*9	6.2.5
	B+P						6.2.5
	B-	s, d	$(d)-(s) \rightarrow (d)$		3	*7	6.2.5
	B-P						6.2.5
	B-	s1, s2, d1	$(s1)-(s2) \rightarrow (d1)$		4	*9	6.2.5
	B-P						6.2.5
Operazioni di somma e sottrazione dati BCD a 8 cifre	DB+	s, d	$(d+1, d)+(s+1, s) \rightarrow (d+1, d)$		3	*9	6.2.6
	DB+P						6.2.6
	DB+	s1, s2, d1	$((s1)+1, s1)+((s2)+1, s2) \rightarrow ((d1)+1, d1)$		4	*1 1	6.2.6
	DB+P						6.2.6
	DB-	s, d	$(d+1, d)+(s+1, s) \rightarrow (d+1, d)$		3	*9	6.2.6
	DB-P						6.2.6
	DB-	s1, s2, d1	$((s1)+1, s1)+((s2)+1, s2) \rightarrow ((d1)+1, d1)$		4	*1 1	6.2.6
	DB-P						6.2.6

** : Il numero di passi di programma dipende dagli operandi usati.
Per il numero di passi esatto, fare riferimento al capitolo indicato.

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Operazioni di moltiplicazione e divisione dati BCD a 4 cifre	B \times	s1, s2, d1	$(s1) \times (s2) \rightarrow ((d1)+1, d1)$		4	*9	6.2.7
	B \times P						6.2.7
	B/	s1, s2, d1	$(s1)/(s2) \rightarrow$ Quoziente (d1), resto ((d1)+1)		4	*9	6.2.7
	B/P						6.2.7
Operazioni di moltiplicazione e divisione dati BCD a 8 cifre	DB \times	s1, s2, d1	$((s1)+1, s1) \times ((s2)+1, s2) \rightarrow$ $((d1)+3, (d1)+2,$ $(d1)+1, d1)$		4	*11	6.2.8
	DB \times P						6.2.8
	DB/	s1, s2, d1	$((s1)+1, s1) / ((s2)+1, s2) \rightarrow$ Quoziente ((d1)+1, d1), resto ((d1)+3, (d1)+2)		4	*11	6.2.8
	DB/P						6.2.8

*: Il numero di passi di programma dipende dagli operandi usati.
Per il numero di passi esatto, fare riferimento al capitolo indicato.

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Operazioni di somma e sottrazione dati in virgola mobile	E+	s, d	$(d+1, d)+(s+1, s) \rightarrow (d+1, d)$		3		6.2.9
	E+P						6.2.9
	E+	s1, s2, d1	$((s1)+1, s1)+((s2)+1, s2) \rightarrow ((d1)+1, d1)$		4		6.2.9
	E+P						6.2.9
	E-	s, d	$(d+1, d)-(s+1, s) \rightarrow (d+1, d)$		3		6.2.9
	E-P						6.2.9
	E-	s1, s2, d1	$((s1)+1, s1)-((s2)+1, s2) \rightarrow ((d1)+1, d1)$		4		6.2.9
	E-P						6.2.9
Operazioni di moltiplicazione e divisione dati in virgola mobile	Ex	s1, s2, d1	$((s1)+1, s1) \times ((s2)+1, s2) \rightarrow ((d1)+1, d1)$		4		6.2.10
	ExP						6.2.10
	E/	s1, s2, d1	$((s1)+1, s1) / ((s2)+1, s2) \rightarrow$ Quoziente $((d1)+1, d1)$		4		6.2.10
	E/P						6.2.10
Operazioni di somma e sottrazione di blocchi BIN	BK+	s1, s2, d, n	Somma l'ennesimo blocco a 16-bit in s1 con l'ennesimo blocco a 16-bit in s2.		5		6.2.11
	BK+P						6.2.11
	BK-	s1, s2, d, n	Sottrae l'ennesimo blocco a 16-bit in s1 dall'ennesimo blocco a 16-bit in s2.		5		6.2.11
	BK-P						6.2.11

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Operazioni di collegamento di stringhe di caratteri	\$+	s, d	I dati della stringa di caratteri in s vengono appesi ai caratteri in d. La stringa di caratteri risultante viene memorizzata in d.		3		6.2.12
	\$+P						6.2.12
	\$+	s1, s2, d1	I dati della stringa di caratteri in s vengono appesi ai caratteri in d. La stringa di caratteri risultante viene memorizzata in d.		4		6.2.12
	\$+P						6.2.12
Operazioni di incremento dati BIN	INC	d	(d)+1 → (d)		2	**3	6.2.13
	INCP						6.2.13
	DINC	d	(d+1, d)+1 → (d+1, d)		*2	**3	6.2.13
	DINCP						6.2.13
Operazioni di decremento dati BIN	DEC	d	(d)-1 → (d)		2	**3	6.2.14
	DECP						6.2.14
	DDEC	d	(d+1, d)-1 → (d+1, d)		*2	**3	6.2.14
	DDECP						6.2.14

*: Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.

- Se viene usata una CPU QnA oppure una CPU a processore singolo del System Q: 2
- Se viene usata una CPU multiprocessore del System Q con operandi interni a word (tranne file registri ZR): 3
costanti: 3
Operandi a bit con numero operando multiplo di 16, la cui sigla di designazione è K8, e che non usa indicizzazione: 3
- Se viene usata una CPU multiprocessore del System Q con operandi diversi da quelli menzionati: 2

** : Il numero di passi di programma dipende dagli operandi usati.

Per il numero di passi esatto, fare riferimento al capitolo indicato.

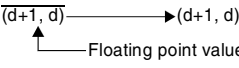
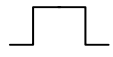
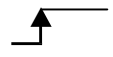
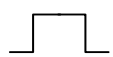



2.4.3 Istruzioni di conversione dati

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Conversione da dati BIN a dati BCD	BCD	s, d	 BCD conversion → (d) ↑ BIN (0 to 9999)		3	*5	6.3.1
	BCDP						
	DBCD	s, d	 BCD conversion (s+1, s) → (d+1, d) ↑ BIN (0 to 99999999)		3	*9	6.3.1
	DBCDP						
Conversione da dati BCD a dati BIN	BIN	s, d	 BIN conversion → (d) ↑ BCD (0 to 9999)		3	*5	6.3.2
	BINP						
	DBIN	s, d	 BIN conversion (s+1, s) → (d+1, d) ↑ BCD (0 to 99999999)		3	*9	6.3.2
	DBINP						
Conversione da dati BIN a dati in virgola mobile	FLT	s, d	 Floating point conversion (s+1, s) → (d) ↑ Binary value (-32768 to 32767)		3		6.3.3
	FLTP						
	DFLT	s, d	 Floating point conversion (s+1, s) → (d+1, d) ↑ Binary value (-2147483648 to 2147483647)		3		6.3.3
	DFLTP						
Conversione da dati in virgola mobile a dati BIN	INT	s, d	 BIN conversion (s+1, s) → (d) ↑ Floating point value (-32768 to 32767)		3		6.3.4
	INTP						
	DINT	s, d	 Floating point conversion (s+1, s) → (d+1, d) ↑ Binary value (-2147483648 bis 2147483647)		3		6.3.4
	DINTP						


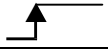







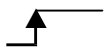
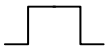

*: Il numero di passi di programma dipende dagli operandi usati.
 Per il numero di passi esatto, fare riferimento al capitolo indicato.

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Conversione da dati BIN a 16 bit a dati BIN a 32 bit	DBL	s, d	Conversion $(s) \rightarrow (d+1, d)$ BIN (-32768 to 32767)		3		6.3.5
	DBLP						6.3.5
Conversione da dati BIN a 32 bit a dati BIN a 16 bit	WORD	s, d	Conversion $(s+1, s) \rightarrow (d)$ BIN (-32768 to 32767)		3		6.3.6
	WORDP						6.3.6
Conversione da dati BIN a 16/32 bit a dati in codice Gray	GRY	s, d	Conversion into Gray code $(s) \rightarrow (d)$ Binary value (-32768 to 32767)		3		6.3.7
	GRYP						6.3.7
	DGRY	s, d	Conversion into Gray code $(s+1, s) \rightarrow (d+1, d)$ Binary value (-2147483648 to 2147483647)		3		6.3.7
	DGRYP						6.3.7
Conversione da dati in codice Gray a dati BIN a 16/32 bit	GBIN	s, d	BIN conversion $(s) \rightarrow (d)$ Gray code (-32768 to 32767)		3		6.3.8
	GBINP						6.3.8
	DGBIN	s, d	BIN conversion $(s+1, s) \rightarrow (d+1, d)$ Gray code (-2147483648 to 2147483647)		3		6.3.8
	DGBINP						6.3.8
Inversione di segno per dati BIN a 16/32 bit (complemento a 2)	NEG	d	$(\bar{d}) \rightarrow (d)$ BIN data		2	*3	6.3.9
	NEGP						6.3.9
	DNEG	d	$(\overline{d+1, d}) \rightarrow (d+1, d)$ BIN data		2	*3	6.3.9
	DNEGP						6.3.9

*: Il numero di passi di programma dipende dagli operandi usati.
 Per il numero di passi esatto, fare riferimento al capitolo indicato.

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Inversione di segno per dati in virgola mobile	ENEG	d			2		6.3.10
	ENEGP						
Conversione da blocchi dati BIN a blocchi dati BCD	BKBCD	s, d, n	Questa istruzione converte ciascun ennesimo blocco dati BIN a 16 bit in s nell'ennesimo blocco dati BCD a 4 cifre. I dati convertiti sono memorizzati in d.		4		6.3.11
	BKBCDP	s, d, n					
Conversione da blocchi dati BCD a blocchi dati BIN	BKBIN	s, d, n	Questa istruzione converte ciascun ennesimo blocco dati BCD a 4 cifre in s nell'ennesimo blocco dati BIN a 16 bit. I dati convertiti sono memorizzati in d.		4		6.3.12
	BKBINP	s, d, n					

2.4.4 Istruzioni di trasferimento dati

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Trasferimento dati BIN a 16-bit	MOV	s, d	(s) → (d)		*3	***5	6.4.1
	MOVP	s, d					
Trasferimento dati BIN a 32-bit	DMOV	s, d	+1, s) → (d+1, d)		**3	***7	6.4.1
	DMOVP	s, d					
Trasferimento dati in virgola mobile	EMOV	s, d	(s+1, s) → (d+1, d) Floating point value		3		6.4.2
	EMOVP	s, d					
Trasferimento dati stringhe di caratteri	\$MOV	s, d	Trasferisce i dati della stringa di caratteri s in d.		3		6.4.3
	\$MOVP	s, d					
Inversione dati BIN a 16-bit	CML	s, d	$\overline{(s)}$ → (d)		*3	***5	6.4.4
	CMLP	s, d					
Inversione dati BIN a 32-bit	DCML	s, d	$\overline{(s+1, s)}$ → (d1+1, d1)		**3	***7	6.4.4
	DCMLP	s, d					

*: Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.

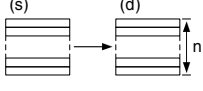

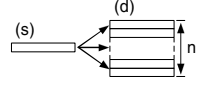



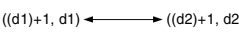
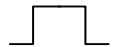
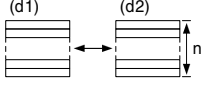

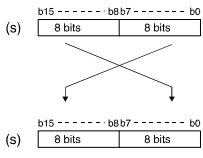
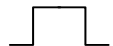
- Se viene usata una CPU QnA oppure una CPU a processore singolo del System Q: 3
- Se viene usata una CPU multiprocessore del System Q con operandi interni a word (tranne file registri ZR): 2
costanti: 2
Operandi a bit con numero operando multiplo di 16, la cui sigla di designazione è K8, e che non usa indicizzazione: 2
- Se viene usata una CPU multiprocessore del System Q con operandi diversi da quelli menzionati: 3

*: Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.

- Se viene usata una CPU a singolo processore del System Q: 2
- Se viene usata una CPU QnA oppure una CPU multiprocessore del System Q: 3


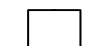
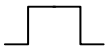
***: Il numero di passi di programma dipende dagli operandi usati.

Per il numero di passi esatto, fare riferimento al capitolo indicato.

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Trasferimento blocchi dati BIN	BMOV	s, n, d			4	*9	6.4.5
	BMOVP	s, n, d					
Trasferimento blocchi dati BIN identici	FMOV	s, n, d			4	*9	6.4.6
	FMOVP	s, n, d					
Scambio dati BIN 16-bit	XCH	d1, d2			3	*5	6.4.7
	XCHP	d1, d2					
Scambio dati BIN 32-bit	DXCH	d1, d2			3	*7	6.4.7
	DXCHP	d1, d2					
Scambio blocchi dati BIN	BXCH	n, d1, d2			4		6.4.8
	BXCHP	n, d1, d2					
Scambio fra byte leggero e byte pesante	SWAP	s			3		6.4.9
	SWAPP	s					

*: Il numero di passi di programma dipende dagli operandi usati.
Per il numero di passi esatto, fare riferimento al capitolo indicato.

2.4.5 Istruzioni di salto di programma

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Istruzioni di salto	CJ	p	Salto condizionale (p = destinazione salto)		2	*3	6.5.1
	SCJ	p	Salto condizionale dalla scansione programma successiva (p = destinazione salto)				
	JMP	p	Istruzioni di salto (p = destinazione salto)		2	*3	6.5.1
	GOEND		Salto a fine programma		1		6.5.2


*: Il numero di passi di programma dipende dagli operandi usati.
Per il numero di passi esatto, fare riferimento al capitolo indicato.

2.4.6 Istruzioni di controllo per interruzione esecuzione programma

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Interruzioni disabilitate	DI		Impedisce l'esecuzione di un programma di interruzione.		1	*1	6.6.1
Interruzioni abilitate	EI		Abilita il richiamo di un programma di interruzione		1	*1	6.6.1
Maschera di bit delle condizioni di esecuzione dei programmi di interruzione	IMASK	s	Nella maschera di bit indicata da s, a ciascun bit è collegato un indirizzo di interruzione.		2	*1	6.6.1
Ritorno da un programma di interruzione al programma principale	IRET		Fine di un programma di interruzione		1	*1	6.6.2

*: Il numero di passi di programma dipende dagli operandi usati.
Per il numero di passi esatto, fare riferimento al capitolo indicato.


2.4.7 Istruzioni per rinfresco dati

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Rinfresco parziale I/O	RFS	s, n	L'istruzione RFS rinfresca gli ingressi e uscite nel campo indicato dei dispositivi di I/O, durante una scansione del programma.		3		6.7.1
Rinfresco parziale I/O	SEG	s, d	L'istruzione SEG abilita il rinfresco di un campo definito di dispositivi di I/O, se la condizione di ingresso è soddisfatta.			*9	6.7.2
Istruzione di rinfresco per dati collegamento e interfaccia e memoria condivisa della CPU	COM		Se SM775 (solo serie Q e System Q) non è attivo (0), i dati di collegamento e interfaccia sono rinfrescati (rinfresco collegamento) mentre viene eseguita l'elaborazione dati generale (elaborazione END). Usato anche per il rinfresco automatico della memoria condivisa multi-CPU		1	*3	6.7.3
Disabilita esecuzione rinfresco collegamento	DI		L'istruzione DI disabilita l'esecuzione del rinfresco di un collegamento fino all'esecuzione di una istruzione EI.		1		6.7.4
Abilita esecuzione rinfresco collegamento	EI		Dopo l'attivazione dell'istruzione EI, viene abilitata l'esecuzione del rinfresco di un collegamento.		1		6.7.4

*: Il numero di passi di programma dipende dagli operandi usati.
Per il numero di passi esatto, fare riferimento al capitolo indicato.

2.4.8 Altre istruzioni utili

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Ingresso a fase singola contatore avanti/ indietro	UDCNT1	s, n, d			4		6.8.1
Ingresso a fase doppia contatore avanti/ indietro	UDCNT2	s, n, d			4		6.8.2
Timer programmabile (apprendimento)	TTMR	d, n	(Impostazione timer) $x n \rightarrow (d)$ $n=0:1, n=1:10, n=2:100$		3		6.8.3
Timer speciale (istruzione per timer a bassa velocità)	STMR	s,n, d	L'istruzione STMR utilizza le uscite indicate da d+0 fino a d+3 per eseguire quattro diverse funzioni di temporizzazione: d+0:Uscita timer OFF ritardata d+1:Uscita timer impulsiva dopo diseccitazione (attivata da fronte di discesa) d+2:Uscita timer impulsiva dopo eccitazione (attivata da fronte di salita) d+3:Uscita timer ON ritardata		3		6.8.4
Timer speciale (istruzione per timer ad alta velocità)	STMRH	s,n, d	vedi sopra		3		6.8.4
Istruzione di posizionamento per tavole rotanti	ROTC	s, n1, n2, d	L'istruzione ROTC ruota il settore indicato da s+2 su una tavola con un numero di settori (divisioni) definito, indicato da n1, fino alla posizione indicata da s+1.		5		6.8.5
Segnale rampa	RAMP	n1, n2, n3, d1, d2	Una istruzione RAMP modifica gradualmente il contenuto di (d1)+0 dal valore iniziale indicato da n1, fino al valore finale indicato da n2.		6		6.8.6
Misura densità di impulso	SPD	s, n, d	L'istruzione SPD conta gli impulsi sull'ingresso indicato da s per il periodo di tempo specificato da n. Il risultato della misura viene memorizzato in d.		4		6.8.7
Uscita a impulsi con numero impostabile di impulsi	PLSY	s1, s2, d	L'istruzione PLSY emette il numero di impulsi specificato da s2 con la frequenza specificata da s1, sull'uscita indicata da d.		4		6.8.8
Modulazione larghezza impulso	PWM	n1, n2, d			4		6.8.9

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Implementazione matrice di ingressi	MTR	s, n, d1, d2	L'istruzione MTR legge informazioni a 16 bit iniziando dall'operando indicato da s. Il numero di ripetizioni (righe) è indicato da n. I valori dei dati letti sono memorizzati a partire dall'operando indicato da d2.		5		6.8.10

2.5 Istruzioni applicative, parte 2

2.5.1 Istruzioni per operazioni logiche

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Prodotto logico	WAND	s, d	$(d) \wedge (s) \longrightarrow (d)$		3	***5	7.1.1
	WANDP						
	WAND	s1, s2, d1	$(s1) \wedge (s2) \longrightarrow (d1)$		4	***7	7.1.1
	WANDP						
	DAND	s, d	$(d+1, d) \wedge (s+1, s) \longrightarrow (d+1, d)$		*4	***9	7.1.1
	DANDP						
	DAND	s1, s2, d	$((s1)+1, s1) \wedge ((s2)+1, s2) \longrightarrow (d+1, d)$		**4		7.1.1
	DANDP						
	BKAND	s1, s2, n, d			5		7.1.2
	BKANDP						

*: Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.


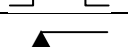

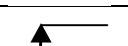
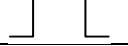

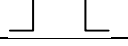

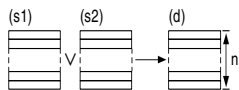
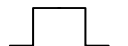
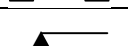
- Se viene usata una CPU QnA: 4
- Se viene usata una CPU a singolo processore del System Q: 3
- Se viene usata una CPU multiprocessore del System Q con operandi interni a word (tranne file registri ZR): 6
costanti: 6
Operandi a bit con numero operando multiplo di 16,
la cui sigla di designazione è K8, e che non usa indicizzazione: 6
- Se viene usata una CPU multiprocessore del System Q con operandi diversi da quelli menzionati: 4

** : Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.

- Se viene usata una CPU QnA: 4
- Se viene usata una CPU del System Q con operandi interni a word (tranne file registri ZR): 6
costanti: 6
Operandi a bit con numero operando multiplo di 16,
la cui sigla di designazione è K8, e che non usa indicizzazione: 6
- Se viene usata una CPU del System Q con operandi diversi da quelli menzionati: 4

***: Il numero di passi di programma dipende dagli operandi usati.

Per il numero di passi esatto, fare riferimento al capitolo indicato.

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Somma logica	WOR	s, d	$(d) \vee (s) \longrightarrow (d)$		3	***5	7.1.3
	WORP						
	WOR	s1, s2, d1	$(s1) \vee (s2) \longrightarrow (d1)$		4	***7	7.1.3
	WORP						
	DOR	s, d	$(d+1, d) \vee (s+1, s) \longrightarrow (d+1, d)$		*4	***9	7.1.3
	DORP						
	DOR	s1, s2, d	$((s1)+1, s1) \vee ((s2)+1, s2) \longrightarrow (d+1, d)$		**4		7.1.3
	DORP						
	BKOR	s1, s2, n, d			5		7.1.4
	BKORP						

*: Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.

- Se viene usata una CPU QnA: 4
- Se viene usata una CPU a singolo processore del System Q: 3
- Se viene usata una CPU multiprocessore del System Q con operandi interni a word (tranne file registri ZR): 6
costanti: 6
Operandi a bit con numero operando multiplo di 16,
la cui sigla di designazione è K8, e che non usa indicizzazione: 6
- Se viene usata una CPU multiprocessore del System Q con operandi diversi da quelli menzionati: 4

** : Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.

- Se viene usata una CPU QnA: 4
- Se viene usata una CPU del System Q con operandi interni a word (tranne file registri ZR): 6
costanti: 6
Operandi a bit con numero operando multiplo di 16,
la cui sigla di designazione è K8, e che non usa indicizzazione: 6
- Se viene usata una CPU del System Q con operandi diversi da quelli menzionati: 4

***: Il numero di passi di programma dipende dagli operandi usati.

Per il numero di passi esatto, fare riferimento al capitolo indicato.

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
OR esclusivo logico	WXOR	s, d	$(d) \vee (s) \longrightarrow (d)$		3	***5	7.1.5
	WXORP						
	WXOR	s1, s2, d1	$(s1) \vee (s2) \longrightarrow (d1)$		4	***7	7.1.5
	WXORP						
	DXOR	s, d	$(d+1, d) \vee (s+1, s) \longrightarrow (d+1, d)$		*3	***9	7.1.5
	DXORP						
	DXOR	s1, s2, d	$((s1)+1, s1) \vee ((s2)+1, s2) \longrightarrow (d+1, d)$		**4		7.1.5
	DXORP						
	BKXOR	s1, s2, n, d	$\begin{matrix} (s1) & (s2) & (d) \\ \vdots & \vdots & \vdots \\ \vdots & \vee & \vdots \\ \vdots & \vdots & \vdots \end{matrix}$		5		7.1.6
	BKXORP						

*: Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.

- Se viene usata una CPU QnA: 4
- Se viene usata una CPU a singolo processore del System Q: 3
- Se viene usata una CPU multiprocessore del System Q con operandi interni a word (tranne file registri ZR): 6 costanti: 6
- Operandi a bit con numero operando multiplo di 16, la cui sigla di designazione è K8, e che non usa indicizzazione: 6
- Se viene usata una CPU multiprocessore del System Q con operandi diversi da quelli menzionati: 4

** : Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.

- Se viene usata una CPU QnA: 4
- Se viene usata una CPU del System Q con operandi interni a word (tranne file registri ZR): 6 costanti: 6
- Operandi a bit con numero operando multiplo di 16, la cui sigla di designazione è K8, e che non usa indicizzazione: 6
- Se viene usata una CPU del System Q con operandi diversi da quelli menzionati: 4

***: Il numero di passi di programma dipende dagli operandi usati.

Per il numero di passi esatto, fare riferimento al capitolo indicato.

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
NOR esclusivo logico	WNXR	s, d	$(d) \vee (s) \longrightarrow (d)$		3	***5	7.1.7
	WNXRP						
	WNXR	s1, s2, d1	$\overline{(s1) \vee (s2)} \overset{\diamond}{\rightarrow} (d1)$		4	***7	7.1.7
	WNXRP						
	DNXR	s, d	$\overline{(d+1, d) \vee (s+1, s)} \overset{\diamond}{\rightarrow} (d+1, d)$		*3	***9	7.1.7
	DNXRP						
	DNXR	s1, s2, d	$\overline{((s1)+1, s1) \vee ((s2)+1, s2)} \overset{\diamond}{\rightarrow} (d+1, d)$		**4		7.1.7
	DNXRP						
	BKXNR	s1, s2, n, d			5		7.1.8
BKXNRP							

*: Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.

- Se viene usata una CPU QnA: 4
- Se viene usata una CPU a singolo processore del System Q: 3
- Se viene usata una CPU multiprocessore del System Q con operandi interni a word (tranne file registri ZR): 6
costanti: 6
Operandi a bit con numero operando multiplo di 16,
la cui sigla di designazione è K8, e che non usa indicizzazione: 6
- Se viene usata una CPU multiprocessore del System Q con operandi diversi da quelli menzionati: 4

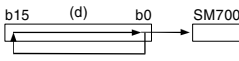
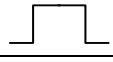
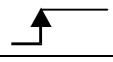
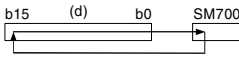
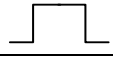
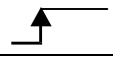
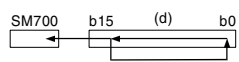
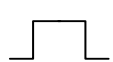

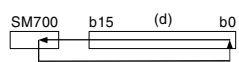
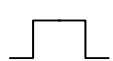

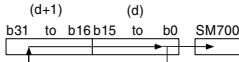
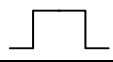

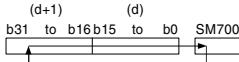
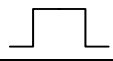

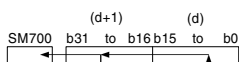
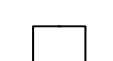

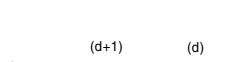

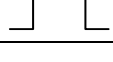
** : Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.

- Se viene usata una CPU QnA: 4
- Se viene usata una CPU del System Q con operandi interni a word (tranne file registri ZR): 6
costanti: 6
Operandi a bit con numero operando multiplo di 16,
la cui sigla di designazione è K8, e che non usa indicizzazione: 6
- Se viene usata una CPU del System Q con operandi diversi da quelli menzionati: 4

***: Il numero di passi di programma dipende dagli operandi usati.

Per il numero di passi esatto, fare riferimento al capitolo indicato.

2.5.2 Istruzioni di rotazione

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Rotazione dati verso destra (16-bit)	ROR	n, d			3	*3	7.2.1
	RORP		ruota di n bit verso destra				
	RRCR	n, d			3	*3	7.2.1
	RRCRP		ruota di n bit verso destra				
Rotazione dati verso sinistra (16-bit)	ROL	n, d			3	*3	7.2.2
	ROLP		ruota di n bit verso sinistra				
	RCL	n, d			3	*3	7.2.2
	RCLP		ruota di n bit verso sinistra				
Rotazione dati verso destra (32-bit)	DROR	n, d			3	*3	7.2.3
	DRORP		ruota di n bit verso destra				
	DRRCR	n, d			3	*3	7.2.3
	DRRCRP		ruota di n bit verso destra				
Rotazione dati verso sinistra (32-bit)	DROL	n, d			3	*3	7.2.4
	DROLP		ruota di n bit verso sinistra				
	DRCL	n, d			3	*3	7.2.4
	DRCLP		ruota di n bit verso sinistra				

*: Il numero di passi di programma dipende dagli operandi usati.
 Per il numero di passi esatto, fare riferimento al capitolo indicato.

2.5.3 Istruzioni di scorrimento

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Scorrimento di una word dati da 16 bit per n bit	SFR	n, d			3	*3	7.3.1
	SFRP						
	SFL	n, d			3	*3	7.3.1
	SFLP						
Scorre n operandi a bit di 1 bit	BSFR	n, d			3	*7	7.3.2
	BSFRP						
	BSFL	n, d			3	*7	7.3.2
	BSFLP						
Scorre n operandi a word di una cifra	DSFR	n, d			3	*7	7.3.3
	DSFRP						
	DSFL	n, d			3	*7	7.3.3
	DSFLP						

*: Il numero di passi di programma dipende dagli operandi usati.
Per il numero di passi esatto, fare riferimento al capitolo indicato.

2.5.4 Istruzioni di elaborazione a bit

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Set / reset di singoli bit	BSET	n, d			3	*3	7.4.1
	BSETP						
	BRST	n, d			3	*7	7.4.1
	BRSTP						
Test dello stato di singoli bit in word dati da 16/32 bit	TEST	s1, s2, d			4		7.4.2
	TESTP						
	DTEST	s1, s2, d			4		7.4.2
	DTESTP						
Reset di sezioni di bit in gruppo	BKRST	s, n			3		7.4.3
	BKRSTP						

*: Il numero di passi di programma dipende dagli operandi usati.
 Per il numero di passi esatto, fare riferimento al capitolo indicato.

2.5.5 Istruzioni per elaborazione dati

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Ricerca dati a 16-bit	SER	s1, s2, n (A) s1, s2, n, d (Q)			5	*9	7.5.1
	SERP						
	DSER	s1, s2, n (A) s1, s2, n, d (Q)			5	*9	7.5.1
	DSERP						
Controllo bit dati (16-/32-bit)	SUM	s (A) s, d (Q)			3	*3	7.5.2
	SUMP						
	DSUM	s (A) s, d (Q)			3	*3	7.5.2
	DSUMP						
Decodifica dati	DECO	s, d, n	Decodifica da 8 a 256 bit 		4	*9	7.5.3
	DECOP						
Codifica dati	ENCO	s, d, n	Codifica da 256 a 8 bit 		4	*9	7.5.4
	ENCOP						
Decodifica 7 segmenti	SEG	s, d			3	7	7.5.5
	SEGP						

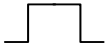

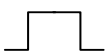
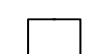

*: Il numero di passi di programma dipende dagli operandi usati.
Per il numero di passi esatto, fare riferimento al capitolo indicato.

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Separa/unisce word dati a 16 bit	DIS	s, n, d	L'istruzione DIS separa un valore a 16 bit in gruppi di 4 bit. Si devono specificare il valore da separare in s, il numero di gruppi da 4 bit in n, e il primo numero dell'operando di destinazione in d.		4	*9	7.5.6
	DISP						
	UNI	s, n, d	L'istruzione UNI separa ciascuno dei 4 bit più leggeri di fino a quattro valori a 16 bit e unisce il loro stato in un valore a 16 bit.		4	*9	7.5.7
	UNIP						
	NDIS	s1, s2, d	L'istruzione NDIS separa i dati negli operandi a partire da s1 in raggruppamenti a bit in cui il numero di bit è indicato da s2. I gruppi di bit separati vengono memorizzati in sequenza a partire dall'operando indicato da d.		4		7.5.8
	NDISP						
	NUNI	s1, s2, d	L'istruzione NUNI separa dei raggruppamenti di bit di dimensione specificata da s2 a partire dagli operandi specificati da s1, ed unisce questi gruppi di bit in un valore dati. I gruppi di bit separati vengono memorizzati in sequenza a partire dall'operando indicato da d.		4		7.5.8
	NUNIP						
	WTOB	s, n, d	Per questa istruzione si devono specificare il valore da separare in s, il numero di byte in n, e il primo numero dell'operando di destinazione in d.		4		7.5.9
	WTOBP						
	BTOW	s, n, d	Occorre specificare il numero iniziale del valore da unire in s, il numero di byte n, e l'operando di destinazione in d.		4		7.5.9
	BTOWP						
Ricerca valore massimo in dati a 16/32 bit	MAX	s, n, d	L'istruzione MAX ricerca il valore massimo in blocchi dati da 16 bit. Il numero di blocchi dati in cui effettuare la ricerca è indicato da n. Il valore più grande trovato da s a s+(n-1) viene memorizzato in d.		4		7.5.10
	MAXP						
	DMAX	s, n, d	L'istruzione DMAX ricerca il valore massimo in blocchi dati da 32 bit. Il numero di blocchi dati in cui effettuare la ricerca è indicato da n. Il valore più grande trovato da s a s+(n-1) viene memorizzato in d.		4		7.5.10
	DMAXP						

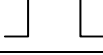


*: Il numero di passi di programma dipende dagli operandi usati.
Per il numero di passi esatto, fare riferimento al capitolo indicato.

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Ricerca valore minimo in dati a 16/32 bit	MIN	s, n, d	L'istruzione MIN ricerca il valore minimo in blocchi dati da 16 bit. Il numero di blocchi dati in cui effettuare la ricerca e indicato da n. Il valore più piccolo trovato da s a s+(n-1) viene memorizzato in d.		4		7.5.11
	MINP						
	DMIN	s, n, d	L'istruzione DMIN ricerca il valore minimo in blocchi dati da 32 bit. Il numero di blocchi dati in cui effettuare la ricerca e indicato da n. Il valore più piccolo trovato da s a s+(n-1) viene memorizzato in d.				
	DMINP						
Ordinamento dati 16/32-bit	SORT	s1, n, s2, d1, d2	L'istruzione SORT ordina i dati a 16 bit specificati da s1 in senso crescente o decrescente. Il numero di dati da ordinare è indicato in n.		6		7.5.12
	SORTP						
	DSORT		L'istruzione DSORT ordina i dati a 32 bit specificati da s1 in senso crescente o decrescente. Il numero di dati da ordinare è indicato in n.				
	DSORTP						
Calcolo del totale di blocchi dati BIN a 16/32 bit	WSUM	s, n, d	L'istruzione WSUM calcola il totale di blocchi dati a 16 bit nell'operando indicato da s. Il risultato viene memorizzato negli operandi indicati da d e d+1.		4		7.5.13
	WSUMP						
	DWSUM	s, n, d	L'istruzione DWSUM calcola il totale di blocchi dati a 32 bit nell'operando indicato da s. Il risultato viene memorizzato negli operandi indicati da d a d+3.		4		7.5.14
	DWSUMP						

2.5.6 Istruzioni per programmi strutturati

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Istruzioni di ripetizione	FOR	n	Il loop FOR/NEXT ripete singole sequenze di programma senza impostare una condizione di ingresso. La sequenza di programma posta fra i comandi FOR e NEXT viene ripetuta n volte.		2	*3	7.6.1
	NEXT				1	*1	
	BREAK	p, d	L'istruzione BREAK interrompe l'esecuzione di un loop FOR/NEXT e salta al puntatore/etichetta specificato da p.		3		7.6.2
	BREAKP						
Richiamo di sottoprogramma	CALL	p	L'istruzione CALL richiama una subroutine specificata da un puntatore Pxx in GX Developer o GX IEC Developer.		2	*3	7.6.3
	CALLP						
	RET		L'istruzione RET contrassegna la fine di una subroutine.		1	*1	7.6.4
	FCALL	p	Se la condizione di esecuzione dell'istruzione FCALL viene resettata, i contatti e bobine all'interno della subroutine specificata da p (puntatore/etichetta) vengono considerati come se l'istruzione corrispondente non fosse settata.		2		7.6.5
	FCALLP						
Richiamo di sottoprogramma fra file di programma	ECALL	nome file, p	L'istruzione ECALL richiama una subroutine specificata da un puntatore (etichetta) in un file di programma indicato da un nome file.		3		7.6.6
	ECALLP						
Richiamo di sottoprogramma fra file di programma	EFCALL	nome file, p	Se la condizione di esecuzione dell'istruzione EFCALL viene resettata, i contatti e bobine all'interno della subroutine specificata da p (puntatore/etichetta) vengono considerati come se l'istruzione corrispondente non fosse settata.		3		7.6.7
	EFCALLP						
Commutazione programma principale/sottoprogramma	CHG		Con la condizione di ingresso attiva, l'istruzione CHG abilita la commutazione fra i programmi MAIN e SUB.			1	7.6.8

*: Il numero di passi di programma dipende dagli operandi usati.
Per il numero di passi esatto, fare riferimento al capitolo indicato.

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Chiamata a programma microcomputer	SUB	n	Se la condizione di ingresso è attiva, l'istruzione SUB richiama il programma microcomputer posto all'indirizzo "n".		3		7.6.9
	SUBP						
Indicizzazione di interi segmenti	IX	s	Le istruzioni IX e IXEND eseguono una indicizzazione per quegli operandi compresi nella porzione di programma compresa fra IX e IXEND.		2		7.6.10
	IXEND				1		
Definizione di valori di indicizzazione di interi segmenti	IXDEV	p, d	Le istruzioni IXDEV e IXSET leggono gli indirizzi degli operandi nell'area di definizione offset e scrivono questi numeri di offset in una tabella di indicizzazione nell'operando indicato da d.		1		7.6.11
	IXSET						

2.5.7 Istruzioni per operazioni su tabelle dati

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi di programma		Riferimento
					Q	A	
Scrittura dati in una tabella dati	FIFW	s, d			3	*7	7.7.1
	FIFWP						
Legge primo dato inserito da una tabella dati	FIFR	s, d			3	*7	7.7.2
	FIFRP						
Legge ultimo dato inserito da una tabella dati	FPOP	s, d			3		7.7.3
	FPOPP						
Cancella blocchi dati definiti da una tabella dati	FDEL	s, n, d			4		7.7.4
	FDELP						
Inserisce blocchi dati definiti in una tabella dati	FINS						
	FINSP						








*: Il numero di passi di programma dipende dagli operandi usati.
 Per il numero di passi esatto, fare riferimento al capitolo indicato.

2.5.8 Istruzioni di accesso a buffer di memoria

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Lettura dati da un modulo funzione speciale	FROM	n1, n2, n3, d	L'istruzione FROM legge dati da una word (16 bit) dal buffer di memoria di un modulo funzione speciale.		5	*9	7.8.1
	FROMP						
	DFRO		L'istruzione DFRO legge dati da una doppia word (32 bit) dal buffer di memoria di un modulo funzione speciale.				
	DFROP						
Scrittura dati su un modulo funzione speciale	TO	s, n1, n2, n3	L'istruzione TO scrive dati da una word (16 bit) dalla memoria della CPU al buffer di memoria di un modulo funzione speciale.		5	*9	7.8.2
	TOP						
	DTO		L'istruzione DTO scrive dati a doppia word (32 bit) dalla memoria della CPU al buffer di memoria di un modulo funzione speciale.				
	DTOP						
Lettura dati da una stazione remota	FROM, PRC	n1, n2, n3, d (FROM(P)/DFRO(P)) s, d PRC	L'istruzione FROM/PRC legge dati da una word (16 bit) dal buffer di memoria di un modulo remoto.		*7/9	7.8.3	
	FROMP, PRC						
	DFRO, PRC		L'istruzione DFRO/PRC legge dati a doppia word (32 bit) dal buffer di memoria di un modulo remoto.				
	DFROP, PRC						
Scrittura dati su una stazione remota	TO, PRC	s, n1, n2, n3 (TO(P)/DTO(P)) s, d (PRC)	L'istruzione TO/PRC scrive dati da una word (16 bit) dalla memoria della CPU al buffer di memoria di un modulo remoto.		*7/9	7.8.4	
	TOP, PRC						
	DTO, PRC		L'istruzione DTO/PRC scrive dati a doppia word (32 bit) dalla memoria della CPU al buffer di memoria di un modulo remoto.				
	DTOP, PRC						

*: Il numero di passi di programma dipende dagli operandi usati.
Per il numero di passi esatto, fare riferimento al capitolo indicato.

2.5.9 Istruzioni di visualizzazione

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Emissione caratteri ASCII	PR	s, d	<p>SM701 (serie Q/ System Q) attivo (1): Emette una stringa da 16 caratteri ASCII verso un modulo di uscita. La stringa di caratteri, divisa in due metà da 8 caratteri, viene letta dall'area di indirizzi s ed emessa sulle uscite definite da d.</p> <p>SM701 (serie Q/ System Q) non attivo (0): Emette i dati della stringa di caratteri ASCII fino al codice carattere "00h" in formato esadecimale leggendo dall'area specificata da s verso le uscite indicate da d.</p>		3	*7	7.9.1
	PRC	s, d	<p>L'istruzione PRC emette il commento di un operando (in codice ASCII) verso un modulo di uscita.</p> <p>Se SM701 (serie Q/ System Q) è attivo (1), vengono emessi 16 caratteri;</p> <p>Se SM701 non è attivo (0), vengono emessi 32 caratteri.</p>		3	*7	7.9.2
Visualizzazione di caratteri ASCII e commenti	LED	s	L'istruzione LED legge dati ASCII (16 caratteri) da una area specificata e li visualizza su un display CPU adeguato.		2	*3	7.9.3
	LEDC	s (Q)	L'istruzione LEDC legge commenti (16 caratteri) da una area specificata e li visualizza su un display CPU adeguato.		2	*3	7.9.4
	LEDA	n	Queste istruzioni visualizzano una stringa di caratteri ASCII sul display a LED di una CPU appropriata.			*13	7.9.5
	LEDB						
Cancella display	LEDR		Cancella spie e segnalazioni di errore		1	*1	7.9.6





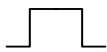
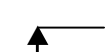

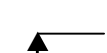




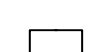
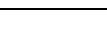
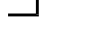

*: Il numero di passi di programma dipende dagli operandi usati.
Per il numero di passi esatto, fare riferimento al capitolo indicato.









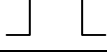


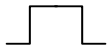



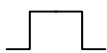

2.5.10 Istruzioni per debug e diagnosi guasti











Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Controllo guasto	CHKST		L'istruzione CHKST avvia l'esecuzione dell'istruzione CHK. Se la condizione di esecuzione per l'istruzione CHKST non è attiva (0), viene eseguito il passo di programma che segue l'istruzione CHK.		1		7.10.1
	CHK (Q)		L'istruzione CHK supporta le operazioni di ricerca guasti per i circuiti a contatti. Se si verifica un errore in questi circuiti, l'operando indicato da d1 si attiva e il codice di errore corrispondente viene memorizzato in d2.				
	CHK (A)	d1, d2	L'istruzione CHK supporta il controllo anomalie in un circuito a contatti con finecorsa. Se si verifica un errore in questi circuiti, l'operando indicato da d1 si attiva e il codice di errore corrispondente viene memorizzato in d2.		5	7.10.2	
	CHKCIR		L'istruzione CHKCIR genera circuiti di controllo errori per l'istruzione CHK ed avvia la sezione del programma con i circuiti di controllo errori generata.		1		7.10.3
	CHKEND		Istruzioni di fine per una sezione di programma comprendente circuiti di controllo generati.				
Memoria stato set/reset	SLT		L'istruzione SLT memorizza temporaneamente il dato dell'operando specificato. Il dato viene inserito nella memoria di stato a ritenuta e può essere controllata e visualizzata.		1	1	7.10.4
	SLTR		L'istruzione SLTR cancella i dati memorizzati temporaneamente nell'area degli stati ritenuti e ripristina (riabilita) l'istruzione SLT.				
Set/reset traccia campionamento	STRA		Attiva la traccia di campionamento		1	1	7.10.5
	STRAR		Disattiva la traccia di campionamento				
Esegue / attiva / disattiva la traccia di programma	PTRA		Attiva la traccia di programma		1		7.10.6
	PTRAR		Disattiva la traccia di programma				
	PTRAEXE		Esegue la traccia di programma.				
	PTRAEXEP						

2.5.11 Istruzioni di elaborazione stringhe di caratteri

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Conversione di dati binari a 16-/32-bit in valori decimali in codice ASCII	BINDA	s, d	L'istruzione BINDA converte un valore 16 bit specificato da s in un valore decimale a 5 cifre in codice ASCII e lo memorizza nell'operando indicato da d.		3		7.11.1
	BINDAP						
	DBINDA		L'istruzione DBINDA converte un valore 32 bit specificato da s in un valore decimale a 10 cifre in codice ASCII e lo memorizza nell'operando indicato da d.				
	DBINDAP						
Conversione di dati binari a 16-/32-bit in valore esadecimale in codice ASCII	BINHA	s, d	L'istruzione BINHA converte un valore 16 bit specificato da s in un valore esadecimale a 4 cifre in codice ASCII e lo memorizza negli operandi indicati da d.		3		7.11.2
	BINHAP						
	DBINHA		L'istruzione DBINHA converte un valore 32 bit specificato da s in un valore esadecimale a 8 cifre in codice ASCII e lo memorizza negli operandi indicati da d.				
	DBINHAP						
Conversione di un dato BCD a 4/8 cifre in codice ASCII	BCDDA	s, d	L'istruzione BCDDA converte il dato BCD a 4 cifre specificato da s in formato ASCII e lo memorizza negli operandi indicati da d.		3		7.11.3
	BCDDAP						
	DBCDDA		L'istruzione DBCDDA converte il dato BCD a 8 cifre specificato da s in formato ASCII e lo memorizza negli operandi indicati da d.				
	DBCDDAP						
Conversione di dati decimali ASCII a dati BIN a 16/32 bit	DABIN	s, d	L'istruzione DABIN converte il dato ASCII decimale a 5 cifre specificato da s in formato BIN a 16 bit e lo memorizza negli operandi indicati da d.		3		7.11.4
	DABINP						
	DDABIN		L'istruzione DDABIN converte il dato ASCII decimale a 10 cifre specificato da s in formato BIN a 32 bit e lo memorizza negli operandi indicati da d.				
	DDABINP						

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Conversione di dati esadecimali ASCII a dati binari a 16/32 bit	HABIN	s, d	L'istruzione HABIN converte il dato ASCII esadecimale a 4 cifre specificato da s in formato BIN a 16 bit e lo memorizza negli operandi indicati da d.		3		7.11.5
	HABINP						
	DHABIN		L'istruzione DHABIN converte il dato ASCII decimale a 8 cifre specificato da s in formato BIN a 32 bit e lo memorizza negli operandi indicati da d.				
	DHABINP						
Conversione di un dato ASCII decimale a 4/8 cifre in codice BCD	DABCD	s, d	L'istruzione DABCD converte il dato ASCII decimale specificato da s in formato BCD a 4 cifre e lo memorizza negli operandi indicati da d.		3		7.11.6
	DABCDP						
	DDABCD		L'istruzione DDABCD converte il dato ASCII decimale specificato da s in formato BCD a 8 cifre e lo memorizza negli operandi indicati da d.				
	DDABCDP						
Lettura di commenti	COMRD	s, d	L'istruzione COMRD legge il commento dall'operando indicato da s e lo memorizza in codice ASCII nell'area d.		3		7.11.7
	COMRDP						
Rilevamento lunghezza stringa di caratteri	LEN	s, d	L'istruzione di lunghezza rileva la lunghezza della stringa di caratteri specificata da s e memorizza il risultato nell'operando indicato da d.		3		7.11.8
	LENP						
Conversione di dati binari a 16-/32-bit in stringhe di caratteri	STR	s1, s2, d	L'istruzione STR aggiunge un punto decimale al valore binario a 16 bit nell'operando indicato da s2, in corrispondenza della cifra indicata dagli operandi s1, poi converte il dato in una stringa di caratteri e lo memorizza nell'area di operandi indicata da d.		4		7.11.9
	STRP						
	DSTR		L'istruzione DSTR aggiunge un punto decimale al valore binario a 32 bit nell'operando indicato da s2, in corrispondenza della cifra indicata dagli operandi s1, poi converte il dato in una stringa di caratteri e lo memorizza nell'area di operandi indicata da d.				
	DSTRP						

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Conversione di stringhe di caratteri a dati BIN a 16/32 bit	VAL	s, d1, d2	L'istruzione VAL converte la stringa di caratteri memorizzata nell'area s in un dato BIN a 16 bit. Il numero di cifre ed il valore binario vengono memorizzati in d1 e d2.		4		7.11.10
	VALP						
	DVAL		L'istruzione DVAL converte la stringa di caratteri memorizzata nell'area s in un dato BIN a 32 bit. Il numero di cifre ed il valore binario vengono memorizzati in d1 e d2.				
	DVALP						
Conversione di un dato in virgola mobile in una stringa di caratteri	ESTR	s1, s2, d	L'istruzione ESTR converte il dato in virgola mobile (numeri reali) indicato da s1 in una stringa di caratteri. Il formato dei dati nella stringa e specificato da s2. Il risultato è memorizzato in d.		4		7.11.11
	ESTRP						
Conversione di una stringa di caratteri in un dato in virgola mobile	EVAL	s, d	L'istruzione EVAL converte la stringa di caratteri indicata da s in un numero decimale in virgola mobile (numero reale). Il risultato è memorizzato in d.		3		7.11.12
	EVALP						
Conversione di dati 16-bit in codice ASCII (Q)	ASC	s, n, d	L'istruzione ASCII converte i dati a 16 bit memorizzati a partire da s in formato ASCII esadecimale, memorizzando il numero di caratteri specificati da n a partire da d.		4		7.11.13
	ASCP						
Conversione di stringhe di caratteri alfanumeriche in codice ASCII (A)	ASC	d	L'istruzione ASC converte stringhe di caratteri alfanumerici lunghe fino a 8 caratteri in codice ASCII. Il risultato è memorizzato a partire da d.			*13	7.11.14
Conversione di valore ASCII esadecimale in valori binari	HEX	s, n, d	L'istruzione HEX converte i caratteri ASCII esadecimali a partire da s in valori binari. Il numero di caratteri da convertire è indicato da n. Il risultato è memorizzato a partire da d.		4		7.11.15
	HEXP						
Estrazione di parti di stringhe di caratteri (parte destra di una stringa di caratteri)	RIGHT	s, n, d	L'istruzione RIGHT memorizza n caratteri del lato destro (parte finale) della stringa di caratteri indicata da s. I caratteri vengono memorizzati in d.		4		7.11.16
	RIGHTP						
Estrazione di parti di stringhe di caratteri (parte sinistra di una stringa di caratteri)	LEFT	s, n, d	L'istruzione LEFT memorizza n caratteri del lato sinistro (parte iniziale) della stringa di caratteri indicata da s. I caratteri vengono memorizzati in d.		4		7.11.16
	LEFTP						

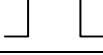


Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Selezione e spostamento di parti di una stringa di caratteri in una stringa di caratteri	MIDR	s1, s2, d	L'istruzione MIDR memorizza una parte specificata della stringa di caratteri indicata da s. Il primo carattere della parte da memorizzare è specificato in s2.		4		7.11.17
	MIDRP						
	MIDW	s1, s2, d	L'istruzione MIDW memorizza una parte di lunghezza specificata della stringa di caratteri indicata da s1 nell'area specificata da d. Il primo indirizzo dell'area di memorizzazione in d è specificato da s2.				
	MIDWP						
Ricerca di stringhe di caratteri	INSTR	s1, s2, n, d	L'istruzione INSTR ricerca la stringa di caratteri specificata da s1 all'interno della stringa specificata da s2. La ricerca inizia dal carattere specificato da n.		5		7.11.18
	INSTRP						
Conversione dati in virgola mobile con rappresentazione BCD	EMOD	s1, s2, d1	L'istruzione EMOD calcola il formato BCD a partire dal numero in virgola mobile (numero reale) indicato da s1, considerando la posizione del punto decimale specificata da s2, partendo da destra. Il risultato è memorizzato in d1.		4		7.11.19
	EMODP						
Conversione di dati BCD con formato decimale in virgola mobile	EREXP	s1, s2, d1	L'istruzione EREXP calcola il formato decimale del numero in virgola mobile (numero reale) partendo dal dato virgola mobile in formato BCD indicato da s1, considerando la posizione decimale indicata da s2. Il risultato è memorizzato in d1.		3		7.11.20
	EREXPP						

*: Il numero di passi di programma dipende dagli operandi usati.
Per il numero di passi esatto, fare riferimento al capitolo indicato.

2.5.12 Istruzioni per funzioni speciali

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Calcolo del seno	SIN	s, d	$SIN(s+1, s) \rightarrow (d+1, d)$		3		7.12.1
	SINP						
Calcolo del coseno	COS	s, d	$COS(s+1, s) \rightarrow (d+1, d)$		3		7.12.2
	COSP						
Calcolo della tangente	TAN	s, d	$TAN(s+1, s) \rightarrow (d+1, d)$		3		7.12.3
	TANP						
Calcolo arco-seno	ASIN	s, d	$SIN^{-1}(s+1, s) \rightarrow (d+1, d)$		3		7.12.4
	ASINP						
Calcolo arco-coseno	ACOS	s, d	$COS^{-1}(s+1, s) \rightarrow (d+1, d)$		3		7.12.5
	ACOSP						
Calcolo arco-tangente	ATAN	s, d	$TAN^{-1}(s+1, s) \rightarrow (d+1, d)$		3		7.12.6
	ATANP						
Conversione da gradi a radianti	RAD	s, d	$(s+1, s) \rightarrow (d+1, d)$ Conversione da gradi a radianti		3		7.12.7
	RADP						
Conversione da radianti a gradi	DEG	s, d	$(s+1, s) \rightarrow (d+1, d)$ Conversione da radianti a gradi		3		7.12.8
	DEGP						
Calcolo radice quadrata	SQR	s, d	$\sqrt{(s+1, s)} \rightarrow (d+1, d)$		3		7.12.9
	SQRP						

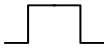

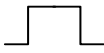

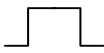

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Valore in virgola mobile come esponente di e	EXP	s, d	$e^{(s+1, s)} \rightarrow (d+1, d)$		3		7.12.10
	EXPP						
Calcolo logaritmo (naturale)	LOG	s, d	$\text{LOG } e^{(s+1, s)} \rightarrow (d+1, d)$		3		7.12.11
	LOGP						
Valore random	RND	d	Memorizza in d il valore random generato.		3		7.12.12
	RNDP						
Aggiorna valori random	SRND	s	Aggiorna la serie di valori random memorizzati in s.		3		7.12.13
	SRNDP						
Calcolo radice quadrata da dato BCD a 4-cifre	BSQR	s, d	$\sqrt{(s)} \rightarrow (d)+0$ Integer +1 Decimal place		3		7.12.13
	BSQRP						
Calcolo radice quadrata da dato BCD a 8-cifre	BDSQR	s, d	$\sqrt{(s+1, s)} \rightarrow (d)+0$ Integer +1 Decimal place		3		7.12.14
	BDSQRP						
Calcolo seno da dato BCD	BSIN	s, d	$\sin(s) \rightarrow (d)+0$ Sign character +1 Integer +2 Decimal place		3		7.12.14
	BSINP						
Calcolo coseno da dato BCD	BCOS	s, d	$\cos(s) \rightarrow (d)+0$ Sign character +1 Integer +2 Decimal place		3		7.12.15
	BCOSP						
Calcolo tangente da dato BCD	BTAN	s, d	$\tan(s) \rightarrow (d)+0$ Sign character +1 Integer +2 Decimal place		3		7.12.16
	BTANP						

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Calcolo arco-seno da dato BCD	BASIN	s, d	$\sin^{-1}(s) \rightarrow$ (d) +0 +1 +2		3		7.12.17
	BASINP						
Calcolo arco-coseno da dato BCD	BACOS	s, d	$\cos^{-1}(s) \rightarrow$ (d) +0 +1 +2		3		7.12.18
	BACOSP						
Calcolo arco-tangente da dato BCD	BATAN	s, d	$\tan^{-1}(s) \rightarrow$ (d) +0 +1 +2		3		7.12.19
	BATANP						

2.5.13 Istruzioni per controllo dati

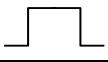

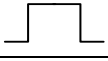
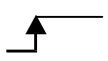
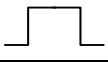

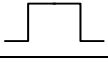
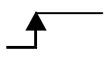

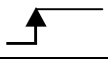

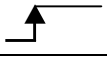
Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Controllo limiti inferiore e superiore per dati BIN a 16/32 bit	LIMIT	s1, s2, s3, d	Se $(s3) < (s1)$ il valore in s1 viene memorizzato in d.		5		7.13.1
	LIMITP		Se $(s1) \leq (s3) \leq (s2)$ il valore in s3 viene memorizzato in d. Se $(s2) < (s3)$ il valore in s2 viene memorizzato in d.				
	DLIMIT	s1, s2, s3, d	Se $((s3)+1, s3) < ((s1)+1, s1)$ il valore in $((s1)+1, s1)$ viene memorizzato in $(d+1, d)$.				
	DLIMITP		Se $((s1)+1, s1) \leq ((s3)+1, s3) < ((s2)+1, s2)$ il valore in $((s3)+1, s3)$ viene memorizzato in $(d+1, d)$. Se $((s2)+1, s2) < ((s3)+1, s3) < ((s2)+1, s2)$ il valore in $((s2)+1, s2)$ viene memorizzato in $(d+1, d)$.				
Controllo zona morta per dati BIN a 16/32 bit	BAND	s1, s2, s3, d	Se $(s1) \leq (s3) \leq (s2)$ $0 \rightarrow (d)$		5		7.13.2
	BANDP		Se $(s3) < (s1)$ $(s3) \rightarrow (s1) \rightarrow (d)$ Se $(s2) < (s3)$ $(s3) \rightarrow (s2) \rightarrow (d)$				
	DBAND	s1, s2, s3, d	Se $((s1)+1, s1) \leq ((s3)+1, s3)$ $\leq ((s2)+1, s2)$ $0 \rightarrow (d+1, d)$				
	DBANDP		Se $((s3)+1, s3) < ((s1)+1, s1)$ $((s3)+1, s3) - ((s1)+1, s1)$ $\rightarrow (d+1, d)$ Se $((s2)+1, s2) < ((s3)+1, s3)$ $((s3)+1, s3) - ((s2)+1, s2)$ $\rightarrow (d+1, d)$				
Controllo banda per dati BIN a 16/32 bit	ZONE	s1, s2, s3, d	Se $s3=0$: $0 \rightarrow (d)$ Se $s3>0$: $s3 + s2 \rightarrow (d)$		5		7.13.3
	ZONEP		Se $s3<0$: $s3 \rightarrow s1 \rightarrow (d)$				
	DZONE	s1, s2, s3, d	Se $((s3)+1, s3)=0$ $0 \rightarrow (d+1, d)$ Se $((s3)+1, s3)>0$ $((s3)+1, s3) + ((s2)+1, s2)$ $\rightarrow (d+1, d)$				
	DZONEP		Se $((s3)+1, s3)<0$ $((s3)+1, s3) + ((s1)+1, s1)$ $\rightarrow (d+1, d)$				

2.5.14 Istruzioni di commutazione File register

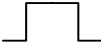
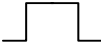
Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Istruzione di commutazione per blocchi di registri file	RSET	s	L'istruzione RSET commuta dal blocco di file register usati da un programma, a un blocco di file register il cui numero è specificato da s.		2		7.14.1
	RSETP						
Istruzione di commutazione per file di registri file	QDRSET	s	L'istruzione QDRSET commuta dal file di file register usati da un programma, a un file di file register il cui numero è specificato da s.		* 2 + n		7.14.2
	QDRSETP						
Istruzione di commutazione per file di commento	QCDSET	s	L'istruzione QCDSET commuta dal file di commento file usato da un programma, a un file di commento il cui numero è specificato da s.		* 2 + n		7.14.3
	QCDSETP						

*: $n = (\text{numero di caratteri del nome programma})/2 = \text{numero di passi aggiuntivi}$ (frazioni decimali vengono arrotondate)

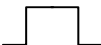
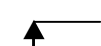
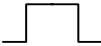
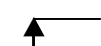
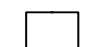
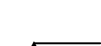
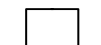
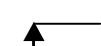
2.5.15 Istruzioni per l'orologio interno

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento																		
					Q	A																			
Lettura dati orologio	DATERD	d	QnA CPU clock → <table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td>d+0</td><td>Year</td></tr> <tr><td>d+1</td><td>Month</td></tr> <tr><td>d+2</td><td>Day</td></tr> <tr><td>d+3</td><td>Hour</td></tr> <tr><td>d+4</td><td>Minute</td></tr> <tr><td>d+5</td><td>Second</td></tr> <tr><td>d+6</td><td>Weekday</td></tr> </table>	d+0	Year	d+1	Month	d+2	Day	d+3	Hour	d+4	Minute	d+5	Second	d+6	Weekday	 	2		7.15.1				
	d+0			Year																					
d+1	Month																								
d+2	Day																								
d+3	Hour																								
d+4	Minute																								
d+5	Second																								
d+6	Weekday																								
DATERDP																									
Scrittura dati orologio	DATEWR	s	<table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td>s+0</td><td>Year</td></tr> <tr><td>s+1</td><td>Month</td></tr> <tr><td>s+2</td><td>Day</td></tr> <tr><td>s+3</td><td>Hour</td></tr> <tr><td>s+4</td><td>Minute</td></tr> <tr><td>s+5</td><td>Second</td></tr> <tr><td>s+6</td><td>Weekday</td></tr> </table> → QnA CPU clock	s+0	Year	s+1	Month	s+2	Day	s+3	Hour	s+4	Minute	s+5	Second	s+6	Weekday	 	2		7.15.2				
	s+0			Year																					
s+1	Month																								
s+2	Day																								
s+3	Hour																								
s+4	Minute																								
s+5	Second																								
s+6	Weekday																								
DATEWRP																									
Somma dati orologio	DATE+	s1, s2, d	<table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td>s1</td><td>Hour</td></tr> <tr><td></td><td>Minute</td></tr> <tr><td></td><td>Second</td></tr> </table> + <table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td>s2</td><td>Hour</td></tr> <tr><td></td><td>Minute</td></tr> <tr><td></td><td>Second</td></tr> </table> → <table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td>d</td><td>Hour</td></tr> <tr><td></td><td>Minute</td></tr> <tr><td></td><td>Second</td></tr> </table>	s1	Hour		Minute		Second	s2	Hour		Minute		Second	d	Hour		Minute		Second	 	4		7.15.3
	s1			Hour																					
	Minute																								
	Second																								
s2	Hour																								
	Minute																								
	Second																								
d	Hour																								
	Minute																								
	Second																								
DATE+P																									
Sottrazione dati orologio	DATE-	s1, s2, d	<table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td>s1</td><td>Hour</td></tr> <tr><td></td><td>Minute</td></tr> <tr><td></td><td>Second</td></tr> </table> - <table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td>s2</td><td>Hour</td></tr> <tr><td></td><td>Minute</td></tr> <tr><td></td><td>Second</td></tr> </table> → <table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td>d</td><td>Hour</td></tr> <tr><td></td><td>Minute</td></tr> <tr><td></td><td>Second</td></tr> </table>	s1	Hour		Minute		Second	s2	Hour		Minute		Second	d	Hour		Minute		Second	 	4		7.15.4
	s1			Hour																					
	Minute																								
	Second																								
s2	Hour																								
	Minute																								
	Second																								
d	Hour																								
	Minute																								
	Second																								
DATE-P																									
Modifica formato orologio da hh:mm:ss a secondi	SECOND	s, d	<table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td>s</td><td>Hour</td></tr> <tr><td></td><td>Minute</td></tr> <tr><td></td><td>Second</td></tr> </table> → <table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td>d</td><td>Second</td></tr> </table>	s	Hour		Minute		Second	d	Second	 	3		7.15.5										
	s			Hour																					
	Minute																								
	Second																								
d	Second																								
SECONDP																									
Modifica formato orologio da secondi a hh:mm:ss	HOUR	s, d	<table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td>s</td><td>Second</td></tr> </table> → <table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td>d</td><td>Hour</td></tr> <tr><td></td><td>Minute</td></tr> <tr><td></td><td>Second</td></tr> </table>	s	Second	d	Hour		Minute		Second	 													
	s			Second																					
d	Hour																								
	Minute																								
	Second																								
HOURP																									

2.5.16 Istruzioni per dispositivi periferici

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Invio messaggi a dispositivi periferici	MSG	s	L'istruzione MSG invia una stringa di caratteri memorizzata nell'operando indicato da s, verso il dispositivo periferico specificato in modo terminale.		2		7.16.1
Ingresso dati tastiera da dispositivo periferico	PKEY	d	I dati di ingresso della tastiera (caratteri) vengono letti dal dispositivo periferico specificato in modo terminale e scritti in formato ASCII sugli operandi indicati da d.		2		7.16.2





2.5.17 Istruzioni di programma

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Commutazione programmi in modo stand-by	PSTOP	s	L'istruzione PSTOP imposta il programma specificato dall'operando s nel modo stand-by.		2		7.17.1
	PSTOPP						
Commutazione programmi in modo stand-by e reset di uscite	POFF	s	L'istruzione POFF dispone il programma specificato dall'operando s in modo stand-by e azzerale uscite comandate dal programma.		2		7.17.2
	POFFP						
Commutazione programmi in modo esecuzione scansione	PSCAN	s	L'istruzione PSCAN imposta il programma specificato dell'operando indicato da s nel modo esecuzione scansione. In questo modo il programma viene eseguito solo una volta per ogni scansione.		3		7.17.3
	PSCANP						
Commutazione programmi nel modo esecuzione lenta	PLOW	s	L'istruzione PLOW imposta il programma specificato dell'operando indicato da s nel modo esecuzione lenta.		3		7.17.4
	PLOWP						

*: $n = (\text{numero di caratteri del nome programma})/2 = \text{numero di passi aggiuntivi (frazioni decimali vengono arrotondate)}$

2.5.18 Altre istruzioni

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Cancella timer di watchdog	WDT		L'istruzione WDT cancella il timer di watchdog (WDT) durante l'esecuzione di un programma di sequenza.		1	1	7.18.1
	WDTP						
Set e reset del riporto	STC		Il flag di riporto memorizza il riporto (0 o 1) delle operazioni di scorrimento e rotazione.			1	7.18.2
	CLC		Il flag di riporto viene cancellato dopo l'esecuzione di una istruzione CLC.				
Imposta il numero delle scansioni in esecuzione	DUTY	n1, n2, d	<p>SM420 to SM424, SM 430 to SM434</p>		4	7	7.18.3
Lettura diretta di un byte	ZRRDB	n, d			3		7.18.4
	ZRRDBP						
Scrittura diretta di un byte	ZRWRB	n, s			3		7.18.5
	ZRWRBP						
Memorizzazione di un indirizzo indiretto	ADRSET	s, d	Memorizza l'indirizzo indiretto dell'operando designato da s su d e d+1. Questo indirizzo viene usato durante l'esecuzione di una lettura indiretta dell'operando.		3		7.18.6
	ADRSETP						
Ingresso tasto numerico da tastiera	KEY	s, n, d1, d2	L'istruzione KEY gestisce l'acquisizione di 8 caratteri ASCII provenienti dalla tastiera sugli ingressi specificati da s (X). I valori inseriti negli ingressi sono codificati in formato esadecimale e memorizzati negli operandi indicati da d1.		5		7.18.7
Salvataggio in blocco del contenuto dei registri indice	ZPUSH	d	L'istruzione ZPUSH salva il contenuto dei registri indice da Z0 a Z15 in d		3		7.18.8
	ZPUSHP						





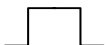
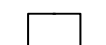

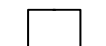
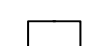
Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Ripristino in blocco del contenuto dei registri	ZPOP	d	L'istruzione ZPOP ripristina da d il contenuto dei registri indice da Z0 a Z15.		3		7.18.9
	ZPOP						
Scrittura in blocco di dati sul registro EEPROM	EROMWR	s, n, d1,d2	L'istruzione EROMWR scrive il numero di word dati specificato da n, contenute nell'operando indicato da s, nel file registri EEPROM specificato da d1.		6		7.18.10
	EROMWR						

2.6 Istruzioni per il collegamento dei dati

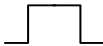

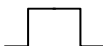

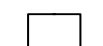


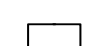
2.6.1 Istruzioni rinfresco rete

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Istruzioni per rinfresco rete	ZCOM	Jn	Rinfresco dati su moduli di rete		6		8.5.1
		Un					

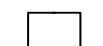
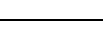
2.6.2 Istruzioni dedicate per il collegamento dei dati

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Istruzioni dedicate per il collegamento dei dati	READ	Jn, s1, s2, d1, d2	Lettura dati operando a word da una diversa stazione		11		8.6.1
		Un, s1, s2, d1, d2					
	SREAD	Jn, s1, s2, d1, d2, d3	Scrittura dati operando a word su una diversa stazione		13		8.6.2
		Un, s1, s2, d1, d2, d3					
	WRITE	Jn, s1, s2, d1, d2	Scrittura dati operando a word su una diversa stazione		12		8.6.3
		Un, s1, s2, d1, d2					
	SWRITE	Jn, s1, s2, d1, d2, d3	Scrittura dati operando a word su una diversa stazione		13		8.6.4
		Un, s1, s2, d1, d2, d3					
	SEND	Jn, s1, s2, d	Invio dati ad altre stazioni		10		8.6.5
		Un, s1, s2, d					
	RECV	Jn, s, d1, d2	Ricezione dati inviati da altre stazioni o ricezione dei dati inviati con l'istruzione SEND		9		8.6.6
		Un, s, d1, d2					
REQ	Jn, s1, s2, d1, d2	Richiesta dati da altre stazioni		10		8.6.7	
	Un, s1, s2, d1, d2						
ZNFR	Jn, s1, s2, d	Lettura dati da moduli funzione speciali in stazioni I/O remote		9		8.6.8	
	Un, s1, s2, d						
ZNTO	Jn, s1, s2, d	Scrittura dati su moduli funzione speciali in stazioni I/O remote		9		8.6.9	
	Un, s1, s2, d						

2.6.3 Istruzioni per il collegamento dei dati compatibili con la serie A


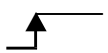
Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Istruzioni per il collegamento dei dati compatibili con la serie A	J.ZNRD	Jn, n1, s, n2, d1, d2	Legge dati QnA da stazioni oggetto in reti oggetto		12		8.7.1
	JP.ZNRD		Lettura dati da stazioni locali				
	J.ZNWR	Jn, n1, s, n2, d1, d2	Scrive dati QnA su stazioni oggetto in reti oggetto		12		8.7.2
	JP.ZNWR		Scrive dati su stazioni locali				
	LRDP	s, n1, n2, d	Solo serie A: Lettura dati da stazioni locali			11	8.7.3
	LWTP	Jn, s, d1, d2	Solo serie A: Scrive dati su stazioni locali			11	8.7.4
	RFRP	n1, n2, n3, d	Lettura dati da un modulo funzione speciale in una stazione remota		9	11	8.7.5
	G.RFRP	Un, n1, n2, d1, d2					
	RTOP	s, n1, n2, n3	Scrittura dati su un modulo funzione speciale in una stazione remota		9	11	8.7.6
	G.RTOP	Un, n1, s, n2, d1					

2.6.4 Lettura/scrittura dati di instradamento

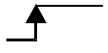
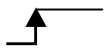
Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Lettura/scrittura dati di instradamento	Z.RTREAD	n, d	L'istruzione RTREAD legge i dati di instradamento dalla rete di destinazione indicata da n. I dati di instradamento sono memorizzati nei parametri di instradamento. Le informazioni lette sono memorizzate a partire da d.		7		8.8.1
	ZP.RTREAD						
	Z.RTWRITE	s, n	L'istruzione RTWRITE scrive le informazioni di instradamento sulla rete di destinazione specificata da n. Le informazioni lette sono memorizzate a partire da s.		8		8.8.2
	ZP.RTWRITE						

2.7 Istruzioni per CPU System Q



2.7.1 Lettura informazioni dei moduli

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Lettura informazioni modulo	UNIRD	n1, d, n2	Legge le informazioni del modulo memorizzate nell'area che parte dal numero di I/O indicato da n1 e le memorizza nell'area che inizia dall'operando indicato da d. Il numero di punti è indicato da n2.		4		9.1.1
	UNIRD						




2.7.2 Istruzioni per debug e diagnosi guasti

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Insieme di trace	TRACE		Memorizza i dati di traccia di un dispositivo periferico in un file sulla memory card con il numero indicato, se SM800, SM801 e SM802 sono ON.		1		9.2.1
Reset traccia	TRACER		Cancela l'insieme di dati impostati dall'istruzione TRACE		1		9.2.1

2.7.3 Scrittura e lettura da file

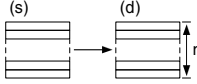
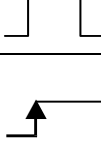
Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Scrittura dati nel file designato	SP.FWRITE	u0, s0, d0, s1, s2, d1	Scrittura dati nel file designato		11		9.3.1
Letture dati dal file designato	SP.FREAD	u0, s0, d0, s1, d1, d2	Legge dati da un file designato		11		9.3.2

2.7.4 Istruzioni di programma

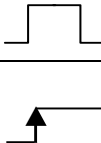
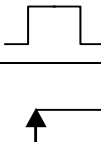
Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Caricamento programma da memoria	PLOADP	s, d	Trasferisce il programma memorizzato su una memory card o sulla memoria standard (diversa dal drive 0) nel drive 0 e pone il programma nello stato stand-by.		3		9.4.1
Scarica programma dalla memoria programmi	PUNLOADP	s, d	Cancela il programma in standby memorizzato nella memoria standard. (drive 0)		3		9.4.2
Carico e scarico	PSWAPP	s1, s2, d	Cancela il programma in stand-by indicato da s1 presente nella memoria standard (drive 0). Successivamente il programma (s2) contenuto in una memory card (diversa dal drive 0) viene trasferito nel drive 0 e posto nello stato stand-by.		4		9.4.3

Le istruzioni sono disponibili solo con GX Developer. GX IEC Developer non supporta il file system.

2.7.5 Istruzioni per trasferimento dati

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Trasferimento ad alta velocità di blocchi del file registri	RBMOV	s, d, n			4		9.5.1
	RBMOVP	s, d, n					

2.7.6 Istruzioni per scambio dati in un sistema multi-CPU

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Scrive nella memoria condivisa della CPU	S.TO	s1, s2, s3, s4, d	Scrive dati dal dispositivo di memoria alla memoria condivisa della stessa CPU (che sta eseguendo l'istruzione S.TO).		5		9.6.1
	S.TOP						
Lettura dalla memoria condivisa di una diversa CPU	FROM	n1, n2, n3, d	Legge dati dalla memoria condivisa di un'altra CPU e memorizza i dati nel dispositivo di memoria della CPU che esegue l'istruzione FROM.		5		9.6.2
	FROMP						
Rinfresco automatico della memoria condivisa della CPU	COM	—	Esegue il rinfresco automatico del modulo funzione intelligente, l'elaborazione dei dati generali, ed il rinfresco automatico della memoria condivisa fra le CPU.		1		6.7.3

2.8 Istruzioni dedicate per Q4ARCPU

2.8.1 Istruzioni per impostazione del modo

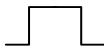


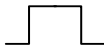

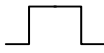

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Impostazione modalità di avviamento	SGMODE	s1, s2	Scelta fra avviamento a freddo o avviamento a caldo				10.1.1
Impostazione del modo operativo quando la CPU viene commutata	CGMODE	s	Scelta dell'azione da compiere durante la commutazione da sistema attivo a sistema in stand-by.				10.1.2

2.8.2 Istruzioni per trasferimento dati

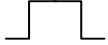



Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Istruzioni per tracciamento dati	TRUCK	s	Trasferimento del contenuto del dispositivo di memoria dalla CPU del sistema attivo alla CPU del sistema in stand-by				10.2.1
Rinfresco in blocco buffer di memoria	SPREF	s	Trasferimento in blocco di dati dalla memoria buffer dei moduli funzione speciali				10.2.2

2.9 Istruzioni per moduli funzione speciali


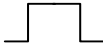



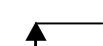
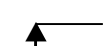

2.9.1 Istruzioni per moduli di comunicazione seriale

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Letture dati da un modulo di comunicazione seriale	BUFRCVS	„Un“, n1, d1	Letture dei dati ricevuti da un modulo di comunicazione seriale QJ71C24 in un programma di interruzione.				11.1.1
Letture di trame utente registrate	GETE	Un, s1, s2, d	Le trame utente registrate vengono lette da un modulo di comunicazione seriale				11.1.2
	GETEP						
Registrazione o cancellazione di trame utente	PUTE	Un, s1, s2, d	Le trame utente vengono registrate o cancellate in un modulo di comunicazione seriale				11.1.3
	PUTEP						
Trasmissione dati	PRR	Un, s, d	Invio dati tramite modulo di comunicazione seriale usando trame utente				11.1.4
	PRRP						

2.9.2 Istruzioni per modulo interfaccia PROFIBUS/DP

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Lettura dati	BBLKRD	„Un“, n1, n2, d	I dati vengono letti dal buffer di memoria di un modulo di interfaccia PROFIBUS/DP e memorizzati nella CPU del PLC				11.2.1
	BBLKRD						
Scrittura dati	BBLKWR	„Un“, n1, n2, s	I dati memorizzati nella CPU del PC vengono scritti nel buffer di memoria di un modulo di interfaccia PROFIBUS/DP				11.2.2
	BBLKWR						

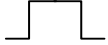

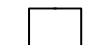

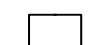

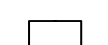

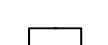

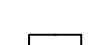

2.9.3 Istruzioni per moduli interfaccia ETHERNET

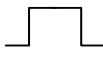
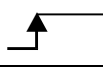
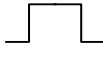
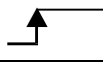

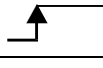
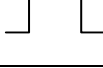


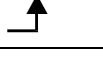
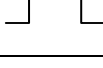
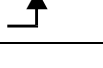

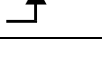
Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Lettura da buffer fisso	BUFRCV	„Un“, s1, s2, d1, d2	I dati ricevuti durante una comunicazione con buffer fisso vengono letti dal modulo di interfaccia ETHERNET				11.3.1
	BUFRCSV	„Un“, s1, d1					11.3.2
Scrittura su buffer fisso	BUFSND	„Un“, s1, s2, s3, d1	I dati memorizzati nella CPU del PLC vengono spostati nel buffer fisso di un modulo interfaccia ETHERNET				11.3.3
Apri connessione	OPEN	„Un“, s1, s2, d1	Esegue l'apertura di una connessione				11.3.4
Chiudi connessione	CLOSE	„Un“, s1, s2, d1	Esegue la chiusura di una connessione				11.3.5
Cancella errori	ERRCLR	„Un“, s1, d1	I codici di errore memorizzati nel buffer di memoria del modulo interfaccia ETHERNET vengono cancellati e il LED "ERR." viene spento.				11.3.6
Lettura codice di errore	ERRRD	„Un“, s1, d1	I codici di errore memorizzati nel buffer di memoria del modulo interfaccia ETHERNET vengono letti dalla CPU del PLC				11.3.7
Reinizializzazione	UINI	„Un“, s1, d1	Esecuzione reinizializzazione di un modulo interfaccia ETHERNET				11.3.8


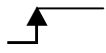
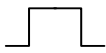

2.9.4 Istruzioni per MELSECNET/10

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Impostazione accoppiamenti	PAIRSET	Jn, s1	Impostazione stazioni per reti duplex				11.4.1

2.9.5 Istruzioni per CC-Link

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Impostazione parametri (serie A)	RLPA	n, d1, d2	Trasferimento impostazione parametri a stazione master di CC-Link		—	23	11.5.1
	RLPA_P						
Impostazione parametri (System Q)	RLPASET	Un, s1 a s5, d1				—	11.5.2
	RLPASET_P						
Impostazione dei parametri di rinfresco automatico (serie A)	RRPA	n, d	Impostazione dei dispositivi su cui viene eseguito il rinfresco automatico fra il modulo master/locale e la CPU del PLC		—	20	11.5.3
	RRPA_P						
Lettura dal buffer di memoria o dal dispositivo di memoria di una CPU (serie A)	RIRD	n1, n2, d1, d2	I dati vengono letti dal buffer di memoria del modulo CC-Link di una diversa stazione o dal dispositivo di memoria della CPU di questa stazione		—	26	11.5.4
	RIRD_P						
Lettura dal buffer di memoria o dal dispositivo di memoria di una CPU (serie QnA e System Q)	RIRD	Un, s, d1, d2			8	—	11.5.5
	RIRD_P						
Scrittura nel buffer di memoria o nel dispositivo di memoria di una CPU (serie A)	RIWT	n1, n2, d1, d2	I dati vengono scritti nel buffer di memoria del modulo CC-Link di una diversa stazione o nel dispositivo di memoria della CPU di questa stazione		—	26	11.5.6
	RIWT_P						

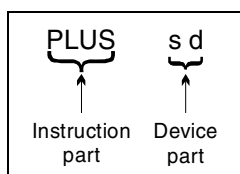
Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento
					Q	A	
Scrittura nel buffer di memoria o nel dispositivo di memoria di una CPU (serie QnA, System Q)	RIWT	Un, s, d1, d2	I dati vengono scritti nel buffer di memoria del modulo CC-Link di una diversa stazione o nel dispositivo di memoria della CPU di questa stazione		8	—	11.5.7
	RIWT_P						
Letture da una stazione intelligente (serie A)	RICV	n1, n2, d1, d2, d3	I dati vengono letti con handshake dal buffer di memoria di una stazione intelligente collegata al CC-Link		—	29	11.5.8
	RICV_P						
Letture da una stazione intelligente (serie QnA, System Q)	RICV	Un, s1, s2, d1, d2	I dati vengono letti con handshake dal buffer di memoria di una stazione intelligente collegata al CC-Link		10	—	11.5.9
	RICV_P						
Scrittura su una stazione intelligente (serie A)	RISEND	n1, n2, d1, d2, d3	I dati vengono scritti con handshake dal buffer di memoria di una stazione intelligente collegata al CC-Link		—	29	11.5.10
	RISEND_P						
Scrittura su una stazione intelligente (serie QnA, System Q)	RISEND	Un, s1, s2, d1, d2	I dati vengono scritti con handshake dal buffer di memoria di una stazione intelligente collegata al CC-Link		10	—	11.5.11
	RISEND_P						
Scrittura nel buffer di memoria ad aggiornamento automatico (serie A)	RITO	n1, n2, n3, n4, d1	I dati vengono trasferiti dal dispositivo di memoria della CPU del PLC nel buffer di memoria ad aggiornamento automatico della stazione master. Questi dati sono poi trasferiti ad un'altra stazione collegata a CC-Link.		—	29	11.5.12
	RITO_P						
Scrittura nel buffer di memoria ad aggiornamento automatico (serie QnA, System Q)	RITO	Un, n1, n2, n3, d	I dati vengono trasferiti dal dispositivo di memoria della CPU del PLC nel buffer di memoria ad aggiornamento automatico della stazione master. Questi dati sono poi trasferiti ad un'altra stazione collegata a CC-Link.		9	—	11.5.13
	RITO_P						

Categoria	Istruzione	Variabili	Significato	Condizione di esecuzione	Numero di passi		Riferimento	
					Q	A		
Lettura da un buffer di memoria ad aggiornamento automatico (serie A)	RIFR	n1, n2, n3, n4, d1	I dati trasmessi da un'altra stazione nel buffer di memoria ad aggiornamento automatico della stazione master vengono trasferiti nella memoria della CPU del PLC		—	29	11.5.14	
	RIFR_P							
Lettura da un buffer di memoria ad aggiornamento automatico (serie QnA, System Q)	RIFR	Un, n1, n2, n3, d			9	—		11.5.15
	RIFR_P							

3 Configurazione delle istruzioni

3.1 La struttura di una istruzione

La maggior parte delle istruzioni è composta da un campo istruzioni e da un campo operando. Altre istruzioni non necessitano di un campo operando e sono quindi formate dal solo campo istruzione.



Campo istruzione

Il campo istruzione descrive le funzioni dell'istruzione.

PLUS $\hat{=}$ somma

Campo operando

Il campo operando descrive le costanti o variabili necessarie. Il campo operando può comprendere tre elementi: la sorgente dei dati (s), la destinazione dei dati (d), ed il numero (n).

3.1.1 Sorgente dei dati (s)

- La sorgente dei dati indica gli operandi che devono essere processati dall'istruzione..
Nelle istruzioni a 16-bit la notazione della sorgente dati è s.
Nelle istruzioni a 32-bit la notazione è s+1 e s.
- Nella sorgente dati si possono specificare costanti o variabili.

Costanti

Le costanti specificano un valore numerico costante che deve essere elaborato dall'istruzione. Questo valore viene costantemente impostato dal programma scritto dall'utente e non può essere alterato durante l'esecuzione del programma. Si consiglia di qualificare l'indice di ciascuna variabile che deve essere usata come costante.

Variabili

Le variabili specificano gli operandi che contengono i dati che devono essere processati dall'istruzione (fare anche riferimento al capitolo 3.4).

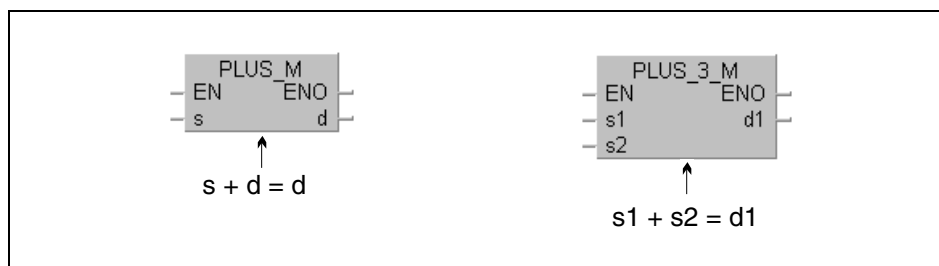
I dati devono essere memorizzati nell'operando prima dell'esecuzione dell'istruzione. I dati memorizzati in variabili possono essere modificati durante l'esecuzione del programma.

3.1.2 Destinazione dei dati (d)

- La destinazione dati indica gli operandi che devono memorizzare i dati dopo che sono stati processati dall'istruzione.
Nelle istruzioni a 16-bit la notazione della destinazione dati è d.
Nelle istruzioni a 32-bit la notazione è d+1 e d.
Tuttavia, alcune istruzioni con 2 operandi hanno bisogno di memorizzare nella destinazione dati d un valore da processare, prima che l'esecuzione venga eseguita. In questo caso, anche il risultato dell'operazione viene memorizzato nello stesso operando.

Esempio:

L'istruzione somma per dati BIN a 16 bit. In questo caso, d contiene prima i dati per l'operazione, e successivamente il risultato:



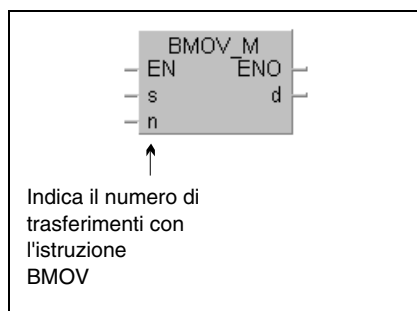
- Un operando usato come memorizzazione dati deve essere sempre impostato come destinazione dati.

3.1.3 Numero (n)

- Il numero n indica quanti operandi vengono usati, oppure quante volte si deve eseguire l'istruzione.

Esempio:

L'istruzione BMOV per il trasferimento di un blocco dati:



- Il valore n può variare fra 0 e 32767. Se viene specificato 0, l'istruzione non viene eseguita.

3.2 Notazione delle istruzioni

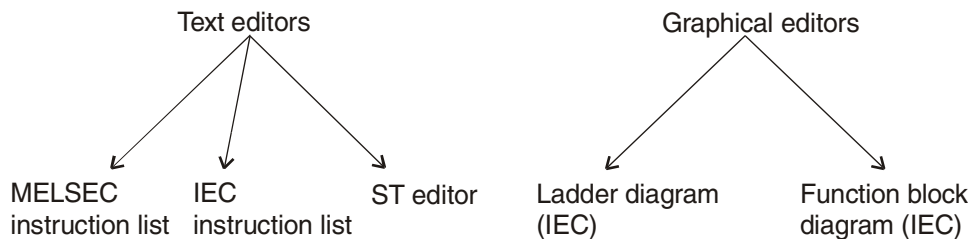
Alcune caratteristiche dell'istruzione possono essere ricavate dalla sua notazione.

3.2.1 16/ 32-bit e impulso

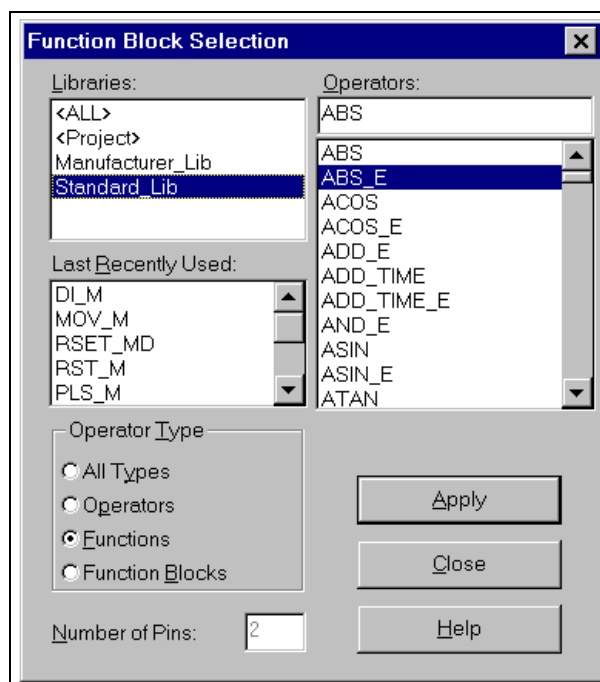
SORT	elaborazione a 16 bit
<u>SORTP</u>	elaborazione a 16 bit con impulso
<u>DSORT</u>	elaborazione a 32 bit
<u>DSORTP</u>	elaborazione a 32 bit con impulso

3.2.2 MELSEC e IEC

Il pacchetto GX IEC developer contiene diversi editor per le istruzioni:

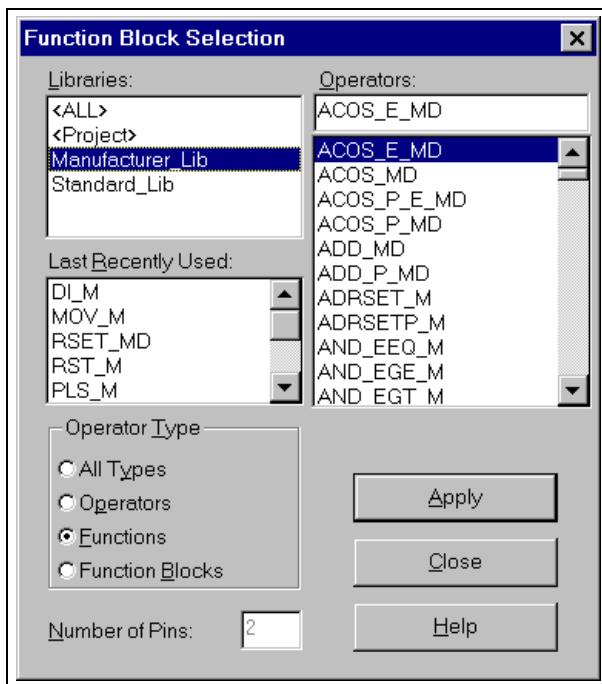


Usando questi editor, le istruzioni vengono rappresentate con notazioni diverse.



Per la scelta di una istruzione in GX IEC Developer, compare questa casella di dialogo. A seconda della libreria scelta si possono scegliere diverse istruzioni:

- ALL: istruzioni MELSEC e IEC
- Project: funzioni e blocchi funzione creati dall'utente
- Manufacturer: istruzioni MELSEC
- Standard: istruzioni IEC



Ad esempio, questa casella di dialogo compare quando viene selezionata la libreria Manufacturer (costruttore). Questo listato contiene le istruzioni MELSEC "adattate".

Le funzionalità delle funzioni "pure" e "adattate" sono identiche. Differiscono solo nella notazione.

Legenda delle estensioni nell'editor IEC:

Estensione nell'editor IEC	Significato
_M	Istruzione MELSEC
_P_M	Esecuzione impulsiva di una istruzione
_MD	Istruzione MELSEC dedicata (fare riferimento anche al capitolo 3.3)
_P_MD	Esecuzione impulsiva di una istruzione dedicata
_K_MD	Uso di una costante in una istruzione dedicata
_K_P_MD	Uso di una costante ed esecuzione impulsiva in una istruzione dedicata.
_S_MD	Istruzione dedicata MELSEC per CPU del System Q
_P_S_MD	Esecuzione impulsiva di una istruzione dedicata MELSEC per CPU del System Q

3.2.3 Ulteriori caratteristiche della notazione istruzioni

La tabella seguente contiene i simboli che rappresentano diverse funzioni nell'editor MELSEC. La colonna sulla destra mostra i corrispondenti nomi delle istruzioni nell'editor IEC.

Esempio:

Editor MELSEC, Editor IEC

LD\$>LD_STRING_GT_M

Editor MELSEC	Editor IEC
\$	STRING
=	EQ
<>	NE
<=	LE
<	LT
>=	GE
>	GT
+	PLUS
-	MINUS
x	MULTI
/	DIVID

3.2.4 Specifica della notazione

I capitoli da 5 a 8, che forniscono una descrizione dettagliata delle istruzioni, contengono illustrazioni relative a entrambi gli editor, cioè entrambe le notazioni. La riga di intestazione contiene l'istruzione MELSEC "pura", come compare nella lista istruzioni MELSEC.

NOTA

La panoramica tabellare all'inizio di ciascuna categoria di istruzioni rappresenta sempre entrambe le notazioni.

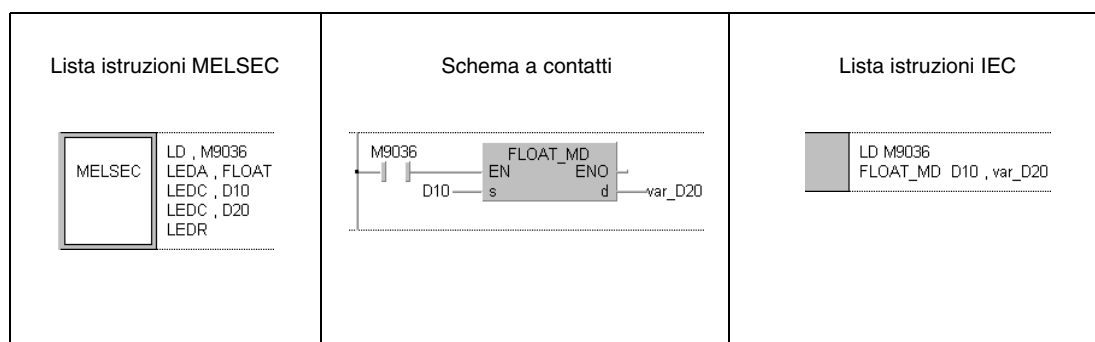
3.3 Programmazione di istruzioni dedicate

Le istruzioni dedicate sono istruzioni personalizzate che non differiscono dalla notazione delle istruzioni MELSEC pure. Queste istruzioni richiedono una particolare tecnica di programmazione per le diverse CPU.

Per poter disporre delle funzioni dell'istruzione `FLOAT_MD` anche nell'editor MELSEC di una CPU serie A, è necessaria una procedura particolare. Nell'editor MELSEC l'istruzione `FLOAT_MD` deve essere programmata in combinazione con le istruzioni `LEDA`, `LEDC`, `LEDR`. Negli editor IEC le istruzioni dedicate possono essere programmate normalmente.

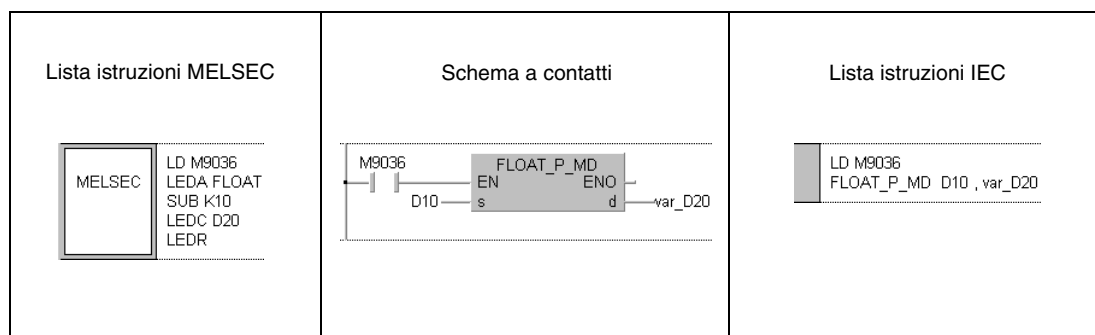
Esempio:

Programmazione dell'istruzione `FLOAT_MD` (esecuzione normale a 16 bit)



Esempio:

Programmazione dell'istruzione `FLOAT_P_MD`
(esecuzione impulsiva a 16-bit, uso di una costante nell'operando s)



Fare riferimento ai manuali seguenti per ulteriori informazioni sulla programmazione delle istruzioni dedicate:

- GX IEC Developer - Manuale di riferimento
- Manuale di programmazione (istruzioni dedicate)

3.4 Programmazione di variabili

3.4.1 Programmazione con GX IEC Developer

Oltre al campo istruzione, la maggioranza delle istruzioni contengono anche un campo operando con variabili specificate. Queste variabili contengono i valori necessari all'esecuzione dell'istruzione.

A seconda dell'editor selezionato in GX IEC Developer si deve utilizzare un metodo diverso di programmazione delle variabili.

Nell'editor MELSEC:

I registri dati D100 e D10 possono essere assegnati direttamente alle variabili D100 e D10.

Il PLC collegato rileva automaticamente che sono in effetti designati gli operandi:

D100=D100 e D101

D10=D10, D11, D12, D13

Nell'editor IEC:

Nell'editor IEC è possibile inserire operandi diretti solo se deve essere designato solo quell'operando.

Esempio: AND D10

Prima di poter processare una istruzione DWSUMP_M, le variabili devono essere definite nell'intestazione dell'unità organizzativa del programma (POU).

Esempio: Intestazione AWL IEC

	Class	Identifier	Type	Initial	Comment
0	VAR	var_D100	DINT	0	
1	VAR	var_D10	ARRAY [0..3] OF INT	4(0)	

var_D100 e var_D10 vengono inseriti come identificatori. In effetti il PLC non assegna gli operandi D100 e D10, ma alloca internamente delle aree libere per le variabili.

Esempio: DWSUMP

DWSUMP	var_D100,	4,	var_D10
	s	n	d
	↑	↑	↑
	32 bit	16 bit	array
		o	
		costante	

La variabile var_D100 è di tipo DINT (32 bit). La variabile var_D10 è di tipo ARRAY. L'array contiene quattro registri a 16 bit di tipo INT (vedi anche capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer").

Specifica della notazione

La designazione var_D100 o var_D10 nella figura indica che non sono definiti direttamente degli operandi, ma degli identificatori. In questi casi la definizione della variabile è obbligatoria! Se una istruzione può essere programmata solo ricorrendo alla definizione di una variabile, questo viene indicato esplicitamente.

NOTA

Come identificatore può essere inserito un nome qualsiasi (ad es. Motore 1, Spia). I nomi var_D100 o var_D10 sono stati scelti in questo caso per chiarezza di confronto con la programmazione nell'editor MELSEC.

La tabella delle variabili all'inizio di ciascuna istruzione fornisce una panoramica dei tipi di dati degli operandi di ciascuna istruzione (l'esempio mostra l'istruzione DWSUM 7.5.14).

Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s	Primo numero dell'operando che contiene i dati da sommare.	BIN 32-bit	ANY32
d	Primo numero dell'operando che memorizza il risultato.	BIN 64-bit	Array [1..4] di ANY16
n	Numero di blocchi dati da sommare	BIN 16-bit	ANY16

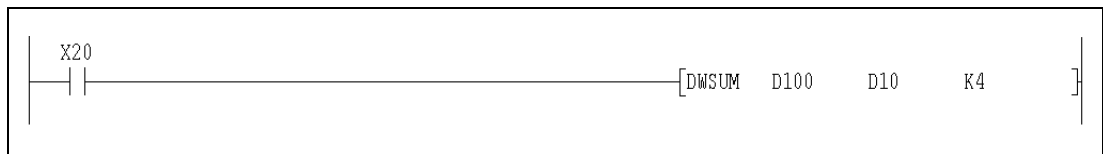
3.4.2 Programmazione con GX Developer

I registri dati D100 e D10 possono essere assegnati direttamente alle variabili D100 e D10.

Il PLC collegato rileva automaticamente che sono in effetti designati gli operandi:

D100= D100 e D101

D10 = D10, D11, D12, D13



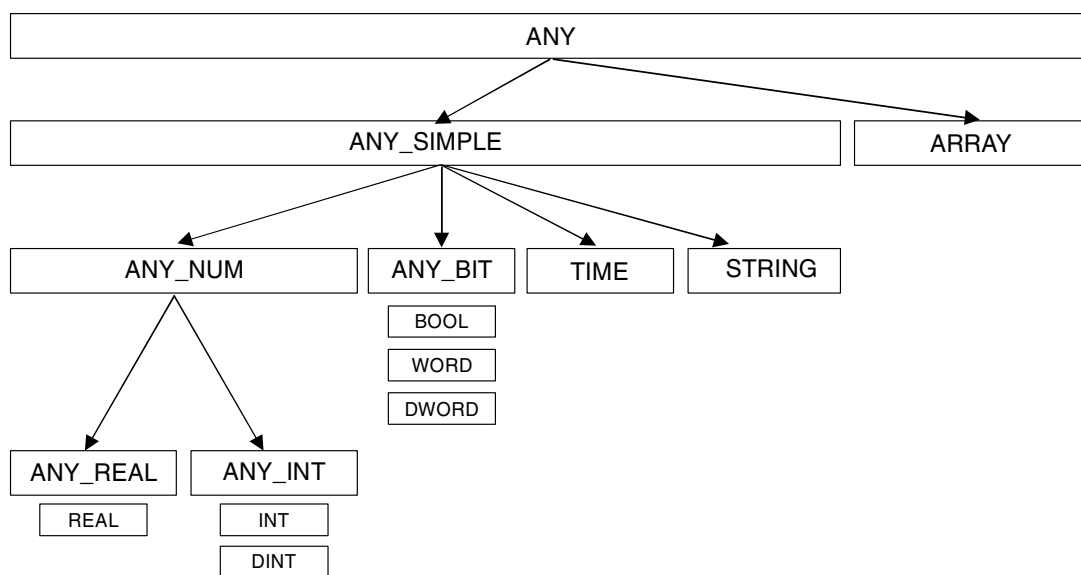
3.5 Tipi di dati

Il tipo di dati determina il numero e l'elaborazione dei bit, insieme al campo di valori delle variabili.

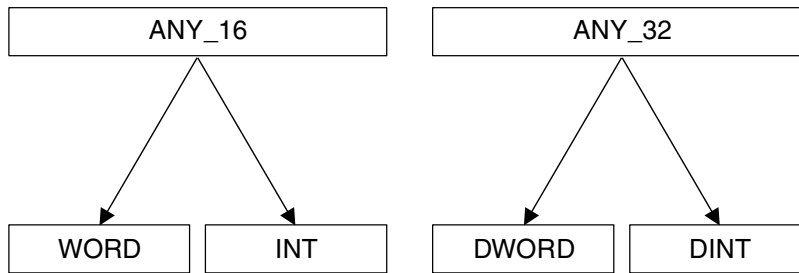
Sono definiti i seguenti tipi di dati:

Tipo dati		Campo di valori	Numero di bit	Tipo CPU applicabile
BOOL	Booleano	0 (FALSE), 1 (TRUE)	1 bit	serie A Serie Q System Q
INT	Intero	da -32.768 a 32.767	16-bit	
DINT	Doppio intero	da -2.147.483.648 a 2.147.483.647	32-bit	
WORD	Stringa di 16 bit	da 0 a 65.535	16-bit	
DWORD	Stringa di 32 bit	da 0 a 4,294,967,295	32-bit	
REAL	Virgola mobile	3.4 +/- 38 (7 cifre)	32-bit	
TIME	Valore temporizzatore	da T#-24d-0h31m23s648.00ms a T#24d20h31m23s647.00ms	32-bit	Serie Q System Q
STRING	Stringa caratteri	max. 50 caratteri		

Gerarchia dei tipi dati ANY



Gerarchia dei tipi dati ANY16 e ANY32



Tipo dati	Significato
ANY	Qualsiasi tipo dati
ANY_SIMPLE	Tipo dati semplice
ANY_NUM	Tipo dati numerico
ANY_REAL	Virgola mobile
ANY_INT	Tipo dati intero
ANY_BIT	Tipo dati elaborazione bit
ANY_16	Qualsiasi tipo dati a 16 bit
ANY_32	Qualsiasi tipo dati a 32 bit
TIME	Tempo
STRING	Stringa caratteri
REAL	Virgola mobile
INT	Valore intero
DINT	Valore doppio intero
BOOL	Valore booleano
WORD	Word (16-bit)
DWORD	Doppia word (32 bit)
ARRAY	Array

● Uso dei blocchi di bit

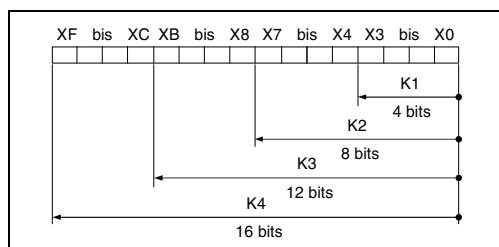
I singoli bit possono essere raggruppati in blocchi di quattro e quindi elaborati come dati a word. Una descrizione dettagliata viene fornita nella sezione seguente: "Elaborazione dati a word (16/32 bit)".

Elaborazione di dati a word (16 bit)

● Uso degli operandi a bit

Gli operandi a bit possono processare dati a word, se viene determinato il numero di operandi a bit (indirizzi). Fino a 16 bit possono essere elaborati in blocchi di 4 bit ciascuno. La lunghezza di ciascun blocco (cioè la definizione delle cifre) viene indicata con la notazione da K1 a K4.

- K1X0 4 indirizzi da X0 a X3
- K2X0 8 indirizzi da X0 a X7
- K3X0 12 indirizzi da X0 a XB
- K4X0 16 indirizzi da X0 a XF

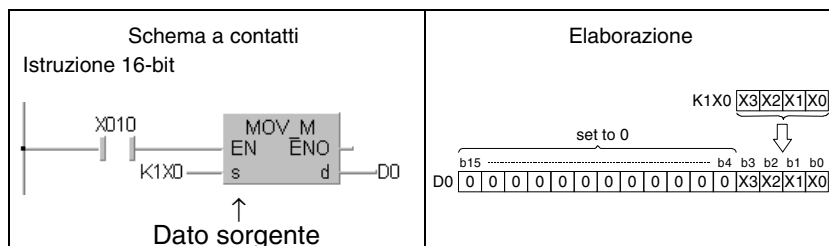


● Indicazione dei blocchi di bit per s

La tabella seguente mostra il campo di valori elaborati come sorgente dati in funzione della definizione delle cifre del dato sorgente (s)

Definizione cifre	Istruzione 16-bit
K1 (4 cifre)	da 0 a 15
K2 (8 cifre)	da 0 a 255
K3 (12 cifre)	da 0 a 4095
K4 (16 cifre)	da -32768 a 32767

Gli indirizzi di bit non utilizzati vengono impostati a zero.

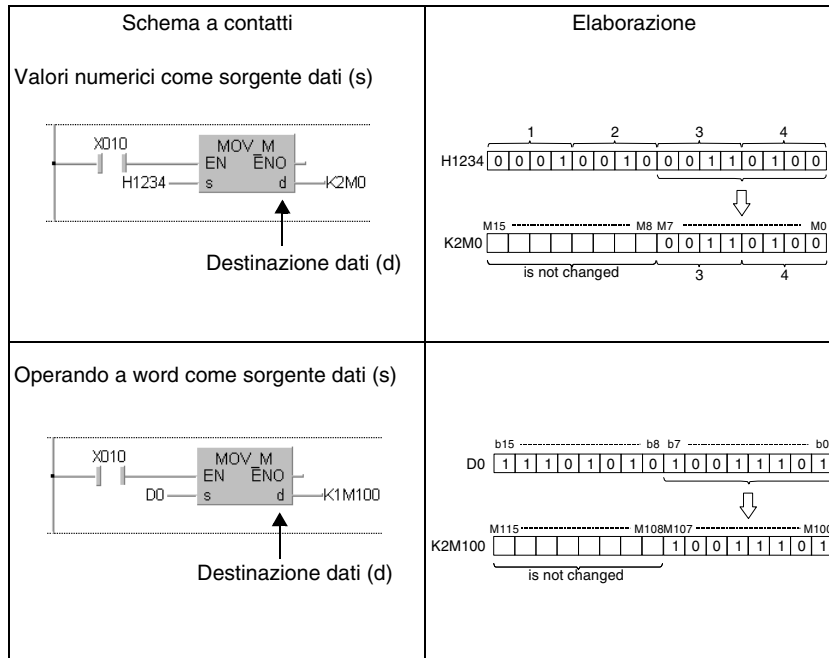


NOTA

Nell'indirizzamento blocco per blocco di operandi a bit, il numero del primo operando a bit (numero iniziale dell'operando) può essere indicato con qualsiasi valore casuale.

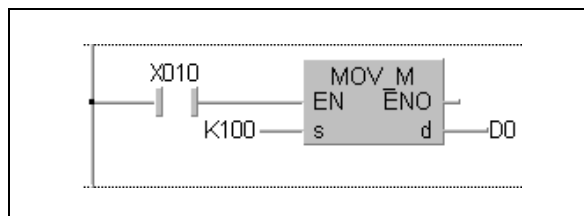
● Indicazione dei blocchi di bit per d

L'indicazione delle cifre per il dato di destinazione (d) determina il campo di indirizzi in cui i dati devono essere scritti. Gli indirizzi di bit che superano il campo di indirizzamento definito vengono ignorati.



● Uso degli operandi a word

Gli operandi a word sono definiti con un indirizzo. Questo indirizzo contiene 16 bit.

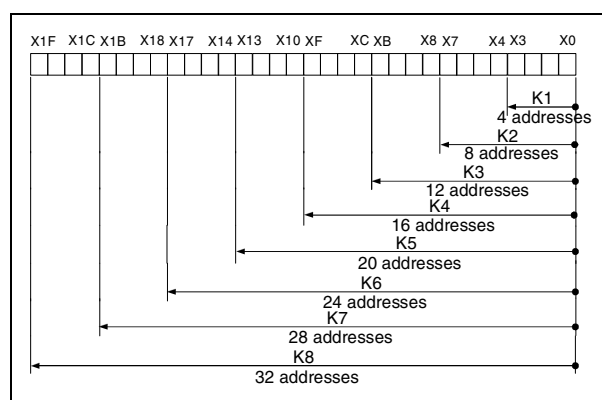


Elaborazione di dati a doppia word (32 bit)

● **Uso degli operandi a bit**

Gli operandi a bit possono processare dati a word, se viene determinato il numero di operandi a bit (indirizzi). Fino a 32 bit possono essere elaborati in blocchi di 4 bit ciascuno. La lunghezza di ciascun blocco (cioè la definizione delle cifre) viene indicata con la notazione da K1 a K8.

- K1X0 4 indirizzi da X0 a X3
- K2X0 8 indirizzi da X0 a X7
- K3X0 12 indirizzi da X0 a XB
- K4X0 16 indirizzi da X0 a XF
- K5X0 20 indirizzi da X0 a X13
- K6X0 24 indirizzi da X0 a X17
- K7X0 28 indirizzi da X0 a X1B
- K8X0 32 indirizzi da X0 a X1F

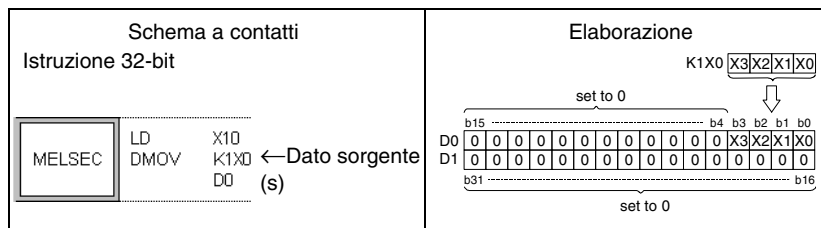


● **Indicazione dei blocchi di bit per s**

La corrispondenza fra indicazione delle cifre e campo di valori processati come sorgente dati, è riportata nella tabella seguente:

Definizione cifre	Istruzione 32-bit
K1 (4 cifre)	da 0 a 15
K2 (8 cifre)	da 0 a 255
K3 (12 cifre)	da 0 a 4095
K4 (16 cifre)	da -32768 a 32767
K5 (20 cifre)	da 0 a 1048575
K6 (24 cifre)	da 0 a 16777215
K7 (28 cifre)	da 0 a 268435455
K8 (32 cifre)	da -2147483648 a 2147483647

Gli indirizzi di bit non utilizzati vengono impostati a zero.

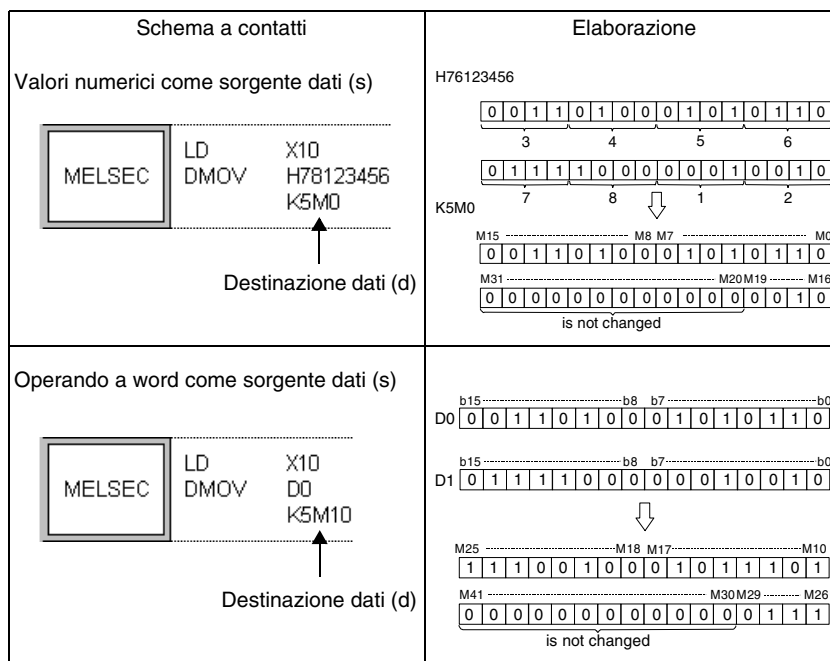


NOTA

Nell'indirizzamento blocco per blocco di operandi a bit, il numero del primo operando a bit (numero iniziale dell'operando) può essere indicato con qualsiasi valore casuale.

- Indicazione dei blocchi di bit per d

L'indicazione delle cifre per il dato di destinazione (d) determina il campo di indirizzi in cui i dati devono essere scritti. Gli indirizzi di bit che superano il campo di indirizzamento definito vengono ignorati.

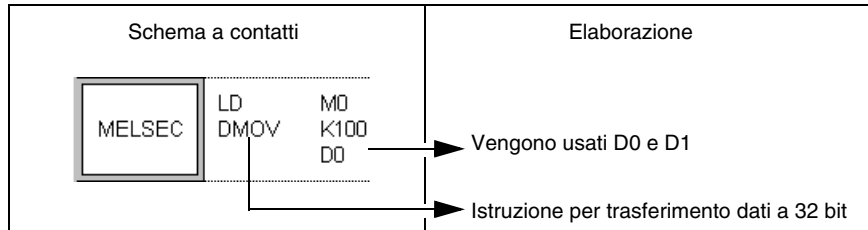


- Uso degli operandi a word

Gli operandi a doppia word sono composti da due operandi a 16 bit.

A seconda del software di programmazione e dell'editor selezionato, gli operandi a doppia word vengono programmati diversamente.

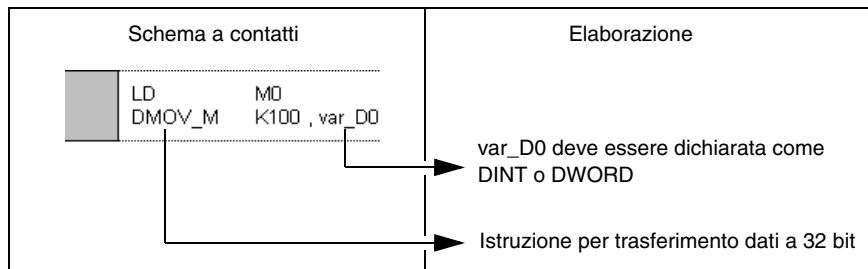
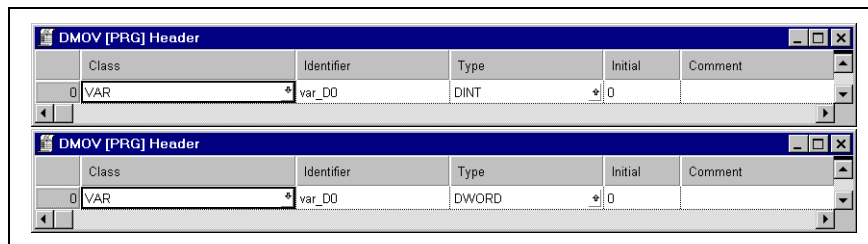
- Nell'editor MM e MELSEC di GX IEC Developer



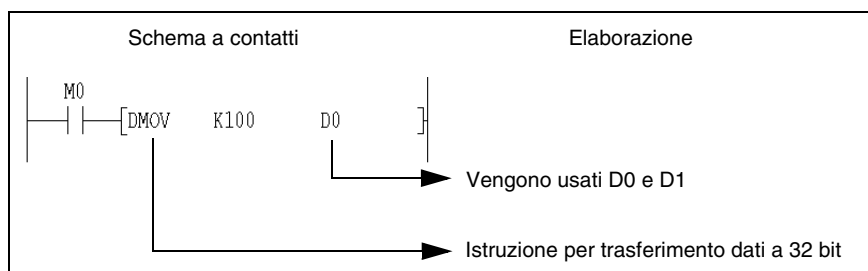
- Nell'editor IEC di GX IEC Developer

Per poter programmare operandi a 32 bit con l'editor IEC di GX IEC Developer, le variabili devono essere definite nell'intestazione dell'unità organizzativa di programma (POU).

I tipi dati DWORD e DINT sono dati a 32-bit.



- Nell'editor di GX Developer



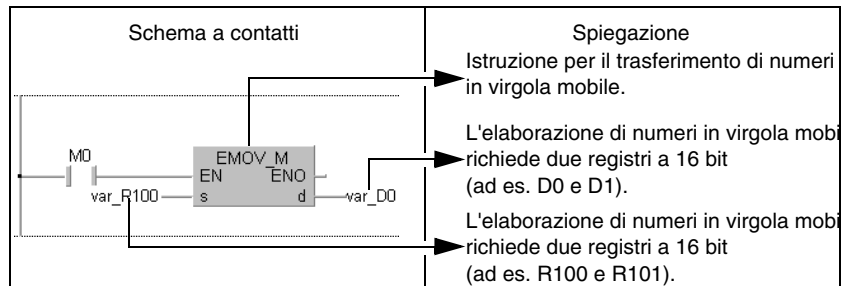
Elaborazione di dati di tipo REAL

I dati tipo REAL sono numeri a 32 bit in virgola mobile.

I dati in virgola mobile possono essere contenuti solo in operandi a word.

Gli operandi che processano numeri in virgola mobile sono indirizzati tramite i 16 bit meno significativi dell'istruzione. Il numero a 32 bit in virgola mobile viene memorizzato in due registri a 16 bit consecutivi.

Se un tipo dati REAL deve essere processato da una CPU AnA/AnU, si devono utilizzare le istruzioni dedicate corrispondenti (vedi capitolo 3.3, "Programmazione di istruzioni dedicate").



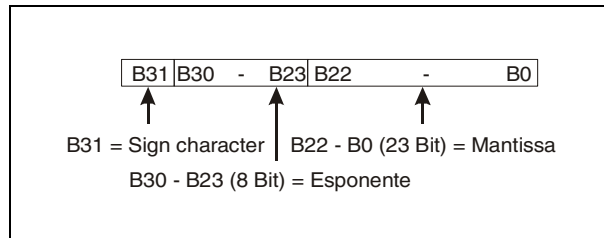
NOTA

GX IEC Developer indica i numeri in virgola mobile con E □. I numeri in virgola mobile per l'elaborazione delle istruzioni iniziano con E.

Per contenere un numero in virgola mobile, sono necessari due operandi a word. Il numero viene suddiviso nei seguenti componenti:

segno; $2^{[\text{esponente}]}$; [mantissa]

La configurazione dei bit nei registri ed il loro contenuto viene mostrata nella figura seguente:



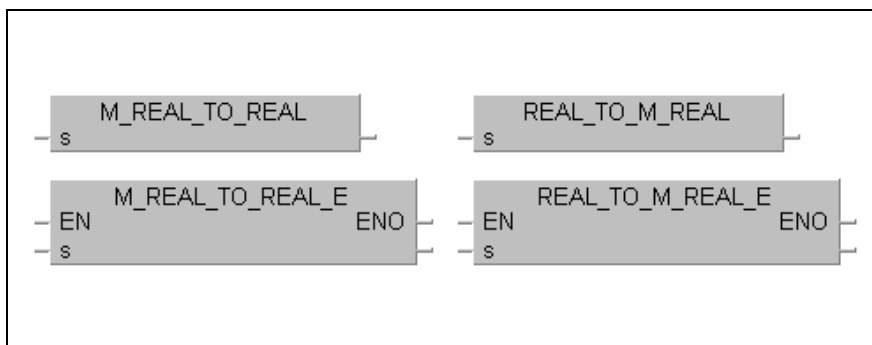
- Segno del numero in virgola mobile: il segno è contenuto in b31.
0 = Positivo
1 = Negativo
- Esponente: il numero n corrispondente a 2^n è contenuto in binario da b23 a b30. Il significato del valore binario di n è mostrato nella figura seguente.

b23 to b30	FFH	FEH	FDH		81H	80H	7FH	7EH		02H	01H	00H
n	free	127	126		2	1	0	-1		-125	-126	free

Esempio: Se i bit da b23 a b30 contengono il valore binario codificato 81H, allora n=2.

- Mantissa: Con i 23 bit da b0 a b22 si possono rappresentare in binario 7 cifre (XXXXXX o 1,XXXXXX).

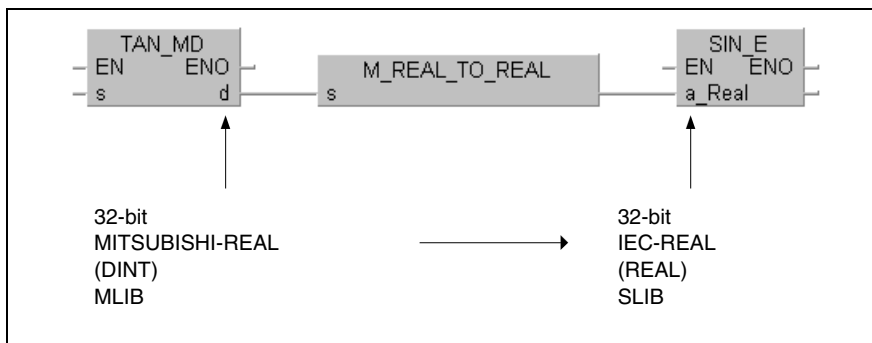
Dato che la funzione REAL definita da IEC utilizza dati di tipo REAL come ingresso/uscita, mentre le istruzioni MELSEC usano dati tipo DINT, vengono fornite le funzioni seguenti per compensare questa differenza:



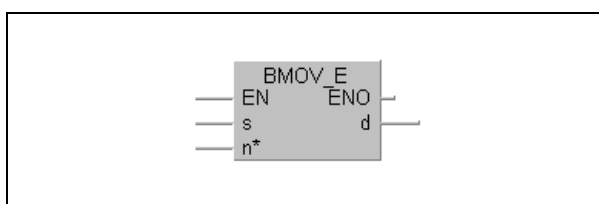
La conversione dal tipo dati REAL IEC al tipo dati MELSEC viene eseguito dalle istruzioni REAL_TO_M_REAL (REAL_TO_M_REAL_E).

La conversione dal tipo dati MELSEC al tipo dati REAL IEC viene eseguito dalle istruzioni M_REAL_TO_REAL (M_REAL_TO_REAL_E).

Esempio: Per utilizzare le istruzioni dedicate che processano il tipo dati REAL e per le istruzioni IEC è necessaria la conversione da REAL a REAL.



Nella programmazione con GX IEC Developer, l'istruzione BMOV_E può essere usata per disabilitare il controllo delle variabili. Non viene creato nessun codice aggiuntivo.



Qualsiasi tipo di dati può essere specificato da s, sono ammessi anche array. n contiene il numero del dato a 16 bit da copiare.

3.5.2 Indirizzamento di array e registri con GX IEC Developer

Indirizzamento di registri a 32 bit

L'indirizzamento di registri a 32 bit (tipi dati DINT, DWORD) necessita della definizione di una variabile nell'intestazione del programma.

Nell'esempio seguente, l'istruzione DMOV richiede due registri a 16 bit per spostare una word dati a 32-bit. Per l'indirizzamento tramite l'editor MELSEC di GX IEC Developer, sono indicati solo i registri iniziali (in questo caso D10, D20). Entrambi i secondi registri a 16 bit necessari (D11, D21) sono indirizzati automaticamente dal compilatore.

Nell'editor IEC di GX IEC Developer, invece del registro iniziale, si deve definire una variabile (in questo caso var_D10, var_D20) con un tipo dati specifico (DINT - 32 bit) nella intestazione dell'unità organizzativa di programma, secondo l'intestazione dell'istruzione. Per queste variabili, il compilatore assegna degli indirizzi interni corrispondenti.

<p style="text-align: center;">Lista istruzioni MELSEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; text-align: center;">MELSEC</td> <td style="width: 15%; text-align: center;">LD</td> <td style="width: 15%; text-align: center;">DMOV</td> <td style="width: 15%; text-align: center;">X0</td> <td style="width: 15%; text-align: center;">D10</td> <td style="width: 15%; text-align: center;">D20</td> </tr> </table>	MELSEC	LD	DMOV	X0	D10	D20	<p style="text-align: center;">Schema a contatti</p>	<p style="text-align: center;">Lista istruzioni IEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%;"></td> <td style="width: 15%; text-align: center;">LD</td> <td style="width: 15%; text-align: center;">DMOV_M</td> <td style="width: 15%; text-align: center;">X0</td> <td style="width: 15%; text-align: center;">var_D10</td> <td style="width: 15%; text-align: center;">var_D20</td> </tr> </table>		LD	DMOV_M	X0	var_D10	var_D20									
MELSEC	LD	DMOV	X0	D10	D20																		
	LD	DMOV_M	X0	var_D10	var_D20																		
<p>Intestazione dell'istruzione DMOV</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 5%;"></th> <th style="width: 25%;">Class</th> <th style="width: 10%;">Identifier</th> <th style="width: 15%;">Type</th> <th style="width: 10%;">Initial</th> <th style="width: 35%;">Comment</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">0</td> <td>VAR_INPUT</td> <td style="text-align: center;">s</td> <td>ANY32</td> <td style="text-align: center;">0</td> <td></td> </tr> <tr> <td style="text-align: center;">1</td> <td>VAR_OUTPUT</td> <td style="text-align: center;">d</td> <td>ANY32</td> <td style="text-align: center;">0</td> <td></td> </tr> </tbody> </table>							Class	Identifier	Type	Initial	Comment	0	VAR_INPUT	s	ANY32	0		1	VAR_OUTPUT	d	ANY32	0	
	Class	Identifier	Type	Initial	Comment																		
0	VAR_INPUT	s	ANY32	0																			
1	VAR_OUTPUT	d	ANY32	0																			
<p>Intestazione dell'unità organizzativa di programma (POU)</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 5%;"></th> <th style="width: 25%;">Class</th> <th style="width: 10%;">Identifier</th> <th style="width: 15%;">Type</th> <th style="width: 10%;">Initial</th> <th style="width: 35%;">Comment</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">0</td> <td>VAR</td> <td style="text-align: center;">var_D10</td> <td>DINT</td> <td style="text-align: center;">0</td> <td></td> </tr> <tr> <td style="text-align: center;">1</td> <td>VAR</td> <td style="text-align: center;">var_D20</td> <td>DINT</td> <td style="text-align: center;">0</td> <td></td> </tr> </tbody> </table>							Class	Identifier	Type	Initial	Comment	0	VAR	var_D10	DINT	0		1	VAR	var_D20	DINT	0	
	Class	Identifier	Type	Initial	Comment																		
0	VAR	var_D10	DINT	0																			
1	VAR	var_D20	DINT	0																			

Indirizzamento di array

Per programmare le istruzioni che utilizzano elementi di un array come operandi di ingresso o uscita (registri a 16 bit), le variabili nell'intestazione del POU devono essere definite in modo conforme all'intestazione dell'istruzione.

I singoli elementi dell'array vengono indirizzati specificando l'array e l'elemento racchiuso in parentesi quadre (var_xx[x]).

Le figure che seguono mostrano l'indirizzamento tramite array dell'istruzione di posizionamento per tavole rotanti (ROTC):

<p style="text-align: center;">Lista istruzioni MELSEC</p> <pre style="font-family: monospace; border: 1px solid black; padding: 5px;"> MELSEC LD X0 OUT M0 LD X1 OUT M1 LD X2 OUT M2 LD X10 ROT D200 K10 K2 M0 </pre>	<p style="text-align: center;">Schema a contatti</p>	<p style="text-align: center;">Lista istruzioni IEC</p> <pre style="font-family: monospace; border: 1px solid black; padding: 5px;"> LD X0 ST var_M0[0] LD X1 ST var_M0[1] LD X2 ST var_M0[2] LD X10 ROT var_D200, 10, 2, var_M0 </pre>																														
<p>Intestazione dell'istruzione ROTC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 5%;"></th> <th style="width: 25%;">Class</th> <th style="width: 15%;">Identifier</th> <th style="width: 25%;">Type</th> <th style="width: 10%;">Initial</th> <th style="width: 20%;">Comment</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>VAR_INPUT</td> <td>s</td> <td>ARRAY [1.. 3] OF ANY16</td> <td>0,0,0</td> <td></td> </tr> <tr> <td>1</td> <td>VAR_INPUT</td> <td>n1</td> <td>ANY16</td> <td>0</td> <td></td> </tr> <tr> <td>2</td> <td>VAR_INPUT</td> <td>n2</td> <td>ANY16</td> <td>0</td> <td></td> </tr> <tr> <td>3</td> <td>VAR_OUTPUT</td> <td>d</td> <td>ARRAY [1.. 8] OF BOOL</td> <td>8(FALSE)</td> <td></td> </tr> </tbody> </table>				Class	Identifier	Type	Initial	Comment	0	VAR_INPUT	s	ARRAY [1.. 3] OF ANY16	0,0,0		1	VAR_INPUT	n1	ANY16	0		2	VAR_INPUT	n2	ANY16	0		3	VAR_OUTPUT	d	ARRAY [1.. 8] OF BOOL	8(FALSE)	
	Class	Identifier	Type	Initial	Comment																											
0	VAR_INPUT	s	ARRAY [1.. 3] OF ANY16	0,0,0																												
1	VAR_INPUT	n1	ANY16	0																												
2	VAR_INPUT	n2	ANY16	0																												
3	VAR_OUTPUT	d	ARRAY [1.. 8] OF BOOL	8(FALSE)																												
<p>Intestazione dell'unità organizzativa di programma (POU)</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 5%;"></th> <th style="width: 25%;">Class</th> <th style="width: 15%;">Identifier</th> <th style="width: 25%;">Type</th> <th style="width: 10%;">Initial</th> <th style="width: 20%;">Comment</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>VAR</td> <td>var_D200</td> <td>ARRAY [0.. 2] OF INT</td> <td>3(0)</td> <td></td> </tr> <tr> <td>1</td> <td>VAR</td> <td>var_M0</td> <td>ARRAY [0.. 7] OF BOOL</td> <td>8(FALSE)</td> <td></td> </tr> </tbody> </table>				Class	Identifier	Type	Initial	Comment	0	VAR	var_D200	ARRAY [0.. 2] OF INT	3(0)		1	VAR	var_M0	ARRAY [0.. 7] OF BOOL	8(FALSE)													
	Class	Identifier	Type	Initial	Comment																											
0	VAR	var_D200	ARRAY [0.. 2] OF INT	3(0)																												
1	VAR	var_M0	ARRAY [0.. 7] OF BOOL	8(FALSE)																												

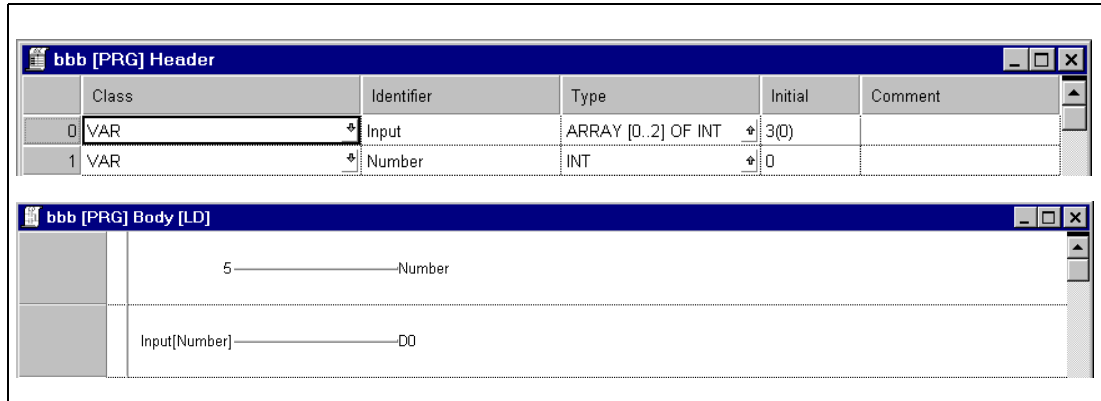
Dall'intestazione dell'istruzione ROTC si può ricavare che l'operando di ingresso s è composto da 3 elementi di un array di tipo ANY16, mentre l'operando di uscita è composto da 8 elementi di un array di tipo BOOL.

In GX Developer e nell'editor MELSEC di GX IEC Developer per indicare la serie di operandi s e d, è necessario specificare solo gli operandi iniziali D200 e M0. Il compilatore indirizza i registri da D200 a D202 per s e da M0 a M7 per d.

Nell'editor IEC si devono definire degli array per s e d. L'array di ingresso s viene definito come var_D200. Questo è composto da 3 elementi di array (var_D200[0] - var_D200[2]) di tipo INT (intero a 16 bit). L'array di uscita d viene definito come var_M0. Questo è composto da 8 elementi di array (var_M0[0] - var_M0[7]) di tipo BOOL (bit). Per queste variabili, il compilatore assegna degli indirizzi interni corrispondenti.

NOTA

Gli array possono essere anche indirizzati in modo variabile. In questo caso, invece del numero di elemento racchiuso fra parentesi quadre, si inserisce un qualsiasi identificatore, ad esempio [Numero]. "Numero" deve essere definito nell'intestazione dell'unità organizzativa di programma. È quindi possibile inserire un valore corrispondente all'elemento di array desiderato nel registro "Numero".



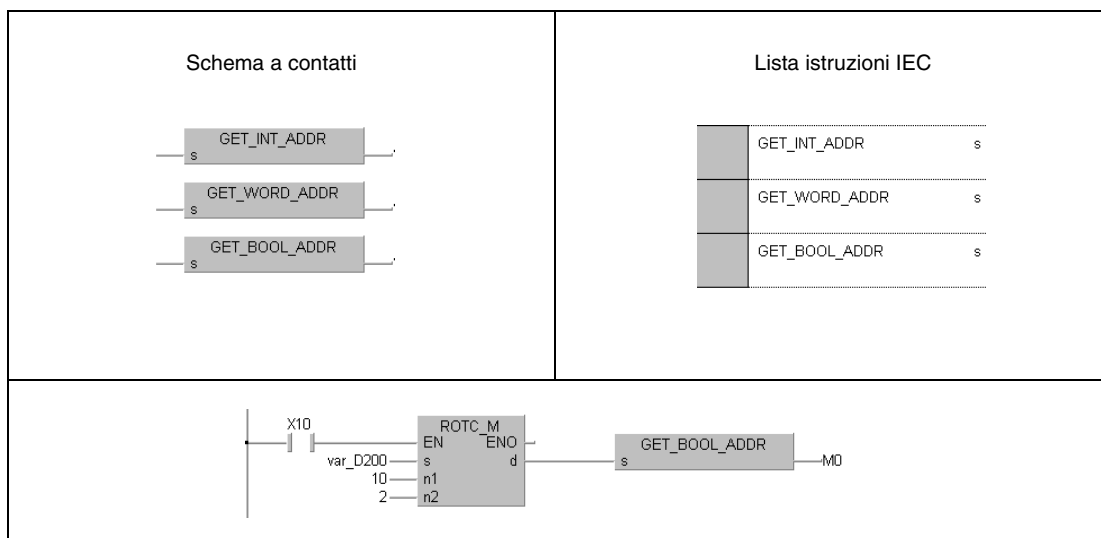
Istruzioni per la conversione fra indirizzo array e indirizzo iniziale

Il set di istruzioni per la conversione di un array di uscita in un indirizzo iniziale di una serie di operandi è formato da tre istruzioni.

L'istruzione GET_INT_ADDR converte un array di uscita con elementi di tipo INT (interi 16 bit) in un indirizzo iniziale di una serie di operandi.

L'istruzione GET_WORD_ADDR converte un array di uscita con elementi di tipo WORD (word 16 bit) in un indirizzo iniziale di una serie di operandi.

L'istruzione GET_BOOL_ADDR converte un array di uscita con elementi di tipo BOOL (bit) in un indirizzo iniziale di una serie di operandi.



Dopo la conversione gli elementi dell'array possono essere utilizzati come operandi singoli. In questo caso quindi, non è necessaria la definizione di una variabile nell'intestazione dell'unità organizzativa di programma.

Nel programma con l'istruzione ROTC illustrato in precedenza, invece degli elementi dell'array var_M0[0] – var_M0[7], possono essere usati i relé da M0 a M7.

I metodi di indirizzamento degli operandi in GX Developer e GX IEC Developer sono identici.

Queste istruzioni convertono solo array di uscita. Gli array di ingresso devono essere indirizzati e dichiarati come descritto in precedenza.

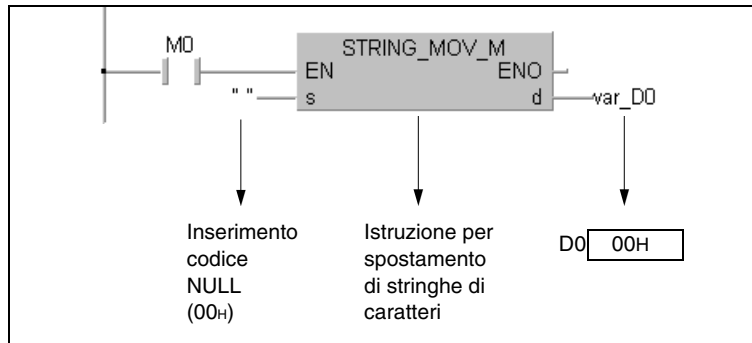
3.5.3 Uso di dati a stringhe di caratteri (STRING)

L'istruzione STRING (\$) processa stringhe di caratteri.

Le stringhe di caratteri sono composte da tutti i caratteri inseriti (massimo 50 caratteri) fino al codice NULL (00H).

- Se il carattere inserito è il codice NULL (00H)

Per memorizzare un codice NULL è necessaria una word (registro) dati.

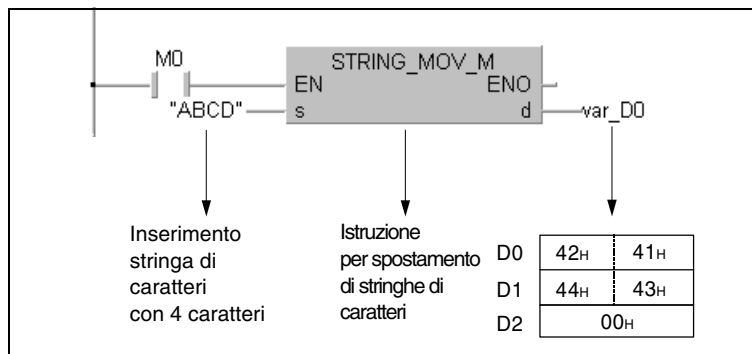


- Se il numero di caratteri contenuti nella stringa è pari

La memorizzazione di una stringa di caratteri con un numero di caratteri pari necessita di un numero di word dati calcolato con la formula seguente:

$$(Numero\ di\ caratteri / 2) + 1$$

Se ad esempio la stringa di caratteri "ABCD" viene spostata in D0, i registri D0 e D1 sono necessari per contenere la stringa, mentre il registro D2 è necessario per il codice NULL che indica la fine della stringa.

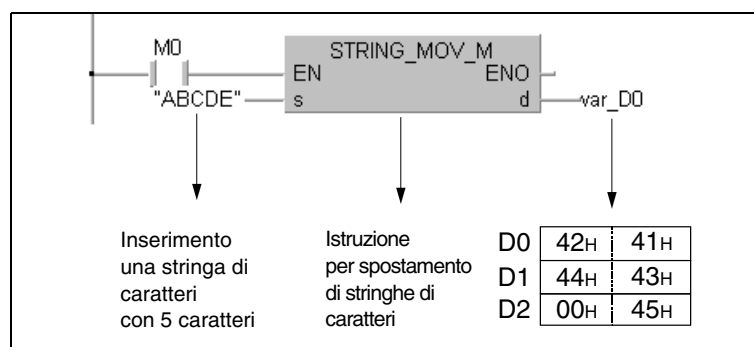


- Se il numero di caratteri contenuti nella stringa è dispari

La memorizzazione di una stringa di caratteri con un numero di caratteri dispari necessita di un numero di word dati calcolato con la formula seguente:

(Numero di caratteri / 2)

Se ad esempio la stringa di caratteri "ABCDE" viene spostata in D0, i registri da D0 a D2 sono necessari per contenere la stringa. Il codice NULL che indica la fine della stringa viene memorizzato nel byte alto di D2.



3.6 Indicizzazione

Dato che il System Q e la serie Q differiscono dalla serie A per quanto riguarda il meccanismo di indicizzazione, le caratteristiche dei diversi tipi di CPU vengono descritte separatamente nei capitoli 3.6.1 e 3.6.2.

L'indicizzazione è un metodo di indirizzamento indiretto di un operando, tramite un registro indice. Tramite l'indicizzazione all'interno di un programma, l'indirizzo di un operando si ottiene sommando il numero dispositivo inserito esplicitamente, con il contenuto del registro indice.

Uso dell'indicizzazione nel programma

Il programma seguente mostra un esempio di indicizzazione. Nella prima riga di programma, il valore 1 viene assegnato al registro indice Z0. Questo registro funge da indice per D10 nella seconda riga di programma. Di conseguenza, D0 memorizza il valore di D11 ($D10Z = D(10+1) = D11$).

Schema a contatti	Spiegazione
	La costante 1 viene inserita nel registro indice Z0.
	Il dato corrispondente al registro indice designato Z0 ($D10+Z0(1)=D11$) viene memorizzato in D0.

Lo schema seguente mostra un altro esempio di indicizzazione che chiarifica l'uso con operandi ($Z0=20$, $Z1=5$).

Schema a contatti	Spiegazione
	La costante 20 viene inserita nel registro indice Z0.
	La costante 5 viene inserita nel registro indice Z1.
	La costante 100 viene indicizzata da Z0 ($100+Z0(20)=120$) e memorizzata nel registro W53 ($W53 + Z1(5)=W58$) indicizzato da Z1.

Operandi che possono essere oggetto di indicizzazione.

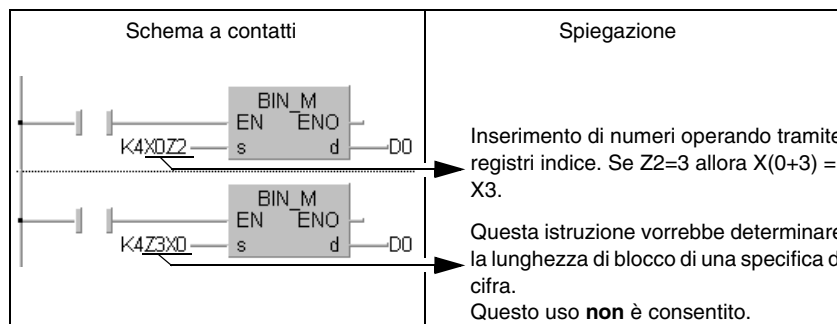
L'indicizzazione può essere applicata a operandi, contatti e bobine. Il registro indice serve come indirizzamento indiretto di un operando e contiene un valore numerico da -32768 a 32767.

Operandi che non possono essere oggetto di indicizzazione.

Operando	Significato
E	Virgola mobile
\$	Stringa caratteri
□.□	Indirizzamento dei bit di operandi a word
FX, FY, FD	Operandi funzione
P	Puntatori usati come etichetta
I	Puntatori di interrupt usati come etichetta
Z	Registri indice
S	Relé di passo
TV, STV	Valori di impostazione dei timer
CV	Valori di impostazione dei contatori
N	Livelli di annidamento
A0	Accumulatore
A1	Accumulatore

Dati a bit (tranne AnN)

Gli operandi possono essere oggetto di indicizzazione anche per la scelta della cifra. La lunghezza dei blocchi dell'indicazione della cifra non può essere modificata.



3.6.1 Caratteristiche speciali delle CPU System Q e QnA

Le CPU del System Q e della serie QnA possiedono 16 registri indice (Z0 - Z15). La tabella seguente mostra i campi di valori di timer e contatori che possono essere determinati tramite indicizzazione:

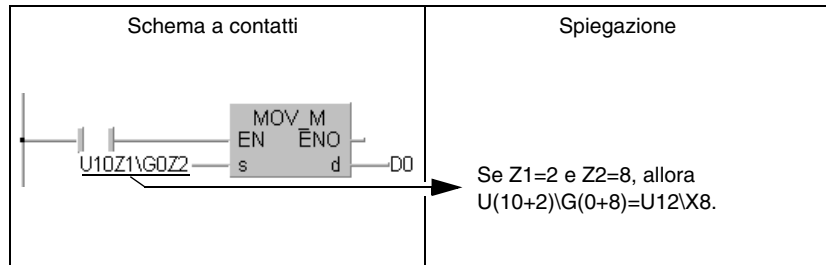
Operando	Significato	Esempio applicativo
TC	Solo i registri Z0 e Z1 possono essere utilizzati per l'indirizzamento di contatti e bobine di timer.	
CC	Solo i registri Z0 e Z1 possono essere utilizzati per l'indirizzamento di contatti e bobine di contatori.	

NOTA Non ci sono limitazioni nell'indirizzamento dei valori attuali di timer e contatori.

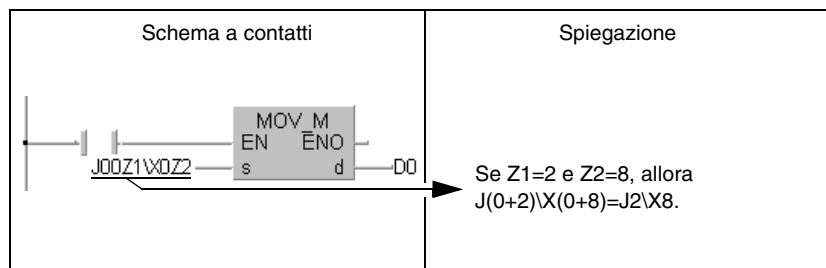
Schema a contatti	Spiegazione
	Impostazione valore del timer (TV). Indicizzazione non supportata.
	Valore attuale del timer (TN). Indicizzazione supportata.
	Impostazione valore contatore (CV). Indicizzazione non supportata.
	Valore attuale del contatore (CN). Indicizzazione

Un'altra differenza rispetto alle CPU serie A è il supporto della indicizzazione per numeri di I/O, indirizzi dei buffer di memoria, numeri di rete e numeri dispositivo di moduli di rete.

Lo schema che segue mostra la designazione di numeri di I/O e indirizzi di buffer di memoria nei moduli funzione speciali.



Lo schema seguente mostra la designazione di numeri di rete e numeri dispositivo di moduli di rete.



NOTA

Vedi anche "Manuale di programmazione CPU QnA (Fondamenti)", "Manuale utente Q CPU (modo Q) (Funzioni/fondamenti di programmazione)" ed i manuali dei moduli corrispondenti, per ulteriori informazioni su moduli funzione speciali e moduli di rete.

3.6.2 Caratteristiche speciali delle CPU AnA, AnAS e AnU

I numeri operando all'interno di un programma possono essere determinati tramite un indice (Z o V).
Nei casi seguenti si verifica un errore di funzionamento durante l'elaborazione delle istruzioni.

- Il campo di indirizzamento degli operandi viene superato durante la indicizzazione.
In questo caso le costanti K e H sono omesse.
- L'indirizzo iniziale di una serie di operandi supera il campo relativo durante la indicizzazione.

NOTA

Per ridurre i tempi di elaborazione, le CPU AnA, AnAS e AnU non verificano i numeri operando durante la indicizzazione. Per questa ragione gli errori dovuti alla indicizzazione non vengono riconosciuti come errori di elaborazione.

Se si verifica un errore dovuto alla indicizzazione, i dati degli operandi possono essere modificati illecitamente.

I programmi che utilizzano il meccanismo di indicizzazione devono essere quindi scritti con la massima cura!

Nelle CPU AnA, AnAS e AnU la indicizzazione può essere effettuata anche con operandi a bit usati con istruzioni LD, OUT o simili.

Memorizzazione di dati a 32 bit in registri indice

Dati a 32 bit possono essere memorizzati nei registri indice estesi (da Z1 a Z6 e da V1 a V6) di una CPU AnA o AnU. I registri indice seguenti devono quindi essere usati a coppie:

Z1 e V1

Z2 e V2

Z3 e V3

Z4 e V4

Z5 e V5

Z6 e V6

Zn contiene i 16 bit meno significativi, Vn contiene i 16 bit più significativi. In una istruzione a 32 bit, si deve indicare solo l'operando Z. Se viene indicato l'operando V, il programma non può essere elaborato.

I valori a 32 bit possono essere memorizzati solo nelle coppie di registri indicate sopra. Altre combinazioni non sono consentite. Se un operando di una coppia di registri viene utilizzato per la indicizzazione di una istruzione, i dati di questo registro vengono elaborati come dati a 16 bit per la indicizzazione.

3.7 Indirizzamento indiretto (solo GX Developer)

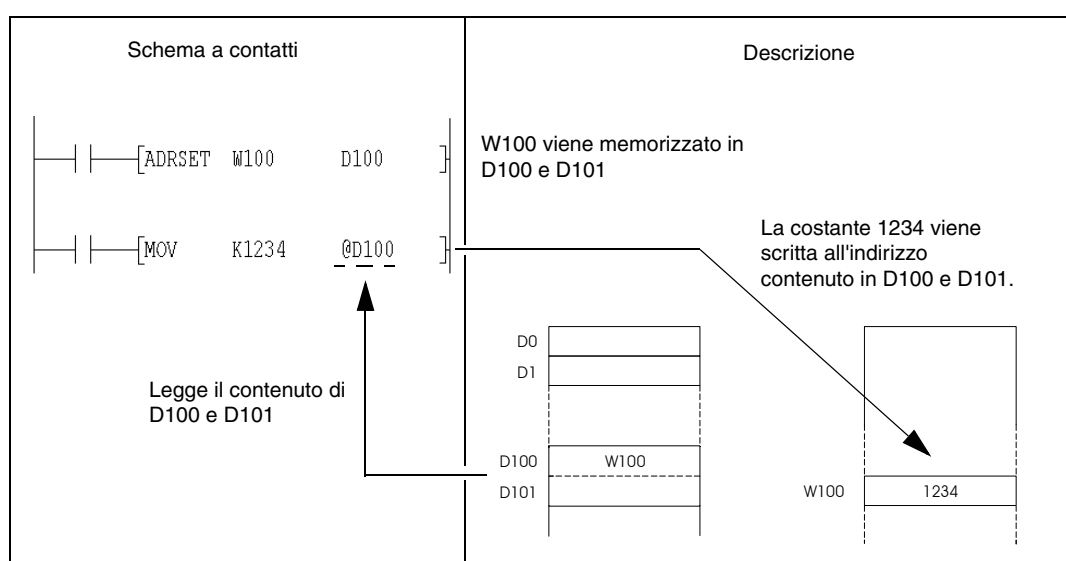
Con l'indirizzamento indiretto, l'indirizzo di un operando è contenuto in un dispositivo a word. Nel programma della sequenza, l'indirizzo dell'operando non viene determinato direttamente. Per le operazioni relative a quell'operando, viene usato invece l'operando a word. Questo metodo può essere utilizzato quando il registro indice non è sufficiente.

L'operando che contiene l'indirizzo dell'operando oggetto dell'indirizzamento indiretto, ha il prefisso "@". Ad esempio, programmando @D100, il contenuto di D100 e D101 viene utilizzato come indirizzo dell'operando.

L'indirizzo dell'operando oggetto dell'indirizzamento indiretto può essere memorizzato in un operando a word tramite l'istruzione ADRSET.

NOTA

L'istruzione ADRSET non è supportata da GX IEC Developer.



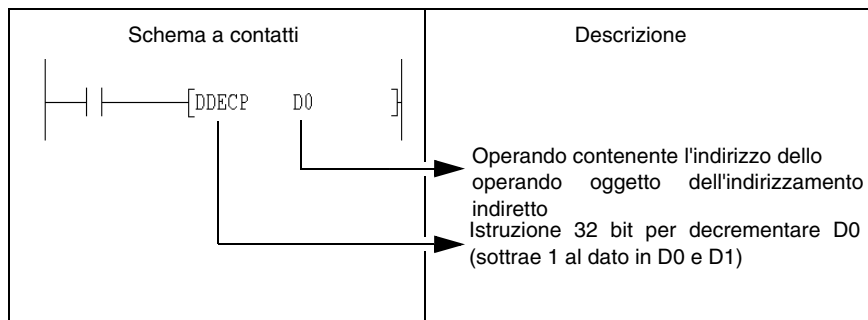
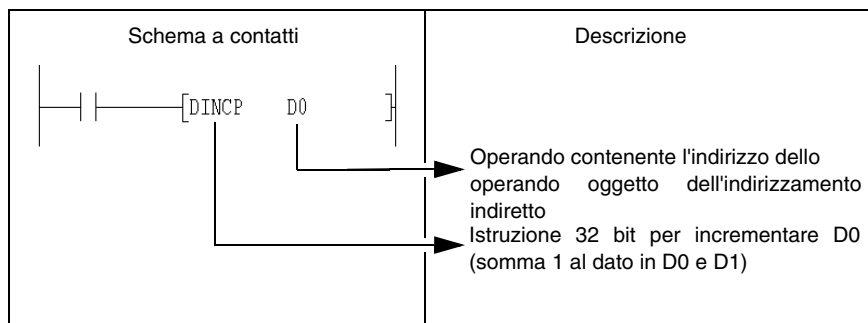
Un elenco di operandi che possono essere indirizzati in modo indiretto è mostrato di seguito.

Tipo operando		Indirizzamento indiretto	Esempio di indirizzamento indiretto
Operandi interni (sistema, utente)	Operandi a bit	Impossibile	—
	Operandi a word	Possibile	<ul style="list-style-type: none"> • @D100 • @D100Z2 (indicizzazione)
MELSECNET/10	Operandi a bit	Impossibile	—
	Operandi a word	Possibile (L'istruzione ADRSET non può essere usata per scrivere l'indirizzo indiretto)	<ul style="list-style-type: none"> • @J1\W10 • @J1Z1\W10Z2 (indicizzazione)
Modulo funzione speciale			<ul style="list-style-type: none"> • @U10\G0 • @U10Z1\G0Z2 (indicizzazione)
Registro indice Zn		Impossibile	—
File registri		Possibile	<ul style="list-style-type: none"> • @R0, @ZR20000 • @R0Z1, @ZR20000Z1 (indicizzazione)
Annidamento		Impossibile	—
Puntatore			—
Costanti			—
Altro			—

NOTA Vedi anche "Manuale di programmazione CPU QnA (Fondamenti)" o il "Manuale utente Q CPU (modo Q) (Funzioni/fondamenti di programmazione)" per ulteriori informazioni sui nomi di operando.

NOTA Per memorizzare un indirizzo per l'indirizzamento indiretto vengono utilizzate due word. Quindi per incrementare o decrementare un indirizzo utilizzato per indirizzamento indiretto tramite istruzioni aritmetiche, sono necessarie operazioni di somma o sottrazione a 32 bit.

Nel programma di esempio che segue, l'operando che memorizza l'indirizzo indiretto viene incrementato e decrementato con istruzioni a 32 bit. Così facendo, l'indirizzo dell'operando oggetto dell'indirizzamento indiretto viene incrementato o decrementato di 1.



3.8 Errore di esecuzione

Un errore di esecuzione si verifica nei casi seguenti:

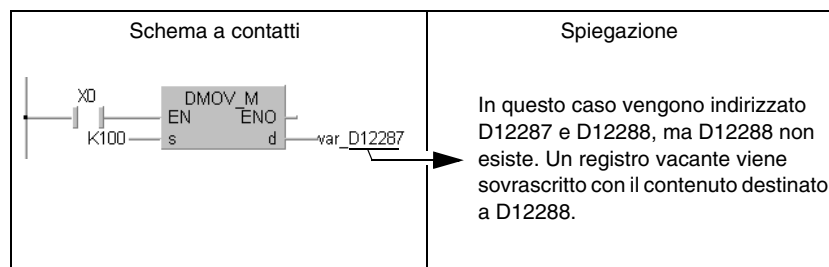
- Se vengono riscontrate le condizioni di errore descritte nel paragrafo "Errori di esecuzione" per le singole istruzioni, viene restituito un codice di errore.
- Se viene utilizzato un registro buffer, ma non esiste nessun modulo funzione speciale collegato al numero di I/O specificato.
- Se viene usato un dispositivo di collegamento, ma la rete corrispondente non esiste.
- Se viene utilizzato un dispositivo di collegamento, ma non esiste nessun modulo di rete collegato al numero di I/O specificato.

NOTA

Se viene specificato nei parametri un file registri, ma nessuna memory card è installata (solo System Q e CPU serie Q), viene restituito un codice di errore (2401 = errore impostazione file). Se viene indirizzato un file registri, ma nessun file registri è specificato nei parametri, viene restituito un codice di errore. Se viene letto il file registri, viene restituito il codice "FFFFH".

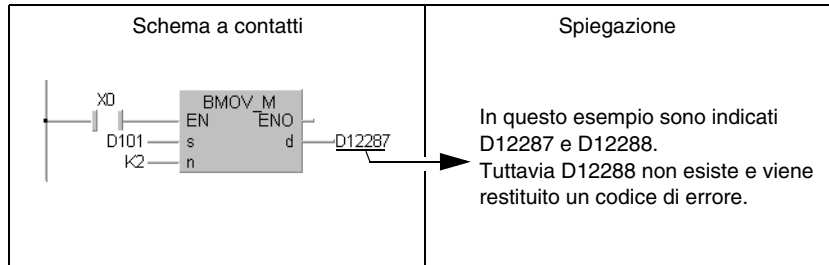
3.8.1 Verifica del campo operando

- Se le istruzioni utilizzano operandi a lunghezza fissa (MOV, DMOV, ecc.), il campo dell'operando non viene verificato.
In quei casi in cui il campo di indirizzamento ammesso viene superato, i dati vengono scritti in un registro vacante.
Se ad esempio, sono configurati 12k indirizzi, non viene restituito nessun codice di errore fino a quando non viene superato l'indirizzo del registro D12287.



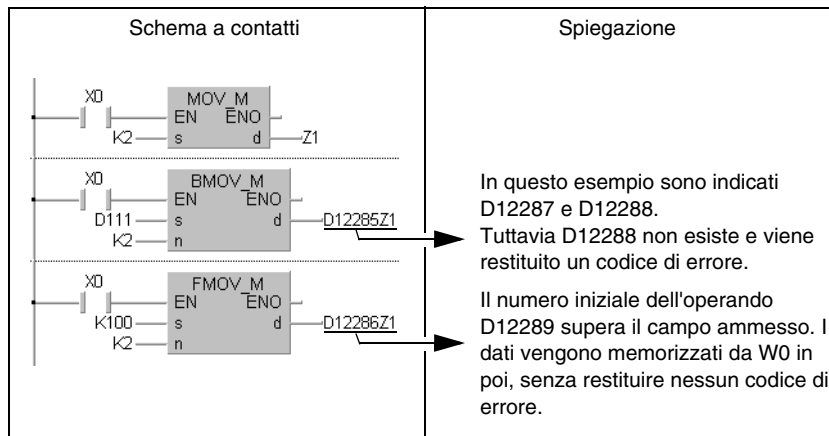
Nell'indicizzazione il campo dell'operando non viene mai verificato.

- Se le istruzioni utilizzano operandi con lunghezza variabile, il campo dell'operando viene verificato (BMOV, FMOV e altre istruzioni che indicano indirizzi iniziali). In quei casi in cui il campo operando ammesso viene superato, viene restituito un codice di errore. Se ad esempio, sono configurati 12k indirizzi, viene restituito un codice di errore solo quando viene superato l'indirizzo D12287.



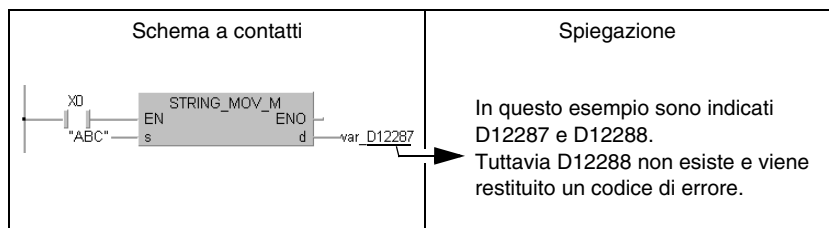
Il campo operando viene verificato anche per le indicizzazioni.

Non viene restituito nessun codice di errore se il numero iniziale dell'operando supera il campo ammesso.



Dato che le stringhe di caratteri sono a lunghezza variabile, il campo operando viene verificato. In quei casi in cui il campo operando ammesso viene superato, viene restituito un codice di errore.

Se ad esempio, sono configurati 12k indirizzi, non viene restituito nessun codice di errore fino a quando non viene superato l'indirizzo del registro D12287.



- Il campo operando viene verificato nell'indicizzazione di una uscita diretta (DY).

3.8.2 Verifica del dato operando

Verifica di dati binari

- Se il risultato dell'operazione supera il campo dei valori ammessi, non viene restituito nessun codice di errore. In questo caso la segnalazione di riporto non viene attivata.

Verifica di dati BCD

- Viene verificata ciascuna cifra dei valori BCD (da 0 a 9).
Se una cifra supera il campo da 0 a 9 (da A a F), viene restituito un codice di errore.
- Se il risultato dell'operazione supera il campo dei valori ammessi, non viene restituito nessun codice di errore. In questo caso la segnalazione di riporto non viene attivata.

Verifica di numeri in virgola mobile

Errori di esecuzione avvengono nei casi seguenti:

- Il valore del numero in virgola mobile diventa 0.
- Il valore assoluto del numero in virgola mobile diventa inferiore a 1.0×2^{-127}
- Il valore assoluto del numero in virgola mobile diventa superiore a 1.0×2^{129}

Verifica di stringhe di caratteri

I dati dell'operando non vengono verificati.

3.9 Condizioni di esecuzione delle istruzioni

3.9.1 Condizione di esecuzione

Esistono 4 diversi tipi di condizioni di esecuzione delle istruzioni:

- Esecuzione incondizionata

Le istruzioni sono eseguite senza considerare lo stato di segnale degli operandi.
Esempio: LD X0, OUT Y10

- Esecuzione su ON

Le istruzioni vengono eseguite fino a quando lo stato del segnale è ON.
Esempio: MOV, FROM

- Esecuzione su fronte di salita

Le istruzioni sono eseguite sul fronte di salita (cambiamento dello stato del segnale da 0 a 1) della condizione di esecuzione.
Esempio: PLS, MOVP

- Esecuzione su fronte di discesa

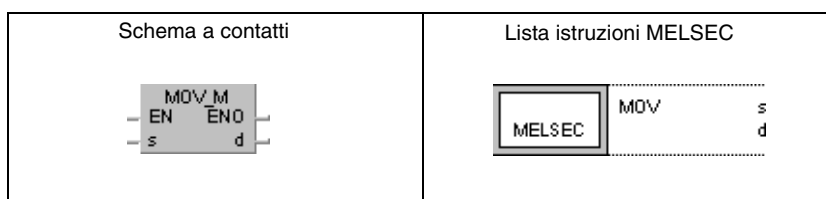
Le istruzioni sono eseguite sul fronte di discesa (cambiamento dello stato del segnale da 1 a 0) della condizione di esecuzione.
Esempio: PLF

La maggioranza delle istruzioni appartengono ai due tipi seguenti:

- Esecuzione su ON
- Esecuzione sul fronte di salita della condizione di esecuzione

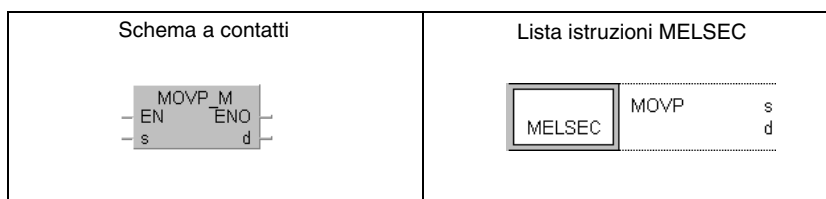
L'istruzione viene eseguita se la condizione di esecuzione è ON. Queste istruzioni non hanno segnalazioni particolari.

Esempio: MOV_M/ MOV

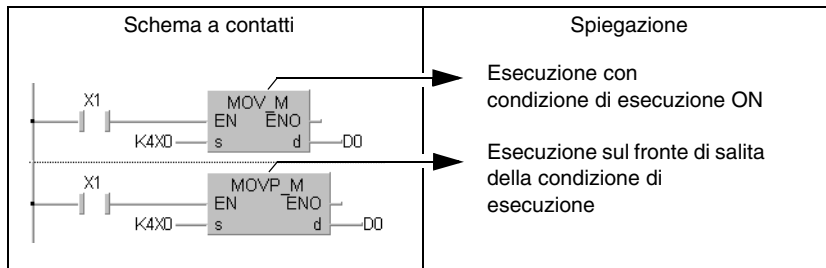


Nella valutazione del fronte di salita della condizione di esecuzione, l'istruzione viene eseguita solo se lo stato del segnale passa da 0 a 1.

Esempio: MOVP_M/ MOVP



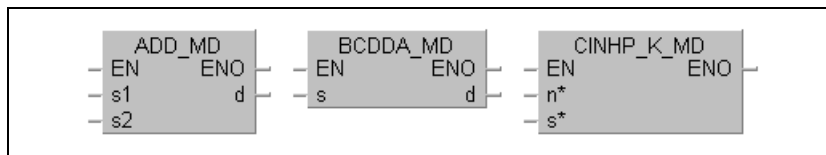
L'esempio seguente mostra l'esecuzione dell'istruzione MOV con la condizione di esecuzione impostata su ON e con esecuzione su fronte di salita:



3.9.2 Ingresso EN e uscita ENO

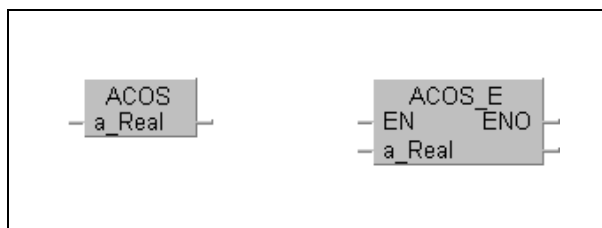
Tutte le istruzioni descritte in questo manuale sono incluse nella libreria costruttore di GX IEC Developer. Oltre alle variabili di ingresso e uscita, queste istruzioni prevedono anche un ingresso EN e una uscita ENO.

La figura seguente mostra diverse istruzioni della libreria costruttore di GX IEC Developer:



Nella libreria standard IEC quasi tutte le istruzioni compaiono due volte. Le funzioni differiscono solo per il suffisso "_E". Queste istruzioni prevedono un ingresso EN e una uscita ENO.

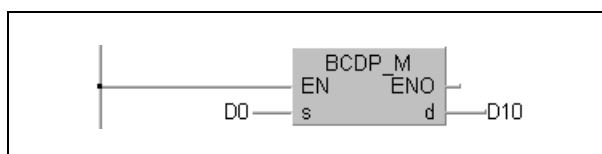
La figura che segue mostra due istruzioni IEC della libreria standard di GX IEC Developer:



L'esempio seguente mostra la differenza di esecuzione dell'istruzione con e senza gli ingressi EN e le uscite ENO.

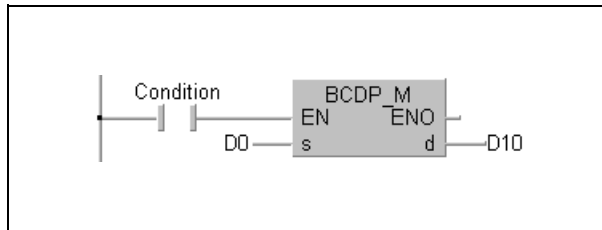
Esempio 1: Senza collegamenti aggiuntivi

Senza collegamento aggiuntivo, le condizioni di esecuzione dell'istruzione è sempre attiva.



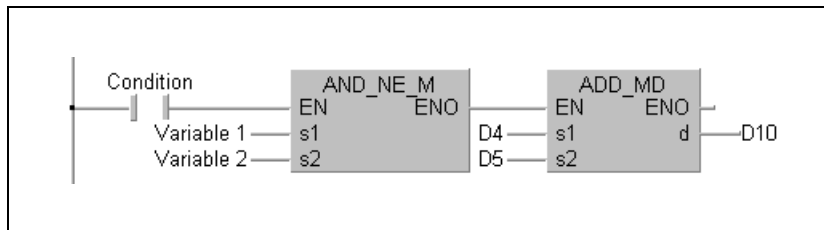
Esempio 2: Collegamento con un contatto

Se l'ingresso EN è collegato ad un contatto, l'istruzione viene eseguita se la condizione è soddisfatta.



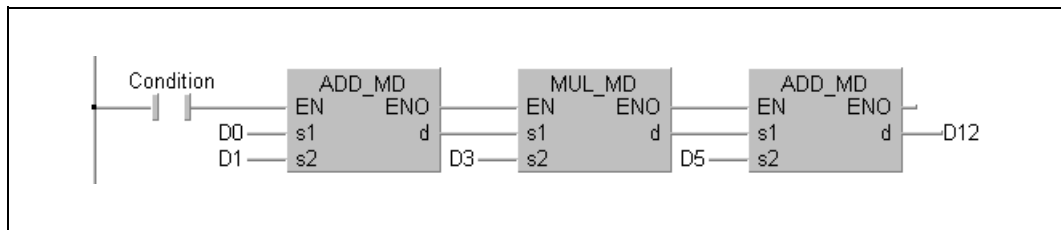
Esempio 3: Collegamento con il risultato di una operazione

Se il risultato booleano di una operazione aritmetica viene collegato all'ingresso EN, l'istruzione viene eseguita solo se il risultato dell'operazione aritmetica è TRUE.



Esempio 4: Collegamento con l'istruzione precedente

Se l'ingresso EN è collegato con l'uscita ENO dell'istruzione precedente, le istruzioni sono eseguite solo se la condizione è soddisfatta.



NOTA

Non è obbligatorio collegare le uscite ENO. Il segnale sull'ingresso EN viene ponticellato con l'uscita ENO. Se l'ingresso EN è "TRUE", anche l'uscita ENO è "TRUE".

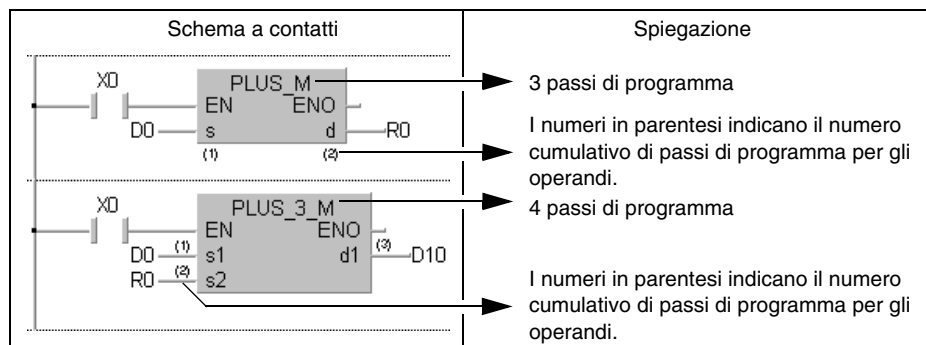
3.10 Numero passi di programma

Per non superare la capacità della memoria interna richiesta e della memoria ROM e RAM delle memory card e delle schede di memoria, è necessario calcolare il numero totale di passi richiesti da un programma. Le sezioni che seguono descrivono il calcolo dei passi di programma per le istruzioni delle CPU del System Q e delle serie Q e A.

3.10.1 Per una CPU System Q e QnA

Il numero di passi di una istruzione dipende dal numero di passi di base. La maggior parte delle istruzioni richiedono solo il numero dei passi di base per la loro esecuzione. Il numero dei passi di base dipende dal numero di operandi usati aumentato di 1.

L'esempio che segue mostra il calcolo del numero di passi di base dell'istruzione PLUS:



- Il numero di passi di programma per l'applicazione delle istruzioni di ingresso e uscita:

Il numero di passi di programma per le istruzioni di ingresso (LD, LDI, AND, ANI, OR, ORI) dipende dagli operandi usati.

Se vengono usati operandi interni o registri file (da R0 a R32767), il numero di passi è 1.

Se vengono usati ingressi ad accesso diretto (DX), il numero di passi è 3.

Il numero di passi di programma per una istruzione di uscita (LDP, LDF, ANDP, ANDF, ORP, ORF) dipende dagli operandi usati.

Se vengono usati operandi interni o registri file (da R0 a R32767), il numero di passi è 2.

Se vengono usati altri operandi, il numero di passi è 4.

- Il numero di passi di programma per diverse istruzioni di trasferimento:

Operandi che aumentano il numero di passi	Passi aggiunti	Esempio
Operandi di moduli funzione speciali	1	MOV <u>U4</u> \G10 D0
Dispositivi di comunicazione		MOV <u>J3</u> \B20 D0
Registri file indirizzati in serie		MOV <u>ZR123</u> D0
Costanti a 32-bit		DMOV <u>K123</u> D0
Numeri in virgola mobile come costanti		EMOV <u>E0.1</u> D0
Stringhe di caratteri	Per un numero dispari: (numero di caratteri / 2) - 1 Per un numero pari: Numero di caratteri / 2	\$MOV <u>„123“</u> D0

Nei casi che presentano diversi di questi fattori, i numeri di passi indicati vengono sommati. Ad esempio, se viene programmata l'istruzione MOV U1\G10 ZR123, 1 passo viene aggiunto per il buffer di memoria ed 1 passo per il file registri indirizzato in serie, per un totale di 2 passi.

3.10.2 Per una CPU AnA, AnAS e AnU

Per le CPU AnA, AnAS o AnU si devono considerare le peculiarità descritte nella sezione che segue.

Il numero di passi aumenta di 1 se una istruzione fa riferimento ad uno dei numeri operando elencati nella tabella seguente (gamma estesa della serie AnA).

Operandi	Campo operando
Relé M, L, S	da 2048 a 8191
Timer T	da 256 a 2047
Contatori C	da 256 a 1023
Relé collegamento B	da 400 a FFF
Registri dati D	da 1024 a 6143
Registri collegamento W	da 400 a FFF

Se qualsiasi operando della gamma di indirizzamento estesa viene indicizzato tramite registro indice, il numero di passi viene incrementato di 1.

La figura seguente mostra diversi esempi per il calcolo dei passi di programma. Il primo esempio mostra la configurazione di passi per la programmazione di istruzioni nel normale campo di indirizzamento.

Gli esempi seguenti mostrano la configurazione di passi di programma per l'uso di operandi del campo di indirizzamento esteso.

Schema a contatti	Spiegazione
	LDT0 1 passo + D0W010 <u>5 passi</u> 6 passi
	LDT300 2 passi + D0W800 <u>6 passi</u> 8 passi
	LDT1000 2 passi + D2000W010 Z1 <u>6 passi</u> 9 passi
	LDT0 1 passo + D2000 Z1D300 <u>5 passi</u> 6 passi

Per l'indicizzazione di istruzioni da 1 passo (ad es. LD o OUT), il numero di passi viene aumentato di 1.

Gli esempi seguenti mostrano la differenza di programmazione con o senza indicizzazione. Il numero di passi aumenta solamente di 1 anche se l'indicizzazione viene applicata tramite un registro indice esteso (da Z1 a Z6, da V1 a V6).

Schema a contatti	Spiegazione
	LDX0 1 passo + OUTY40 <u>1 passo</u> 2 passi
	LDX0 Z 2 passi + OUTY40 <u>1 passo</u> 3 passi Indicizzazione

4 Organizzazione e struttura dei capitoli

Questo capitolo fornisce una introduzione ai capitoli da 5 a 9 e descrive l'organizzazione e la struttura delle descrizioni delle istruzioni per le serie MELSEC A e Q e per il System Q.

La figura seguente mostra che ciascuno di questi capitoli inizia con una tabella che elenca e commenta la struttura e suddivisione in categorie delle istruzioni descritte nel capitolo.

Istruzione	Significato
Istruzioni per operazioni di confronto	Confronto di dati fra loro (ad es. =, >, ≥)
Istruzioni per operazioni aritmetiche	Somma, sottrazione, moltiplicazione, divisione, incremento e decremento di dati BIN e BCD, dati in virgola mobile e blocchi dati BIN Collegamento stringhe di caratteri
Istruzioni di conversione dati	Conversione fra tipi di dati (ad es. BCD -> BIN, BIN -> BCD)
Istruzioni per trasferimento dati	Trasmette un insieme di dati definiti
Istruzioni di salto di programma	Comandi di salto di programma
Istruzioni per controllo esecuzione del programma	Abilita e disabilita gli interrupt di programma
Istruzione di rinfresco	Rinfresca operandi a bit, collegamenti e interfacce di I/O
Altre istruzioni utili	Conteggio avanti e indietro di ingressi single ended o differenziali, timer ad apprendimento, timer per funzioni speciali, controllo rotazione tavola girevole con scelta percorso più breve, segnale di rampa, misura densità impulsi, uscita impulsiva a ciclo fisso, modulazione larghezza impulso, ingresso matrice

Ognuna di queste categorie viene descritta di seguito nel capitolo, e illustrata da esempi di programma.

4.1 Panoramica delle istruzioni

Ciascuna categoria inizia con una tabella che elenca tutte le singole istruzioni descritte nella sezione. Come mostrato dalla figura, tutte le variazioni delle istruzioni vengono rappresentate nella notazione degli editor MELSEC e IEC.

6.1 Istruzioni per operazioni di confronto					
Le istruzioni per operazioni di confronto eseguono confronti fra valori di dati (ad es. uguale =, maggiore >, minore <). La programmazione delle istruzioni di confronto è simile alle istruzioni di base corrispondenti: LD, LDI ⇒ LD=, LDD= AND, ANI ⇒ AND=, ANDD= OR, ORI ⇒ OR=, ORD=					
Funzione	MELSEC Istruzione con Editor MELSEC	MELSEC Istruzione con Editor IEC	Funzione	MELSEC Istruzione con Editor MELSEC	MELSEC Istruzione con Editor IEC
= uguale	LD=	LD_EQ_M	≤ minore o uguale	LD<=	LD_LE_M
	AND=	AND_EQ_M		AND<=	AND_LE_M
	OR=	OR_EQ_M		OR<=	OR_LE_M
	LDD=	LDD_EQ_M		LDD<=	LDD_LE_M
	ANDD=	ANDD_EQ_M		ANDD<=	ANDD_LE_M
	ORD=	ORD_EQ_M		ORD<=	ORD_LE_M
	LDE=	LD_EEQ_M		LDE<=	LD_ELE_M
	ANDE=	AND_EEQ_M		ANDE<=	AND_ELE_M
	ORE=	OR_EEQ_M	ORE<=	OR_ELE_M	

Se si utilizza GX IEC Developer, scegliere sempre l'istruzione IEC, anche se sono disponibili altre notazioni.

4.2 La tabella CPU

Questa sezione che descrive le istruzioni, inizia con una tabella che indica ciascuna CPU (AnS, AnN, AnA, AnAS, AnU, QnA, QnAS, Q4AR, System Q) capace di elaborare l'istruzione. Le CPU abilitate sono indicate con un punto nero.

INT, INTP, DINT, DINTP						Istruzioni di conversione dati
6.3.4 INT, INTP, DINT, DINTP						
CPU	AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
			● ¹	● ¹	●	● ²
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR. ² Non disponibile su Q00JCPU, Q00CPU e Q01CPU						

Qualsiasi dettaglio di elaborazione relativo ad una determinata CPU viene commentato in una nota a pie' di pagina (ad es. istruzioni estese, vedi anche "3.3 Programmazione di istruzioni estese").

4.4 Operandi MELSEC Q

Con il termine "MELSEC Q" vengono raggruppate tutte le CPU del System Q e le CPU delle serie QnA, QnAS e Q4AR.

La tabella "Operandi MELSEC Q" elenca tutti gli operandi che possono essere utilizzati come variabili interne (ad es. s1, s2, d).

Gli operandi non sono elencati separatamente; viene solamente indicato se l'istruzione è capace di indirizzare operandi a bit e/o a word.

Operandi MELSEC Q	Operandi utilizzabili										
	Operando interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro	Segnale di errore	Numero di passi
	Bit	Word		Bit	Word						
s1	●	●	●	●	●	●	●	●	—	—	3
s2	●	●	●	●	●	●	●	●	—	—	

La colonna "File register" indica se l'istruzione supporta l'accesso ai file registri.

La colonna "MELSECNET/10 Direct J□□" specifica se l'istruzione supporta operazioni di lettura/scrittura di dati a bit e/o a word da/per stazioni collegate alla rete MELSECNET/10. "J□" specifica il numero di stazione e "□" il numero di operando.

La colonna "Modulo funzione speciale U□G□" indica se l'istruzione supporta operazioni di lettura/scrittura di dato da/per il buffer di memoria di un modulo funzione speciale installato. "U□" specifica l'indirizzo iniziale di un modulo funzione speciale e "G□" l'indirizzo del buffer di memoria.

La colonna "Registro indice Zn" indica se l'istruzione può essere utilizzata con indicizzazione. L'indicazione se l'istruzione è capace di elaborare costanti decimali (K) o esadecimali (H, 16#) è riportata nella colonna "Costanti K, H (16#)".

La colonna "Altro" indica se l'istruzione utilizza qualsiasi altro operando o costanti.

La colonna "Errore" indica se l'istruzione è capace di impostare il segnale di errore.

Il numero di passi di programma usati è elencato nella colonna "Numero di passi".

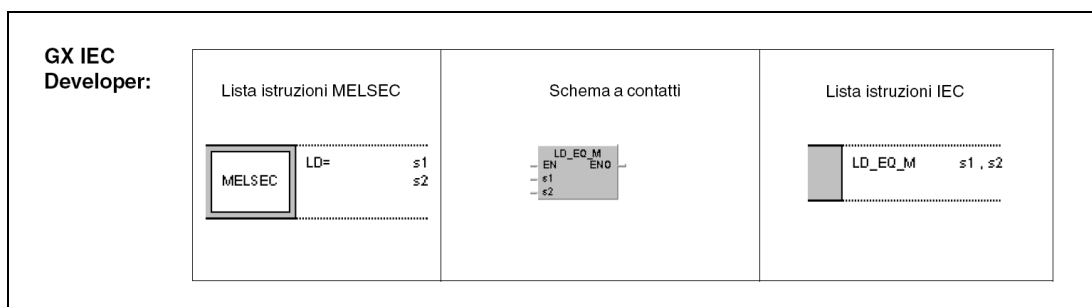
Qualsiasi dettaglio particolare viene commentato in note disposte sotto alla tabella.

4.5 Rappresentazione del formato di una istruzione

4.5.1 Rappresentazione in GX IEC Developer

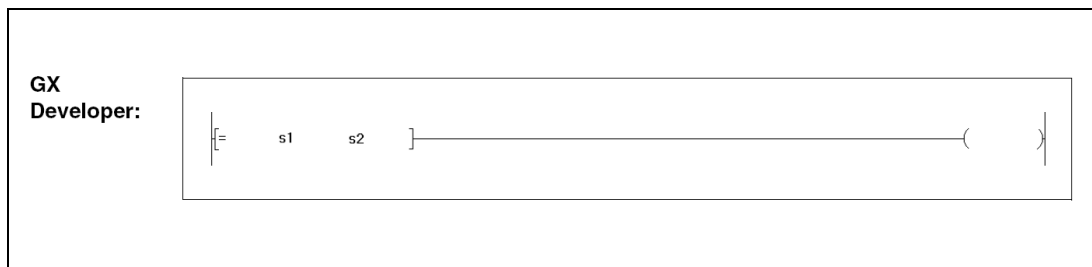
Le tabelle degli operandi sono seguite dalla rappresentazione del formato dell'istruzione in GX IEC Developer.

La figura seguente mostra da sinistra a destra la rappresentazione dell'istruzione LD_EQ_M nell'editor MELSEC (lista istruzioni MELSEC) e nell'editor IEC (schema a contatti e lista istruzioni IEC).



4.5.2 Rappresentazione in GX Developer

La rappresentazione del formato dell'istruzione in GX IEC Developer è seguita dalla rappresentazione del formato istruzione in GX Developer.



4.6 Variabili

La tabella delle variabili elenca tutte le variabili interne dell'istruzione.

Variabili	Dati impostati	Significato	Tipo dati	
			MELSEC	IEC
s		s+0: Misura numero di giri della tavola (solo uso interno).	BIN 16-bit	Array [1..3] di ANY16
		s+1: Numero di posizione		
		s+2: Numero del settore.		
n1		Numero settori (divisioni) della tavola (da 2 a 32767).		ANY16
n2		Numero settori lenti (da 0 a n1).		ANY16
d		d+0: Fase A segnale di ingresso	Bit	Array [1.0.8] di BOOL
		d+1: Fase B segnale di ingresso		
		d+2: Segnale d'ingresso rilevamento posizione di zero.		
		d+3: Segnale di uscita avanti veloce (solo uso interno).		
		d+4: Segnale di uscita avanti lento (solo uso interno).		
		d+5: Segnale di uscita Stop (solo uso interno).		
		d+6: Segnale di uscita indietro veloce (solo uso interno).		
		d+7: Segnale di uscita indietro lento (solo uso interno).		

La colonna "Significato" descrive le funzioni degli operandi ed elementi degli operandi. La colonna "Tipo dati" elenca i tipi dati degli operandi. Eventuali differenze fra i tipi dati negli editor MELSEC e IEC vengono descritte. Vedere anche i capitoli "3.4 Programmazione di variabili" e "3.5 Tipi di dati" per ulteriori dettagli sulle variabili.

4.7 Funzioni

La sezione "Funzioni" descrive in dettaglio le funzionalità dell'istruzione.

La figura seguente mostra la descrizione delle funzionalità dell'istruzione LDF/LDP.

Funzioni	<p>Inizio operazione impulsiva</p> <p>LDP fronte di salita</p> <p>LDF fronte di discesa</p> <p>Simili alle istruzioni LD e LD_I; queste istruzioni indicano contatti specificati da operandi a bit o a word. Il risultato dell'istruzione LDP è 1 se il bit indirizzato dell'operando passa da 0 a 1 (fronte di salita). Il risultato dell'istruzione LDF è 1 se il bit indirizzato passa da 1 a 0 (fronte di discesa). Come istruzione singola, LDP esegue la stessa funzione di una istruzione PLS e genera un impulso di uscita in corrispondenza del fronte di salita della condizione d'ingresso.</p> <p>Il programma di esempio sulla sinistra mostra uno schema a contatti che usa una istruzione LDP. L'esempio sulla destra non usa una istruzione LDP.</p>
-----------------	---

4.8 Note

La sezione "NOTE" riporta dettagli particolari, errori, e cause di malfunzionamento da considerare nell'uso dell'istruzione.

Le istruzioni MEP e MEF non funzionano occasionalmente nel modo appropriato se la conversione in impulso viene applicata a contatti indicizzati da un sottoprogramma o da una istruzione FOR/NEXT. In questi casi si devono utilizzare le istruzioni EGP/EGF.

Le istruzioni MEP/MEF considerano il risultato operazione immediatamente prima delle istruzioni stesse. Per questa ragione, dovrebbero essere utilizzate nella posizione adatta per una istruzione AND. Le istruzioni MEP e MEF non possono essere utilizzate nella posizione delle istruzioni LD o OR.

4.9 Errori di esecuzione

La descrizione degli errori di esecuzione si riferisce principalmente ai codici di errore della serie Q e del System Q (vedi "11.1 Tabella codici di errore; Q001, Q00 e Q01CPU" e "11.2 Tabella codici di errore; serie Q e System "). Per altre informazioni sui codici di errore della serie A, fare riferimento ai capitoli "11.3 Tabella codici di errore; serie A (tranne AnA e AnAS)" e "11.4 Tabella codici di errore; CPU AnA e AnAS".

La figura seguente mostra gli errori di esecuzione dell'istruzione DELTA/DELTA_P.

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

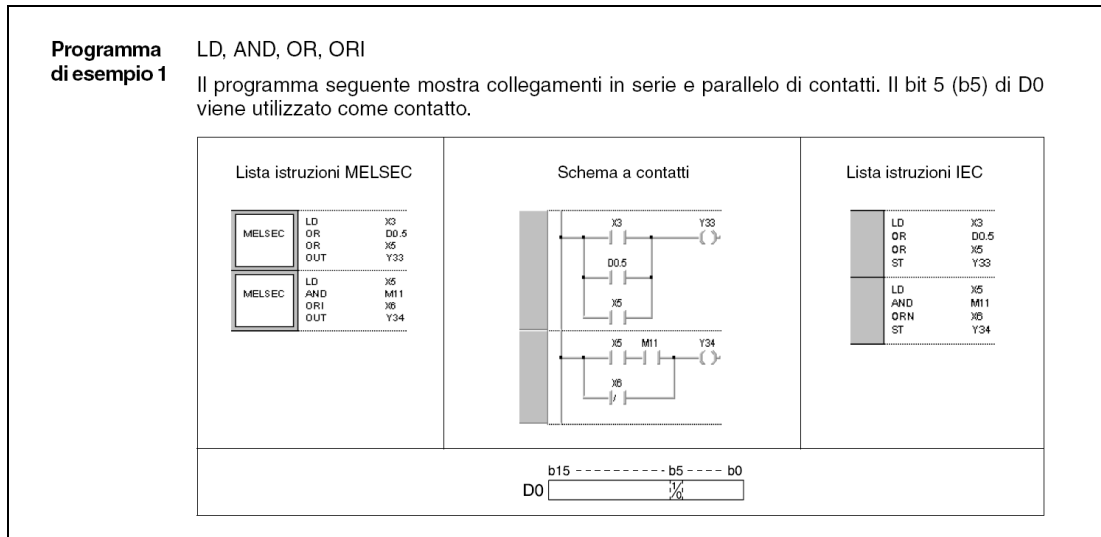
- Il numero dell'uscita indicata da d supera il campo delle uscite. (codice di errore: 4101).

4.10 Esempi di programma

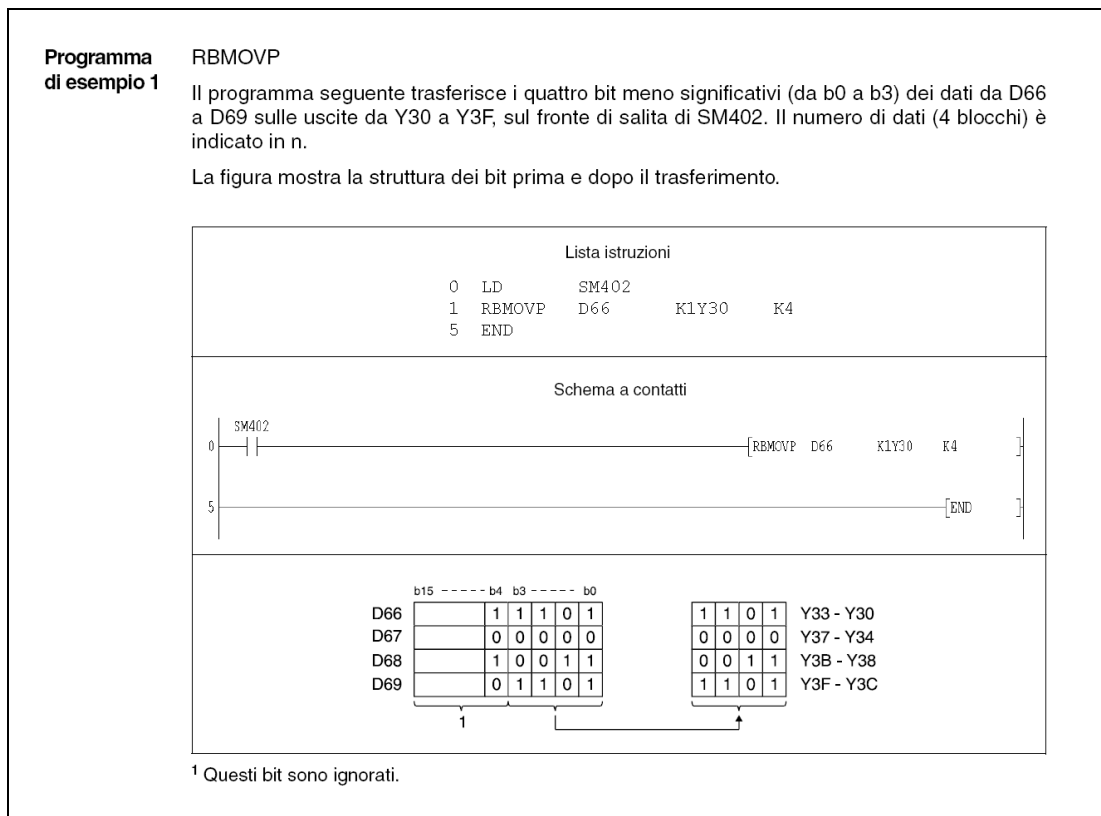
Gli esempi di programma forniti alla fine di ciascuna sezione contengono prevalentemente programmi per la serie Q e per il System Q.

I programmi di esempio usano il formato di rappresentazione della lista istruzioni MELSEC e lo schema a contatti e lista istruzioni IEC. Per una migliore comprensione, in molti casi sono state aggiunte delle illustrazioni.

La figura seguente mostra un esempio di programma per le istruzioni LD, AND, OR e ORI.



La figura seguente mostra un esempio di programma per l'istruzione RBMOVP. La rappresentazione dell'istruzione è quella di GX Developer.



5 Istruzioni di sequenza









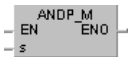
Oltre alle istruzioni convenzionali per la programmazione di contatti di ingresso e uscita, le istruzioni di sequenza comprendono anche comandi per salti di programma, istruzioni di collegamento blocchi e di scorrimento, controllo primario, terminazione programma e altre istruzioni. Queste sono le istruzioni fondamentali per la programmazione dei PLC delle serie MELSEC.

La tabella seguente mostra la suddivisione del set di istruzioni fondamentale:

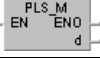
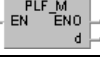
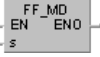
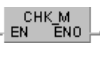
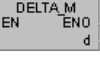
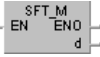
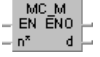
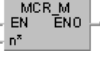
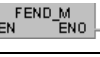
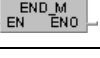
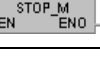
Istruzione	Significato
Istruzione di ingresso	Inizio operazione, collegamento contatti in serie e parallelo.
Istruzione di collegamento	Collegamento blocchi in serie e parallelo, memorizzazione ed elaborazione risultati operazione, inversione risultato operazione, conversione risultato operazione in impulsi, comando di relé comandati su fronte.
Istruzioni di uscita	Operandi a bit, contatti di contatori e temporizzatori, uscita, set, e reset di spie di errore, set e reset di operandi, uscita su fronte di salita e discesa, inversione uscita operandi a bit, generazione di impulsi.
Istruzioni di scorrimento	Scorrimento di operandi a bit
Istruzioni di controllo primario	Set e reset di singole parti di un programma.
Istruzione di terminazione	Fine di una parte di programma, fine sequenza e routine di programma.
Istruzioni miscellanee	Arresto sequenza di programma, nessuna operazione.

NOTA

La tabella seguente, oltre alle istruzioni MELSEC nei diversi editor, contiene anche le corrispondenti istruzioni IEC:

in editor MELSEC	Istruzione MELSEC			Istruzione IEC in Editor IEC
	in editor IEC		Schema a contatti	
	Lista istruzioni			
LD	—		—	LD
LDI	—		—	LDN
AND	—			AND
ANI	—			ANDN
OR	—		—	OR
ORI	—		—	ORN
LDP	LDP_M	—	—	—
LDF	LDF_M	—	—	—
ANDP	ANDP_M	—		—

Istruzione MELSEC				Istruzione IEC in Editor IEC
in editor MELSEC	in editor IEC			
	Lista istruzioni	Schema a contatti		
ANDF	ANDF_M	—		—
ORP	ORP_M	—		—
ORF	ORF_M	—		—
ANB	—		—	AND (...)
ORB	—		—	OR (...)
MPS	MPS_M			—
MRD	MRD_M			—
MPP	MPP_M			—
INV	INV_M			NOT
MEP	MEP_M	—		—
MEF	MEF_M	—		—
EGP	EGP_M	—		—
EGF	EGF_M	—		—
OUT	OUT_M			ST
OUT T	TIMER_M	—		—
OUT TH	TIMER_H_M	—		—
OUT C	COUNTER_M	—		—
SET	SET_M			S
RST	RST_M			R

Istruzione MELSEC				Istruzione IEC in Editor IEC
in editor MELSEC	in editor IEC			
	Lista istruzioni	Schema a contatti		
PLS	PLS_M	—		R_TRIG ● ¹
PLF	PLF_M	—		R_TRIG ● ¹
FF	FF_M	—		—
CHK	CHK_M	—		—
DELTA	DELTA_M	—		—
SFT	SFT_M	—		SHL/SHR
MC	MC_M	—		—
MCR	MCR_M	—		—
FEND	FEND_M	—		● ²
END	END_M	—		● ²
STOP	STOP_M	—		—
NOP	—	—	—	—

¹ Questi sono blocchi funzione IEC.

² FEND e END sono inserite automaticamente da GX Developer e GX IEC Developer.

5.1 Istruzioni di ingresso

5.1.1 LD, LDI, AND, ANI, OR, ORI

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

Operandi utilizzabili														Definizione cifra	Numero di passi	Indice	Segnale di riporto	Segnale di errore							
Operandi a bit				Operandi a word (16-bit)				Costanti		Puntatore		Livello													
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H (16#)	P	I	N	M9012	M9010 M9011			
●	●	●	●	●	●	●	●	●															1		

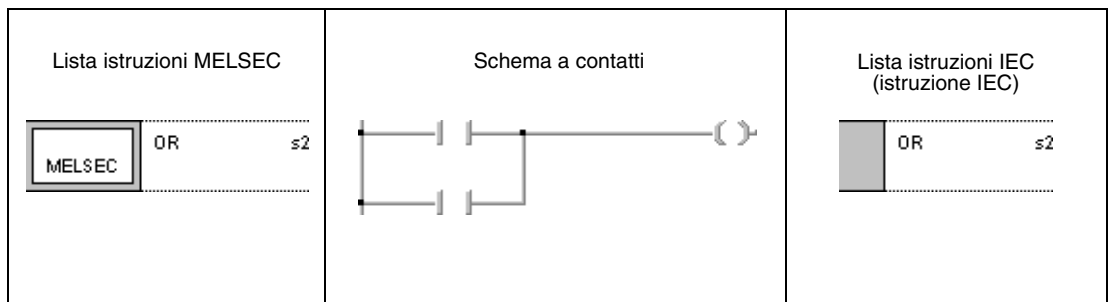
¹ Consultare la sezione "Programmazione con CPU AnA, AnAS e AnU" del manuale di programmazione per il corrispondente numero di passi.

Operandi MELSEC Q

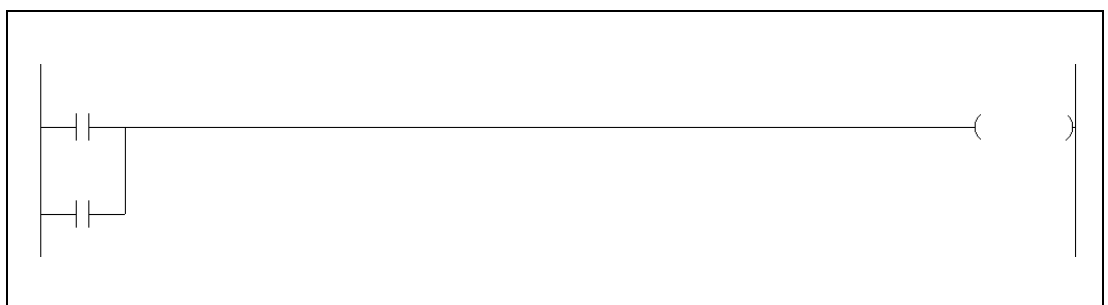
s	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operando interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
●	●	●	●	●	●	—	—	●	—	1 ¹	

¹ Il numero dei passi varia se:
 - si usa un operando interno o un file registri da R0 a R32767: 1 passo
 - si usano ingressi ad accesso diretto (DX): 2 passi
 - si usano altri operandi: 3 passi

GX IEC Developer:



GX Developer:



Variabili

Dati impostati	Significato	Tipo dati
s	Operandi usati come collegamenti.	bit

Funzioni **Inizio operazione****LD** **Carica (contatto normalmente aperto)****LDI** **Carica invertito (contatto normalmente chiuso)**

Ogni operazione inizia con una istruzione LD (**LoaD**) o LDI (**LoaD Inverse**). L'istruzione LD indica un contatto NO (normalmente aperto) mentre l'istruzione LDI indica un contatto NC (normalmente chiuso). L'operando indicato dall'istruzione è la condizione di ingresso (risultato operazione) per l'istruzione che segue.

Collegamento in serie**AND** **di contatti norm. aperti****ANI** **di contatti norm. chiusi**

I contatti vengono collegati in serie tramite una istruzione AND, come contatto NO, o tramite una istruzione ANI come contatto NC.

Entrambi i comandi sono collegamenti logici e non devono essere programmati all'inizio di una operazione.

Collegamento parallelo**OR** **di contatti norm. aperti****ORI** **di contatti norm. chiusi**

Il collegamento in parallelo di contatti viene effettuato tramite una istruzione OR come contatto NO, oppure tramite una istruzione ORI come contatto NC. L'operando indicato dall'istruzione imposta la condizione di ingresso per l'istruzione che segue.

Entrambi i comandi sono collegamenti logici e non devono essere programmati all'inizio di una operazione.

NOTA

Gli operandi indicati dalle istruzioni possono essere anche operandi a word. In questo caso la condizione di un bit specifico viene letta come contatto (solo serie Q e System Q).

Gli operandi a word sono indicati in codice esadecimale. Ad esempio, il bit b11 di D0 viene indicato come D0.0B (solo serie Q e System Q).

Per ulteriori informazioni sull'indirizzamento a bit di operandi a word, consultare il capitolo "Configurazione delle istruzioni" (solo serie e System Q).

Programma di esempio 1

LD, AND, OR, ORI

Il programma seguente mostra collegamenti in serie e parallelo di contatti. Il bit 5 (b5) di D0 viene utilizzato come contatto.

<p style="text-align: center;">Lista istruzioni MELSEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="width: 15%;"></td><td style="width: 15%;">LD</td><td style="width: 15%;">X3</td></tr> <tr><td style="border: 1px solid black;">MELSEC</td><td>OR</td><td>D0.5</td></tr> <tr><td></td><td>OR</td><td>X5</td></tr> <tr><td></td><td>OUT</td><td>Y33</td></tr> <tr><td colspan="3"> </td></tr> <tr><td style="border: 1px solid black;">MELSEC</td><td>LD</td><td>X5</td></tr> <tr><td></td><td>AND</td><td>M11</td></tr> <tr><td></td><td>ORI</td><td>X6</td></tr> <tr><td></td><td>OUT</td><td>Y34</td></tr> </table>		LD	X3	MELSEC	OR	D0.5		OR	X5		OUT	Y33				MELSEC	LD	X5		AND	M11		ORI	X6		OUT	Y34	<p>Schema a contatti</p>	<p style="text-align: center;">Lista istruzioni IEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="width: 15%;"></td><td style="width: 15%;">LD</td><td style="width: 15%;">X3</td></tr> <tr><td style="border: 1px solid black;"></td><td>OR</td><td>D0.5</td></tr> <tr><td></td><td>OR</td><td>X5</td></tr> <tr><td></td><td>ST</td><td>Y33</td></tr> <tr><td colspan="3"> </td></tr> <tr><td style="border: 1px solid black;"></td><td>LD</td><td>X5</td></tr> <tr><td></td><td>AND</td><td>M11</td></tr> <tr><td></td><td>ORN</td><td>X6</td></tr> <tr><td></td><td>ST</td><td>Y34</td></tr> </table>		LD	X3		OR	D0.5		OR	X5		ST	Y33					LD	X5		AND	M11		ORN	X6		ST	Y34
	LD	X3																																																						
MELSEC	OR	D0.5																																																						
	OR	X5																																																						
	OUT	Y33																																																						
MELSEC	LD	X5																																																						
	AND	M11																																																						
	ORI	X6																																																						
	OUT	Y34																																																						
	LD	X3																																																						
	OR	D0.5																																																						
	OR	X5																																																						
	ST	Y33																																																						
	LD	X5																																																						
	AND	M11																																																						
	ORN	X6																																																						
	ST	Y34																																																						
<p>b15 ----- b5 ---- b0</p> <p>D0 1/0</p>																																																								

Programma di esempio 2

LD, LDI, AND, ANI, OR

Il programma seguente mostra collegamenti combinati. Alcuni contatti sono collegati tramite istruzioni ORB e ANB. I bit (b1 e b4) di D6 sono utilizzati come contatti.

<p style="text-align: center;">Lista istruzioni MELSEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="width: 15%;"></td><td style="width: 15%;">LD</td><td style="width: 15%;">X3</td></tr> <tr><td style="border: 1px solid black;">MELSEC</td><td>AND</td><td>D6.1</td></tr> <tr><td></td><td>LDI</td><td>D6.4</td></tr> <tr><td></td><td>ANI</td><td>X7</td></tr> <tr><td></td><td>ORB</td><td></td></tr> <tr><td></td><td>ANI</td><td>M9</td></tr> <tr><td></td><td>OUT</td><td>Y33</td></tr> <tr><td colspan="3"> </td></tr> <tr><td style="border: 1px solid black;">MELSEC</td><td>LD</td><td>X5</td></tr> <tr><td></td><td>LD</td><td>M8</td></tr> <tr><td></td><td>OR</td><td>M9</td></tr> <tr><td></td><td>ANB</td><td></td></tr> <tr><td></td><td>ANI</td><td>M11</td></tr> <tr><td></td><td>OUT</td><td>Y34</td></tr> </table>		LD	X3	MELSEC	AND	D6.1		LDI	D6.4		ANI	X7		ORB			ANI	M9		OUT	Y33				MELSEC	LD	X5		LD	M8		OR	M9		ANB			ANI	M11		OUT	Y34	<p>Schema a contatti</p>	<p style="text-align: center;">Lista istruzioni IEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="width: 15%;"></td><td style="width: 15%;">LD</td><td style="width: 15%;">X3</td></tr> <tr><td style="border: 1px solid black;"></td><td>AND</td><td>D6.1</td></tr> <tr><td></td><td>ORL</td><td>D6.4</td></tr> <tr><td></td><td>ANDN</td><td>X7</td></tr> <tr><td></td><td>NOT</td><td></td></tr> <tr><td></td><td>ANDN</td><td>M9</td></tr> <tr><td></td><td>)</td><td></td></tr> <tr><td></td><td>ANDN</td><td>M8</td></tr> <tr><td></td><td>ST</td><td>Y33</td></tr> <tr><td colspan="3"> </td></tr> <tr><td style="border: 1px solid black;"></td><td>LD</td><td>X5</td></tr> <tr><td></td><td>AND</td><td>M8</td></tr> <tr><td></td><td>OR</td><td>M9</td></tr> <tr><td></td><td>)</td><td></td></tr> <tr><td></td><td>ANDN</td><td>M11</td></tr> <tr><td></td><td>ST</td><td>Y34</td></tr> </table>		LD	X3		AND	D6.1		ORL	D6.4		ANDN	X7		NOT			ANDN	M9)			ANDN	M8		ST	Y33					LD	X5		AND	M8		OR	M9)			ANDN	M11		ST	Y34
	LD	X3																																																																																										
MELSEC	AND	D6.1																																																																																										
	LDI	D6.4																																																																																										
	ANI	X7																																																																																										
	ORB																																																																																											
	ANI	M9																																																																																										
	OUT	Y33																																																																																										
MELSEC	LD	X5																																																																																										
	LD	M8																																																																																										
	OR	M9																																																																																										
	ANB																																																																																											
	ANI	M11																																																																																										
	OUT	Y34																																																																																										
	LD	X3																																																																																										
	AND	D6.1																																																																																										
	ORL	D6.4																																																																																										
	ANDN	X7																																																																																										
	NOT																																																																																											
	ANDN	M9																																																																																										
)																																																																																											
	ANDN	M8																																																																																										
	ST	Y33																																																																																										
	LD	X5																																																																																										
	AND	M8																																																																																										
	OR	M9																																																																																										
)																																																																																											
	ANDN	M11																																																																																										
	ST	Y34																																																																																										
<p>b15 ----- b4 -- b1b0</p> <p>D6 1/0 1/0</p>																																																																																												

Programma di esempio 3

LD, AND, ANI

Il programma che segue invia in uscita i risultati delle operazioni sugli operandi da Y35 a Y37.

<p style="text-align: center;">Lista istruzioni MELSEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="width: 15%;"></td><td style="width: 15%;">LD</td><td style="width: 15%;">X5</td></tr> <tr><td style="border: 1px solid black;">MELSEC</td><td>OUT</td><td>Y35</td></tr> <tr><td></td><td>AND</td><td>X8</td></tr> <tr><td></td><td>OUT</td><td>Y36</td></tr> <tr><td></td><td>ANI</td><td>X9</td></tr> <tr><td></td><td>OUT</td><td>Y37</td></tr> </table>		LD	X5	MELSEC	OUT	Y35		AND	X8		OUT	Y36		ANI	X9		OUT	Y37	<p>Schema a contatti</p>	<p style="text-align: center;">Lista istruzioni IEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="width: 15%;"></td><td style="width: 15%;">LD</td><td style="width: 15%;">X5</td></tr> <tr><td style="border: 1px solid black;"></td><td>ST</td><td>Y35</td></tr> <tr><td></td><td>AND</td><td>X8</td></tr> <tr><td></td><td>ST</td><td>Y36</td></tr> <tr><td></td><td>ANDN</td><td>X9</td></tr> <tr><td></td><td>ST</td><td>Y37</td></tr> </table>		LD	X5		ST	Y35		AND	X8		ST	Y36		ANDN	X9		ST	Y37
	LD	X5																																				
MELSEC	OUT	Y35																																				
	AND	X8																																				
	OUT	Y36																																				
	ANI	X9																																				
	OUT	Y37																																				
	LD	X5																																				
	ST	Y35																																				
	AND	X8																																				
	ST	Y36																																				
	ANDN	X9																																				
	ST	Y37																																				

5.1.2 LDP, LDF, ANDP, ANDF, ORP, ORF

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●


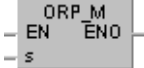
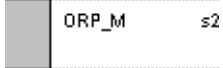
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro DX
	Bit	Word		Bit	Word						
s	●	●	●	●	●	—	—	●	—	2 ¹ ●	

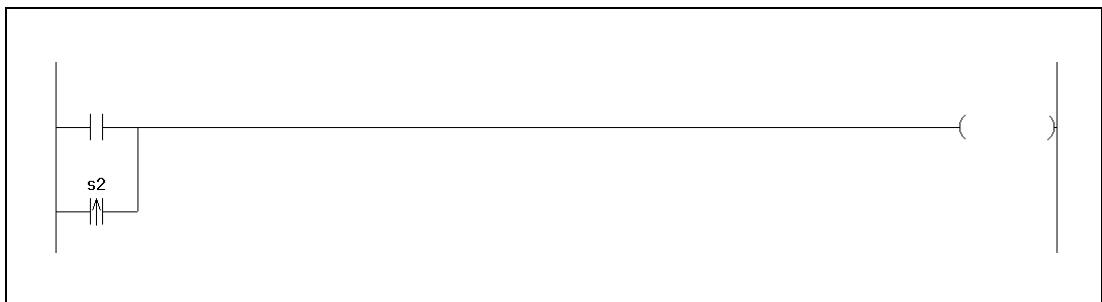
¹ Il numero dei passi varia se:

- si usa un operando interno o un file registri da R0 a R32767: 2 passi
- si usano ingressi ad accesso diretto (DX): 3 passi
- si usano altri operandi: 4 passi

GX IEC Developer:

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
		

GX Developer:



Variabili

Dati impostati	Significato	Tipo dati
s	Operandi usati come collegamenti.	bit

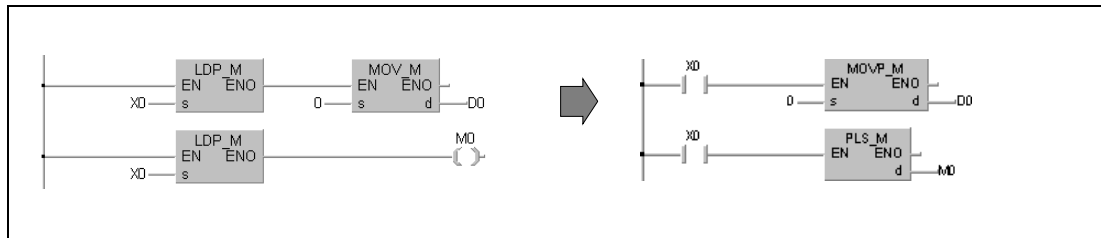
Funzioni Inizio operazione impulsiva

LDP fronte di salita

LDF fronte di discesa

Simili alle istruzioni LD e LDI; queste istruzioni indicano contatti specificati da operandi a bit o a word. Il risultato dell'istruzione LDP è 1 se il bit indirizzato dell'operando passa da 0 a 1 (fronte di salita). Il risultato dell'istruzione LDF è 1 se il bit indirizzato passa da 1 a 0 (fronte di discesa). Come istruzione singola, LDP esegue la stessa funzione di una istruzione PLS e genera un impulso di uscita in corrispondenza del fronte di salita della condizione d'ingresso.

Il programma di esempio sulla sinistra mostra uno schema a contatti che usa una istruzione LDP. L'esempio sulla destra non usa una istruzione LDP.



Collegamento impulso in serie

ANDP fronte di salita

ANDF fronte di discesa

L'istruzione ANDP collega un contatto in serie con un contatto specificato da un operando a bit o a word. Questo contatto assume la condizione 1 se il bit dell'operando indirizzato passa da 0 a 1.

Usando l'istruzione ANDF, il contatto specificato assume la condizione 1 se il bit dell'operando indirizzato passa da 1 a 0.

Collegamento impulso in parallelo

ORP fronte di salita

ORF fronte di discesa

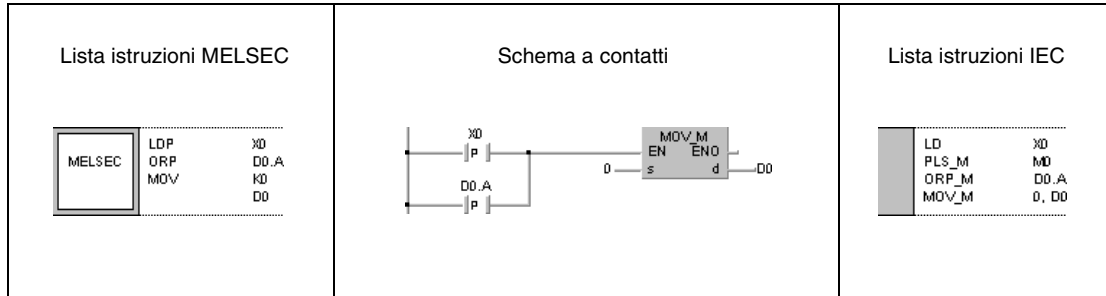
L'istruzione ORP collega un contatto in parallelo a un contatto specificato da un operando a bit o a word. Questo contatto assume la condizione 1 se il bit dell'operando indirizzato passa da 0 a 1.

Usando l'istruzione ORF, il contatto specificato assume la condizione 1 se il bit dell'operando indirizzato passa da 1 a 0.

Operando specificato dalla istruzione ANDP/ORP	Risultato della istruzione ANDP/ORP	Operando specificato dalla istruzione ANDF/ORF	Risultato della istruzione ANDF/ORF
Operando a bit o a word		Operando a bit o a word	
0 → 1	1	0 → 1	0
0	0	0	
1		1	
1 → 0		1 → 0	1

NOTA *Gli operandi a word sono indicati in codice esadecimale. Ad esempio il bit b11 di D0 viene indicato come D0.0B.*

Esempio di programma ORP
 Con il fronte di salita di X0 o impostando a 1 (fronte di salita) il bit 10 (b10) nel registro dati D0, il programma che segue esegue una istruzione MOV.



Funzioni **Collegamento in serie di blocco di contatti****ANB** **Collegamento in serie di un blocco**

L'istruzione ANB (AND di blocco) collega in serie due o più blocchi collegati in parallelo e fornisce un risultato operazione per l'istruzione successiva.

Se vengono collegati in serie più di due blocchi, è necessario programmare una istruzione ANB dopo ciascun blocco in parallelo.

Il collegamento ANB è una istruzione indipendente e non richiede alcun operando.

L'istruzione ANB può essere programmata liberamente all'interno di un programma.

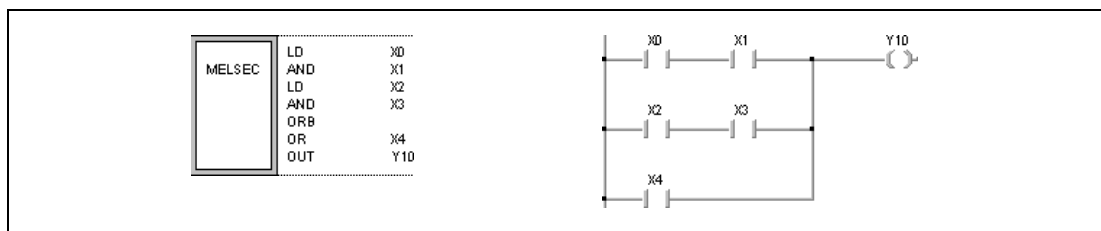
Se vengono collegati consecutivamente più di due blocchi, il numero di istruzioni ANB è limitato a 15 (= 16 blocchi) con una CPU QnA, AnA, AnAS o AnU, oppure a 7 (= 8 blocchi) con tutte le altre CPU. Il superamento di questi limiti provoca malfunzionamenti.

Collegamento in parallelo di blocco a contatti**ORB** **Collegamento in parallelo di un blocco**

L'istruzione ORB (OR di blocco) collega in parallelo due o più blocchi collegati in serie e fornisce un risultato operazione per l'istruzione successiva.

Se vengono collegati in parallelo più di due blocchi, è necessario programmare una istruzione ORB dopo ciascun blocco in serie.

Per il collegamento in parallelo di blocchi che contengono un solo contatto, si devono usare le istruzioni OR o ORI.



Il collegamento ORB è una istruzione indipendente e non richiede alcun operando.

L'istruzione ORB può essere programmata liberamente all'interno di un programma.

Se vengono collegati consecutivamente più di due blocchi, il numero di istruzioni ORB è limitato a 15 (= 16 blocchi) con una CPU QnA, AnA, AnAS o AnU, oppure a 7 (= 8 blocchi) con tutte le altre CPU. Il superamento di questi limiti provoca malfunzionamenti.

Esempio di programma ANB, ORB

Il programma seguente collega in serie il blocco in parallelo formato da X0 e X2 con il blocco in parallelo formato da X1 e X3. Il risultato viene collegato in parallelo con il collegamento in serie di X4 e X5.

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC																																
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center; width: 100px;">MELSEC</td> <td style="width: 100px;">LD X0</td> <td style="width: 100px;">OR X2</td> </tr> <tr> <td></td> <td>LD X1</td> <td>OR X3</td> </tr> <tr> <td></td> <td>ANB</td> <td></td> </tr> <tr> <td></td> <td>LD X4</td> <td>AND X5</td> </tr> <tr> <td></td> <td>ORB</td> <td></td> </tr> <tr> <td></td> <td>OUT M0</td> <td></td> </tr> </table>	MELSEC	LD X0	OR X2		LD X1	OR X3		ANB			LD X4	AND X5		ORB			OUT M0		<p>Schema a contatti</p>	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 100px;">LD X0</td> <td style="width: 100px;">OR X2</td> </tr> <tr> <td>AND(</td> <td>X1</td> </tr> <tr> <td>)</td> <td>X3</td> </tr> <tr> <td>OR(</td> <td>X4</td> </tr> <tr> <td>)</td> <td>X5</td> </tr> <tr> <td>AND</td> <td>M0</td> </tr> <tr> <td>ST</td> <td></td> </tr> </table>	LD X0	OR X2	AND(X1)	X3	OR(X4)	X5	AND	M0	ST	
MELSEC	LD X0	OR X2																																
	LD X1	OR X3																																
	ANB																																	
	LD X4	AND X5																																
	ORB																																	
	OUT M0																																	
LD X0	OR X2																																	
AND(X1																																	
)	X3																																	
OR(X4																																	
)	X5																																	
AND	M0																																	
ST																																		

5.2.2 MPS, MRD, MPP

NOTA Queste istruzioni non dovrebbero essere utilizzate con gli editor IEC.

CPU

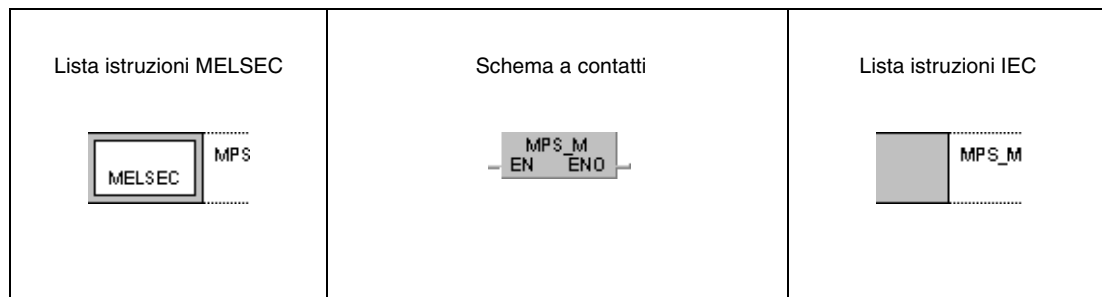
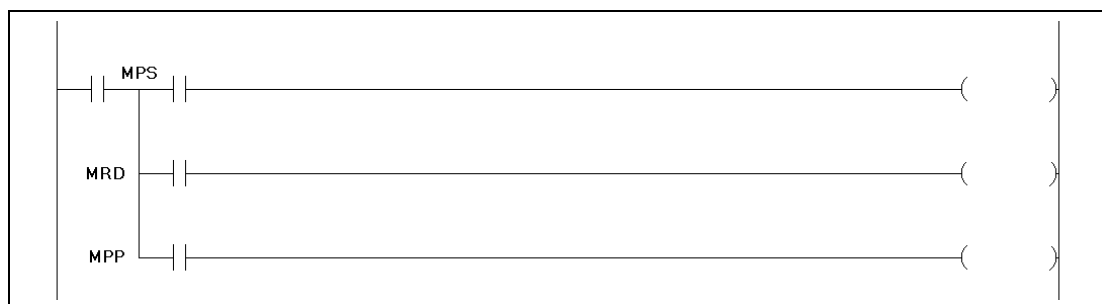
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Segnale di errore	Segnale di errore					
Operandi a bit						Operandi a word (16-bit)						Costanti		Puntatore							Livello				
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V				K	H (16#)	P	I	N	M9012	M9010 M9011
																							1		

Operandi MELSEC Q

Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word							
—	—	—	—	—	—	—	—	—	—	—	1

GX IEC Developer:**GX Developer:****Variabili**

Dati impostati	Significato	Tipo dati
—	—	—

Funzioni Elaborazione risultato operazione**MPS memorizza risultato operazione (inserimento in memoria)**

L'istruzione MPS memorizza il risultato dell'operazione precedente.

Usando una CPU QnA, AnA, AnAS o AnU, si possono programmare fino a 16 istruzioni MPS consecutive per ogni segmento. Con tutte le altre CPU il limite è di 12 istruzioni. Se una istruzione MPP viene inserita fra due istruzioni MPS, il limite viene ridotto di uno.

MRD lettura risultato operazione (lettura da memoria)

L'istruzione MRD legge i risultati delle operazioni memorizzati con l'istruzione MPS. L'esecuzione dell'operazione successiva dipende dal risultato della lettura.

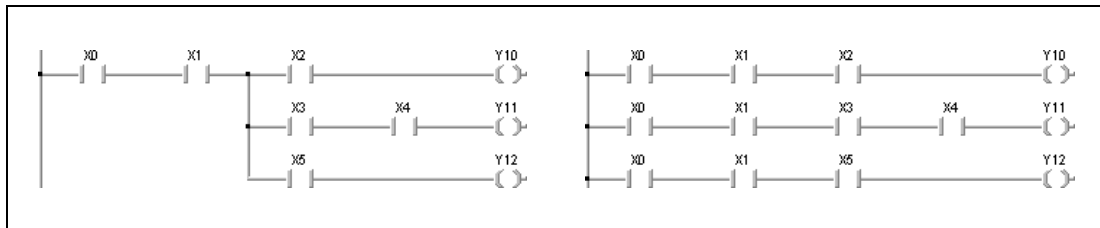
MPP lettura e cancellazione risultato operazione (estrazione da memoria)

L'istruzione MPP legge i risultati delle operazioni memorizzati con l'istruzione MPS. L'esecuzione dell'operazione successiva dipende dal risultato della lettura. Il risultato viene poi azzerato.

Le istruzioni MPS, MRD e MPP sono istruzioni indipendenti che non richiedono alcun operando.

Nella programmazione a contatti, le istruzioni MPS, MRD e MPP non vengono visualizzate esplicitamente. Il tipo di collegamento MPS, MRD o MPP dipende dalla struttura dello schema a contatti.

L'esempio sulla sinistra mostra uno schema a contatti che applica le istruzioni MPS, MRD o MPP. L'esempio sulla sinistra mostra uno schema a contatti senza le istruzioni MPS, MRD o MPP.



Il numero di istruzioni MPS presenti in un programma deve corrispondere a quello delle istruzioni MPP.

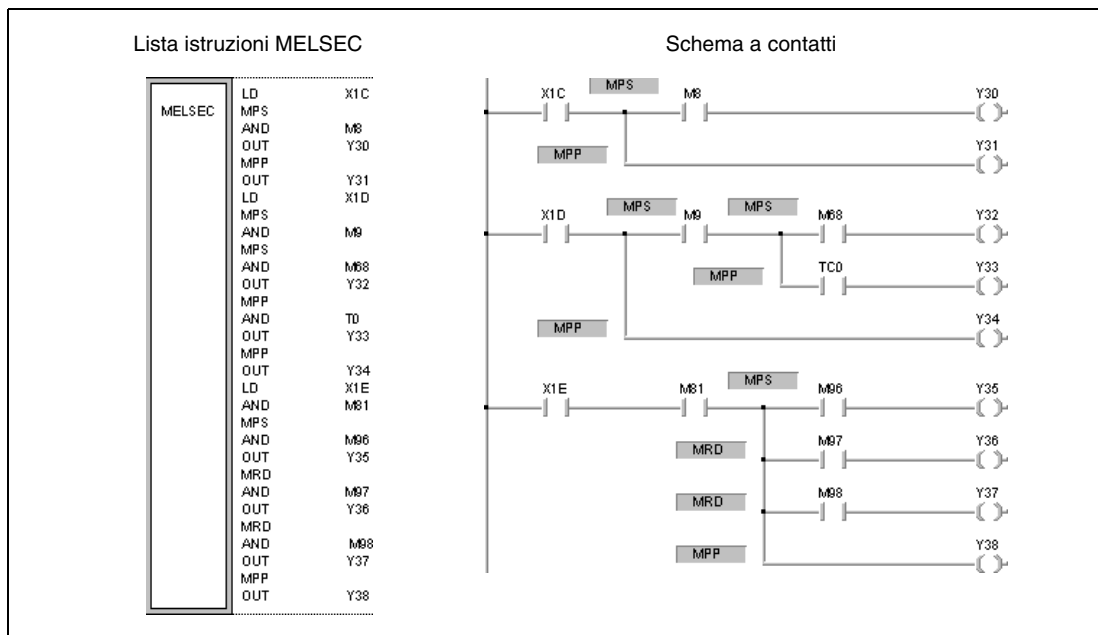
Se il numero delle istruzioni MPS supera quello delle istruzioni MPP, viene inserito un NOP invece di una istruzione MPP e il resto del programma viene modificato di conseguenza.

Se il numero delle istruzioni MPP supera quello delle istruzioni MPS, l'elaborazione del programma viene sospesa. In questo caso l'esecuzione del programma non può procedere e la CPU restituisce un codice di errore.

Programma di esempio 1

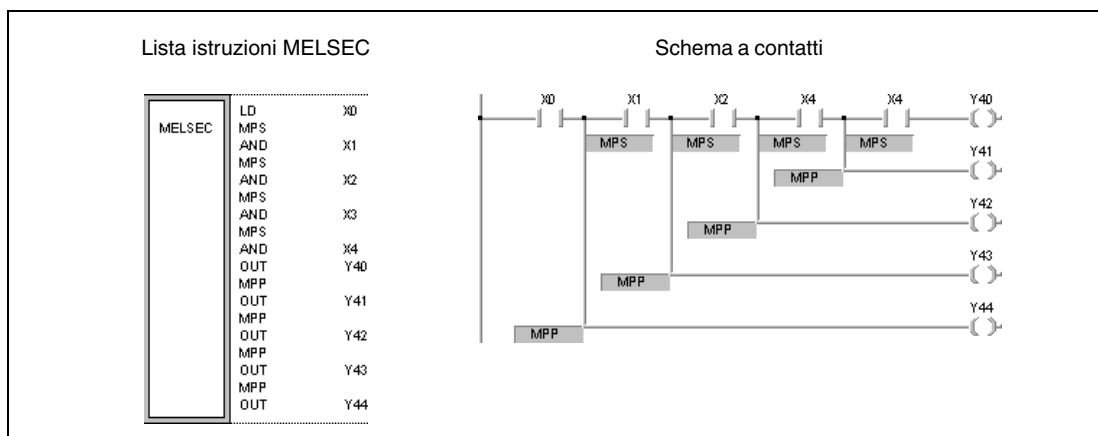
MPS, MRD, MPP

Il programma che segue mostra l'uso delle istruzioni per la programmazione di collegamenti combinati.

**Programma di esempio 2**

MPS, MRD, MPP

Il programma seguente mostra la programmazione di istruzioni che emettono risultati temporanei di un collegamento in serie.



5.2.3 INV


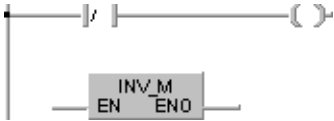
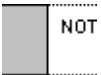
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

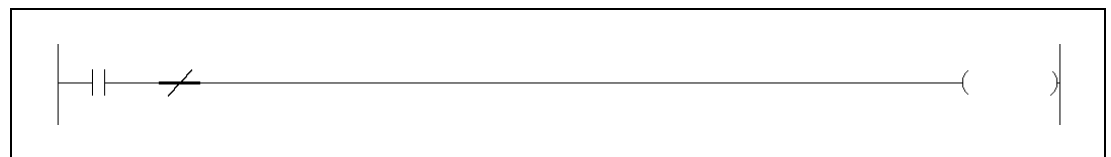
Operandi MELSEC Q

Operandi utilizzabili									Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti K, H (16#)	Altro		
Bit	Word		Bit	Word				U		
—	—	—	—	—	—	—	—	—	1	

GX IEC Developer:

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC (istruzione IEC)</p> 
--	---	--

GX Developer:



Variabili

Dati impostati	Significato	Tipo dati
—	—	—

Funzioni **Inversione risultato operazione****INV** **istruzione di inversione**

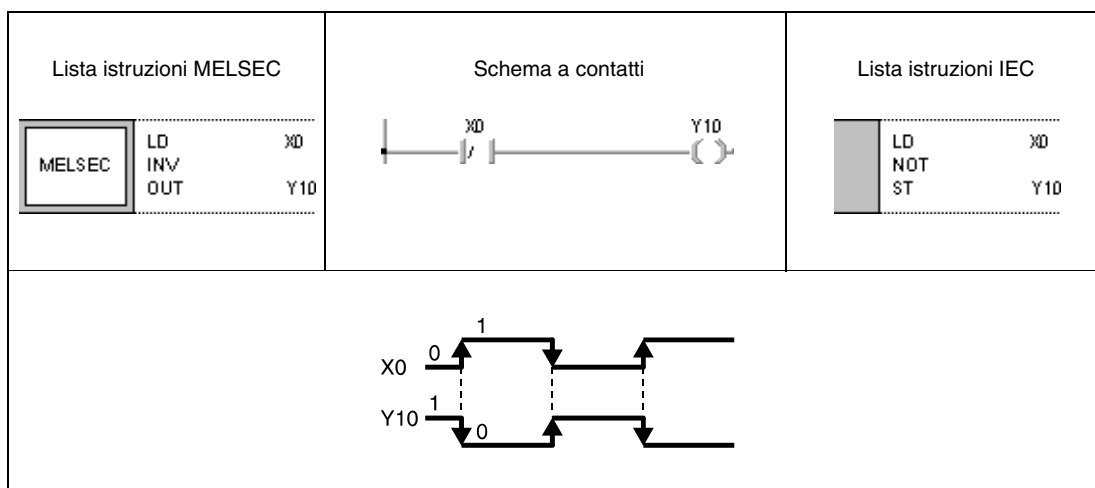
L'istruzione INV inverte il risultato dell'operazione precedente.

Se il risultato precedente all'istruzione INV era 1, diventa 0

Se il risultato precedente all'istruzione INV era 0, diventa 1.

Esempio di programma

Il programma che segue inverte lo stato di X0 ed invia in uscita il segnale invertito su Y10.



5.2.4 MEP, MEF

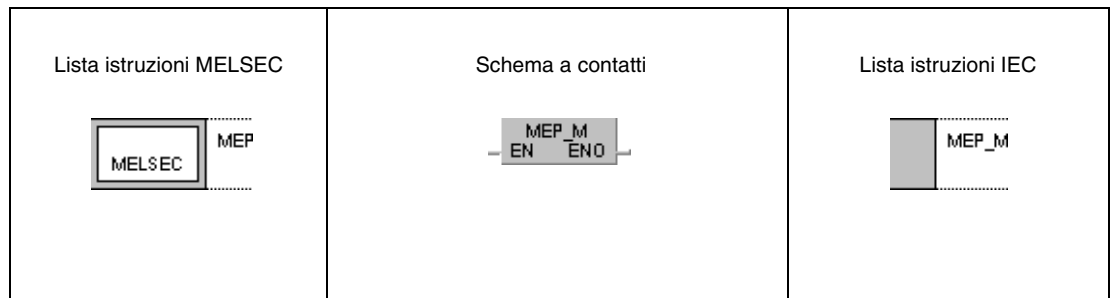
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

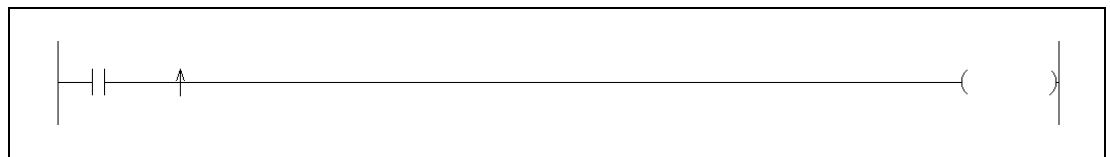
Operandi MELSEC Q

Operandi utilizzabili									Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti K, H (16#)	Altro U		
Bit	Word		Bit	Word						
—	—	—	—	—	—	—	—	—	—	1

GX IEC Developer:



GX Developer:



Variabili

Dati impostati	Significato	Tipo dati
—	—	—

Funzioni **Conversione risultato operazione in impulso****MEP generazione impulso su fronte di salita del risultato operazione**

L'istruzione MEP viene usata nei casi in cui le istruzioni utilizzate non possono emettere i risultati delle operazioni usate sotto forma di impulso. L'istruzione MEP viene inserita dopo le altre istruzioni e genera un impulso di uscita quando il suo segnale d'ingresso passa da 0 a 1 (fronte di salita). L'impulso successivo viene generato quando si incontra nuovamente un fronte di salita sul segnale d'ingresso.

MEF generazione impulso su fronte di discesa del risultato operazione

L'istruzione MEF viene usata nei casi in cui le istruzioni utilizzate non possono emettere i risultati delle operazioni usate sotto forma di impulso. L'istruzione MEF viene inserita dopo le altre istruzioni e genera un impulso di uscita quando il suo segnale d'ingresso passa da 1 a 0 (fronte di discesa). L'impulso successivo viene generato quando si incontra nuovamente un fronte di discesa sul segnale d'ingresso.

Queste due istruzioni sono particolarmente adatte per il collegamento di contatti multipli. Ad esempio, diversi contatti NO (normalmente aperti) collegati in serie manterrebbero il risultato dell'operazione a 1 se tutti i contatti sono chiusi. Se un relé viene eccitato dal risultato di questa operazione, non è possibile diseccitarlo. Con una istruzione MEP collegata in serie ai contatti NO, il relé può essere diseccitato perché l'istruzione emette un impulso solo quando il risultato del collegamento in serie passa da 0 a 1.

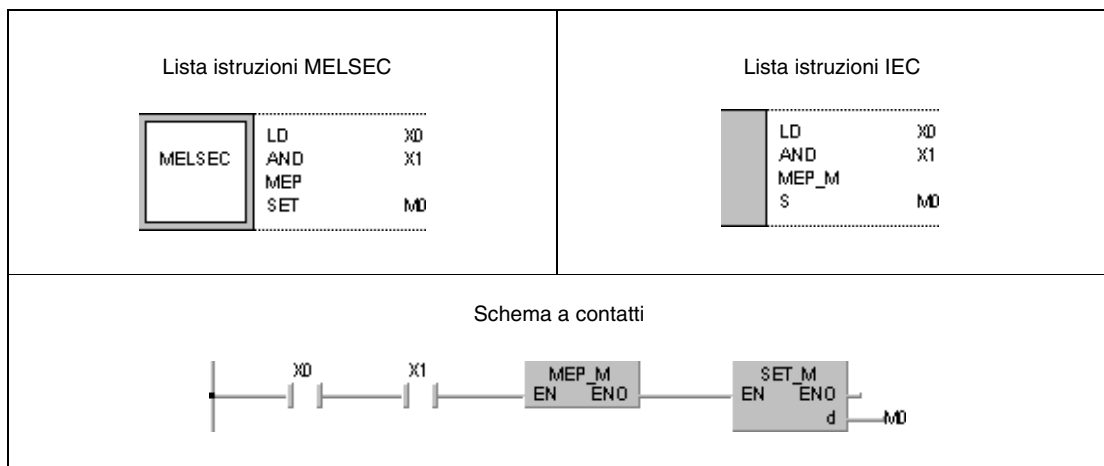


Le istruzioni MEP e MEF non funzionano occasionalmente nel modo appropriato se la conversione in impulso viene applicata a contatti indicizzati da un sottoprogramma o da una istruzione FOR/NEXT. In questi casi si devono utilizzare le istruzioni EGP/EGF.

Le istruzioni MEP/MEF considerano il risultato operazione immediatamente prima delle istruzioni stesse. Per questa ragione, dovrebbero essere utilizzate nella posizione adatta per una istruzione AND. Le istruzioni MEP e MEF non possono essere utilizzate nella posizione delle istruzioni LD o OR.

Esempio di programma**MEP**

Con il fronte di salita del risultato del collegamento in serie di X0 e X1, il programma che segue imposta il relé M0.



5.2.5 EGP, EGF

CPU

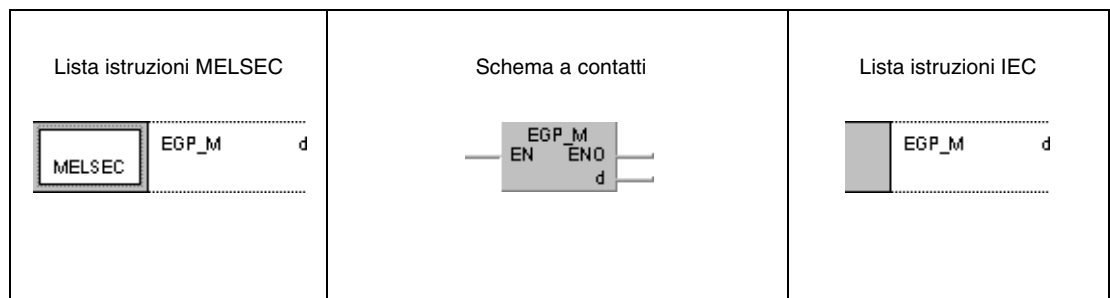
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi MELSEC Q

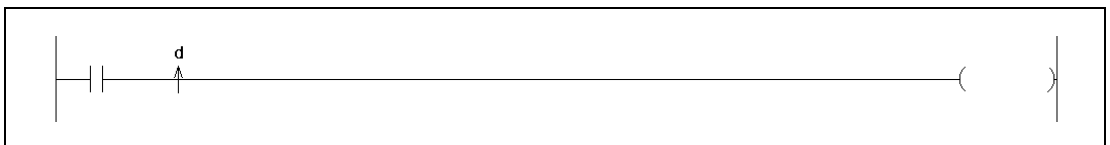
Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word				U			
d	● ¹	—	—	—	—	—	—	—	—	—	1

¹ solo V

GX IEC Developer:



GX Developer:



Variabili

Dati impostati	Significato	Tipo dati
d	Relé impulsivo, che memorizza il risultato dell'operazione.	bit (solo V)

Funzioni

Attivazione di relé comandati su fronte.

EGP attivazione di un relé con il fronte di salita di un risultato operazione

L'istruzione EGP imposta il relé impulsivo (V) in funzione del risultato operazione dell'istruzione precedente. Se il risultato passa da 0 a 1, il relé impulsivo viene eccitato. In tutte le altre condizioni dell'istruzione EGP, ad esempio nel passaggio da 1 a 0, oppure se permangono le condizioni 1 o 0, il relé impulsivo non viene eccitato.

EGF attivazione di un relé impulsivo con il fronte di discesa di un risultato operazione

L'istruzione EGF imposta il relé impulsivo (V) in funzione del risultato operazione dell'istruzione precedente. Se il risultato passa da 1 a 0, il relé impulsivo viene eccitato. In tutte le altre condizioni dell'istruzione EGF, ad esempio nel passaggio da 0 a 1, oppure se permangono le condizioni 0 o 1, il relé impulsivo non viene eccitato.

Le istruzioni EGP e EGF sono utilizzate in sottoprogrammi o all'interno delle istruzioni FOR/NEXT e nelle operazioni di indicizzazione tramite registro indice.

Le istruzioni EGP e EGF possono essere utilizzate come una istruzione AND.

5.3 Istruzioni di uscita

5.3.1 OUT

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

d	Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Segnale di errore	Segnale di errore							
	Operandi a bit						Operandi a word (16-bit)						Costanti		Puntatore							Livello						
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V						K	H (16#)	P	I	N	N	M9012
	●	●	●	●	●																				● ¹	● ²		

¹ In generale, 1 passo. Eccezione: 3 passi per la programmazione di relé interni o spie di errore come operando per l'istruzione OUT. Consultare la sezione "Programmazione con CPU AnA, AnAS e AnU" del manuale di programmazione per il corrispondente numero di passi.

² L'indicizzazione è disponibile solo con le CPU AnA, AnAS o AnU.

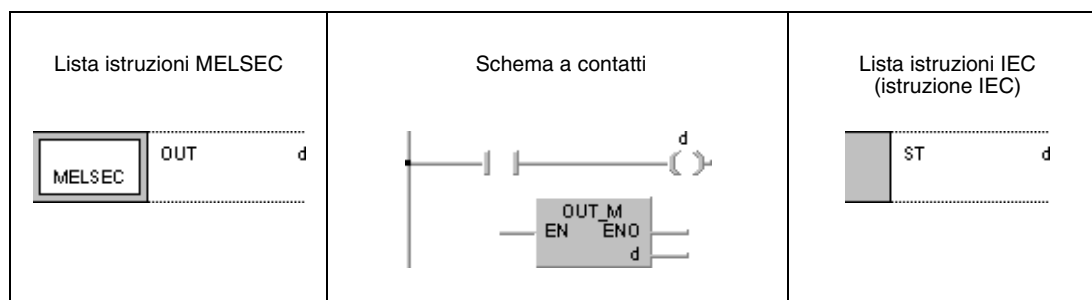
Operandi MELSEC Q

d	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
	● ¹	●	●	●	●	●	—	—	●	—	1 ● ²

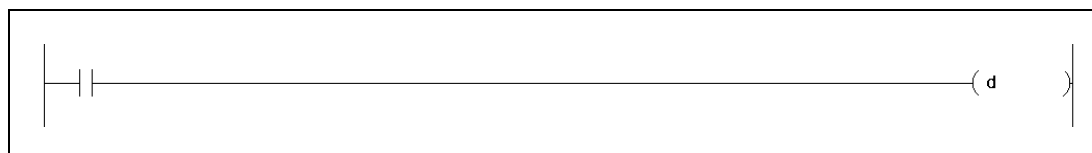
¹ Tranne T, C, F

² 1 passo utilizzando operandi interni, 2 passi usando uscite ad accesso diretto DY, 3 passi con tutti gli altri operandi (compresi file registri con accesso per numero di serie).

GX IEC Developer:



GX Developer:



Variabili

Dati impostati	Significato	Tipo dati
d	Numero dell'operando da eccitare (1) o diseccitare (0).	bit

Funzioni Istruzioni di uscita**OUT Istruzioni di impostazione uscite**

Una uscita viene impostata in funzione della condizione d'ingresso precedente.

Diverse istruzioni OUT possono essere programmate in parallelo, dopo una condizione di ingresso.

Il risultato operazione di un contatto OUT può essere utilizzato come condizione di ingresso per i passi di programma successivi, come contatto NO (normalmente aperto) o NC (normalmente chiuso).

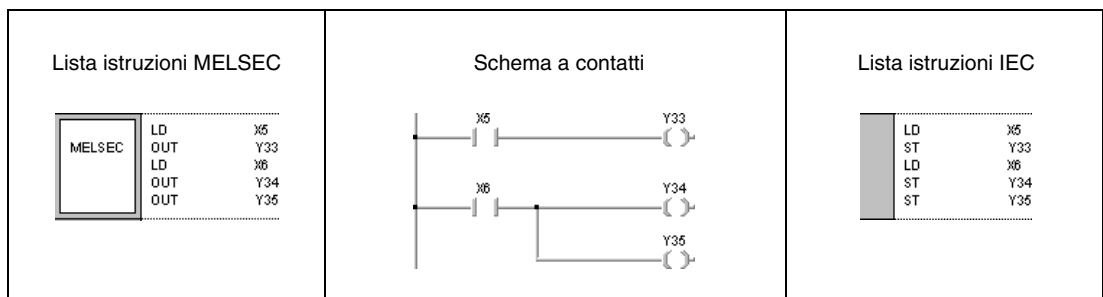
Condizione d'ingresso	Istruzione OUT			Se è specificato un operando a bit
	Contatto di uscita	Tipo di contatto		Bit specificato
		Contatto NO	Contatto NC	
0	OFF	Mancanza di continuità	Continuità	0
1	ON	Continuità	Mancanza di continuità	1

Errori di esecuzione

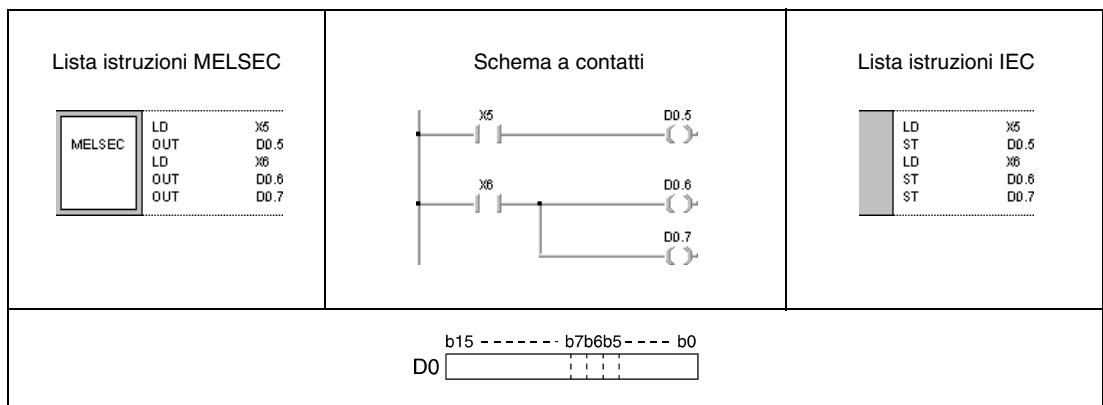
Vedi Manuale di programmazione, parte 1.

Programma di esempio 1**OUT**

Il programma che segue mostra la programmazione di una istruzione OUT usando operandi a bit come uscite (da Y33 a Y35).

**Programma di esempio 2****OUT**

Il programma seguente mostra la programmazione di una istruzione OUT usando bit dell'operando a word D0 come uscite (bit da b5 a b7).



5.3.2 OUT T, OUTH T

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

	Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Segnale di errore	Segnale di errore							
	Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello												
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1							Z	V	K	H (16#)	P	I	N
d								●									●	●					1				
● ¹										● ²							● ²										

¹ Impostazione tempo

² Consultare la sezione "Impostazione valori di timer e contatori estesi" nel manuale di programmazione delle CPU AnA, AnAS o AnU.

Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K			Altro
	Bit	Word		Bit	Word						
d	● ¹	—	—	—	—	—	—	—	—	4	
● ²	—	● ³	●	—	●	●	—	● ⁴	—	4	

¹ solo T

² Impostazione tempo

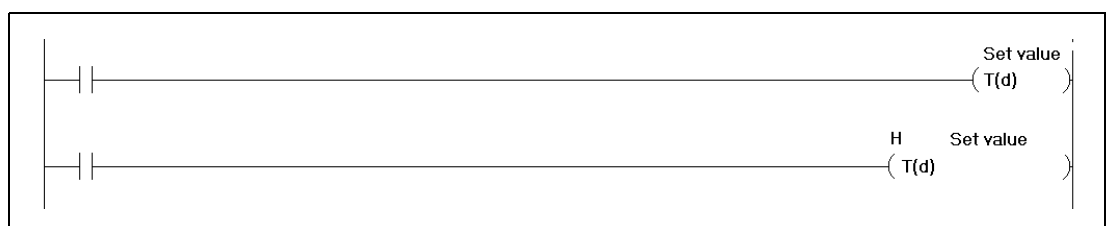
³ Tranne T e C

⁴ Specifica delle impostazioni di tempo con sole costanti decimali (K). Le costanti esadecimali non possono essere utilizzate.

GX IEC Developer:

<p style="text-align: center;">Lista istruzioni MELSEC</p> <table border="1" style="width: 100%;"> <tr> <td style="width: 15%; text-align: center;">MELSEC</td> <td style="width: 15%;">OUT</td> <td style="width: 20%;">T(d) Set Value</td> <td style="width: 50%;">(* Low Speed Timer *)</td> </tr> <tr> <td></td> <td>OUTH</td> <td>T(d) Set Value</td> <td>(* High Speed Timer *)</td> </tr> </table> <p style="text-align: center;">Lista istruzioni IEC</p> <table border="1" style="width: 100%;"> <tr> <td style="width: 15%; text-align: center;">TIMER_M</td> <td style="width: 15%;">TC(d) , Set Value</td> <td style="width: 70%;">(* Low Speed Timer *)</td> </tr> <tr> <td style="text-align: center;">TIMER_H_M</td> <td>TC(d) , Set Value</td> <td>(* High Speed Timer *)</td> </tr> </table>	MELSEC	OUT	T(d) Set Value	(* Low Speed Timer *)		OUTH	T(d) Set Value	(* High Speed Timer *)	TIMER_M	TC(d) , Set Value	(* Low Speed Timer *)	TIMER_H_M	TC(d) , Set Value	(* High Speed Timer *)	<p style="text-align: center;">Schema a contatti</p>
MELSEC	OUT	T(d) Set Value	(* Low Speed Timer *)												
	OUTH	T(d) Set Value	(* High Speed Timer *)												
TIMER_M	TC(d) , Set Value	(* Low Speed Timer *)													
TIMER_H_M	TC(d) , Set Value	(* High Speed Timer *)													

GX Developer:



Variabili

Dati impostati	Significato	Tipo dati
d	Numero del timer.	bit
Impostazione valore	Impostazione tempo.	BIN 16-bit

Funzioni

Impostazione timer

OUT T timer lenti (100 ms)**OUTH T timer veloci (10 ms)**

Se la condizione di ingresso di una istruzione OUT(H) T è vera, il contatto del timer si chiude (1) e rimane chiuso per un tempo specificato. Questo tempo è determinato direttamente con una costante, oppure in modo variabile tramite il valore di un registro dati.

Il risultato operazione di un contatto OUT(H) T può essere programmato come condizione di ingresso in uno o più passi di programma successivi, esattamente come un normale contatto NO (normalmente aperto) o NC (normalmente chiuso).

Dopo che è trascorso il tempo specificato (valore attuale = valore impostato) il contatto di ingresso successivo viene chiuso.

Diverse istruzioni OUT(H) T possono essere programmate dopo una singola condizione di ingresso.

Timer come contatto di uscita			Timer come condizione di ingresso			
Tipo	Contatto esecuzione	Valore attuale	Condizione contatto prima che il tempo impostato sia trascorso		Condizione contatto dopo che il tempo impostato è trascorso	
			Contatto NO	Contatto NC	Contatto NO	Contatto NC
100 ms	OFF	0	Mancanza di continuità	Continuità	Mancanza di continuità	Continuità
10 ms						
100 ms ritentivo	OFF	Valore attuale mantenuto	Mancanza di continuità	Continuità	Continuità	Mancanza di continuità
10 ms ritentivo						

Il risultato operazione di un timer ritentivo viene mantenuto fino a quando non viene cancellato da una istruzione RST.

Un timer non può ricevere valori di impostazione negativi (da -32768 a -1). Una impostazione di tempo 0 viene considerata come 1.

L'esecuzione dell'istruzione OUT(H) T viene processata come segue:

La bobina del timer indicato da d viene eccitata o diseccitata.

Il contatto corrispondente del timer viene chiuso o aperto.

Le impostazioni di tempo vengono rinfrescate.

Se un programma salta su una istruzione OUT(H) T durante la sua esecuzione, le condizioni del contatto e le impostazioni del timer vengono mantenute.

Se una istruzione viene eseguita ripetutamente in solo ciclo, il valore delle ripetizioni viene rinfrescato.

La designazione di bobine e contatti di contatori tramite registri indice (indicizzazione) può essere eseguita solo con i registri indice Z0 e Z1.

NOTA

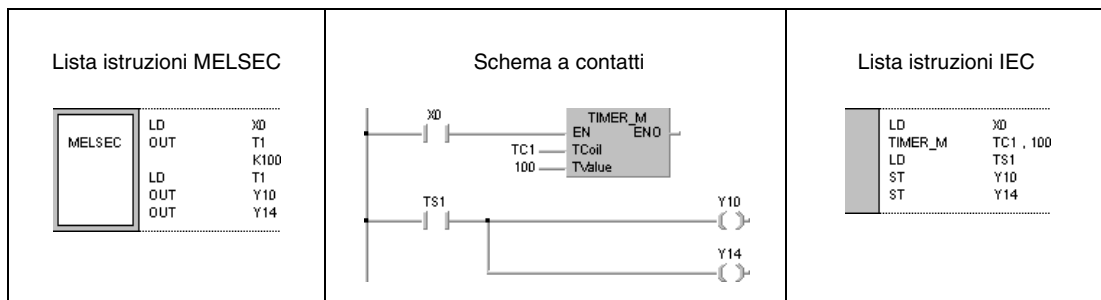
Il registro di impostazione del timer non può essere definito in modo indiretto!

Fare riferimento al capitolo A.3.4 per ulteriori informazioni sui timer.

Programma di esempio 1

OUT T

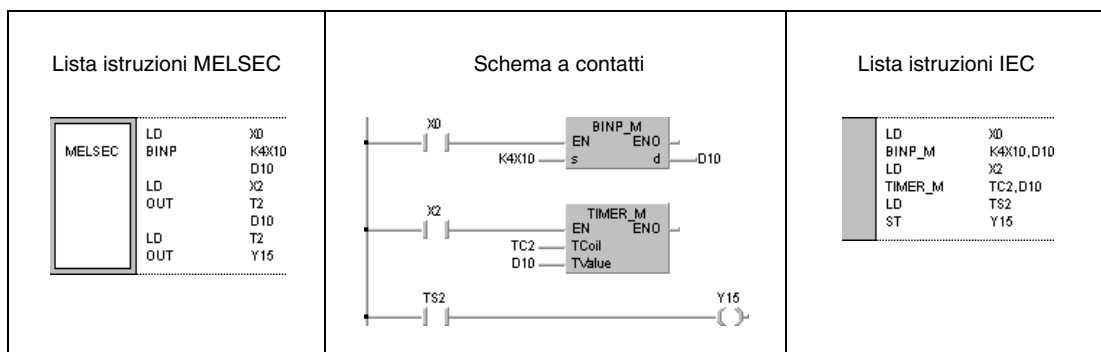
10 secondi dopo l'attivazione di X0, il programma seguente eccita le uscite Y10 e Y14. Viene utilizzato un timer lento (100 ms).



Programma di esempio 2

OUT T

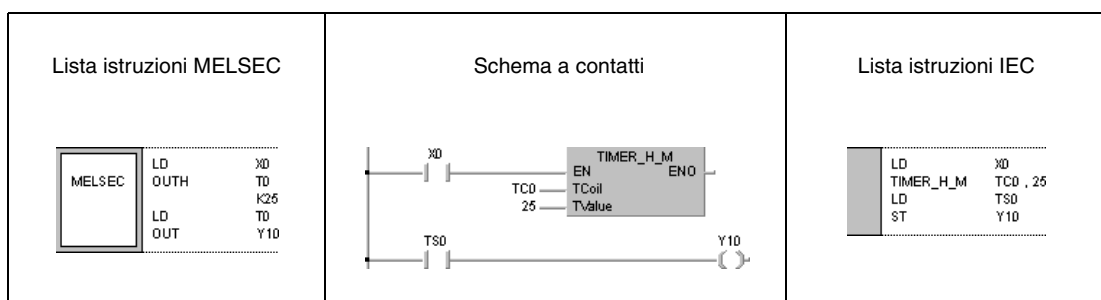
Il programma seguente legge le impostazioni di tempo tramite gli ingressi da X10 a X1F in formato BCD. Con il fronte di salita di X0 il dato BCD viene convertito in dato BIN e successivamente memorizzato in D10. All'attivazione di X2, viene letto il tempo impostato. Quando il tempo impostato è trascorso, l'uscita Y15 viene eccitata. Viene utilizzato un timer lento (100 ms).



Programma di esempio 3

OUTH T

250 ms dopo l'attivazione di X10, il programma seguente eccita l'uscita Y10. Viene utilizzato un timer veloce (10 ms).



Funzioni **Impostazione contatori**

OUT C Contatore

Se la condizione di ingresso di una istruzione OUT C è attiva, il valore attuale del contatore viene incrementato di 1.

Il risultato operazione di un contatto OUT C può essere programmato come condizione di ingresso in uno o più passi di programma successivi, esattamente come un normale contatto NO (normalmente aperto) o NC (normalmente chiuso).

Dopo che il contatore ha raggiunto il valore impostato, il contatto di ingresso successivo viene chiuso.

Se la condizione di ingresso di una istruzione OUT C rimane vera, l'operazione di conteggio non prosegue. Il contatore non richiede quindi un ingresso impulsivo.

Al completamento di una operazione di conteggio, il valore di conteggio impostato e il risultato operazione possono essere cancellati solo con una istruzione RST.

Se vengono utilizzati i contatori estesi da C256 a C1023 con una CPU AnA, AnAS o AnU, fare riferimento alla sezione "Impostazione valori di timer e contatori estesi" in questo manuale di programmazione.

Un contatore non può ricevere valori di impostazione negativi (da -32768 a -1). Una impostazione di conteggio 0 viene considerata come 1.

La designazione di bobine e contatti di contatori tramite registri indice (indicizzazione) può essere eseguita solo con i registri indice Z0 e Z1.

NOTA

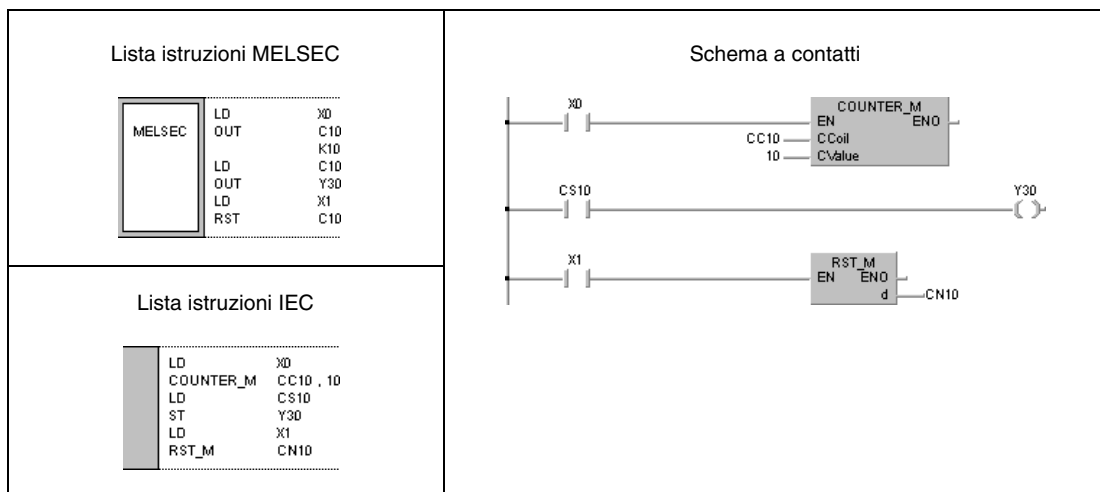
Il registro di impostazione del conteggio non può essere definito in modo indiretto!

Fare riferimento alla sezione A.3.5 di questo manuale per ulteriori informazioni sui contatori.

Programma di esempio 1

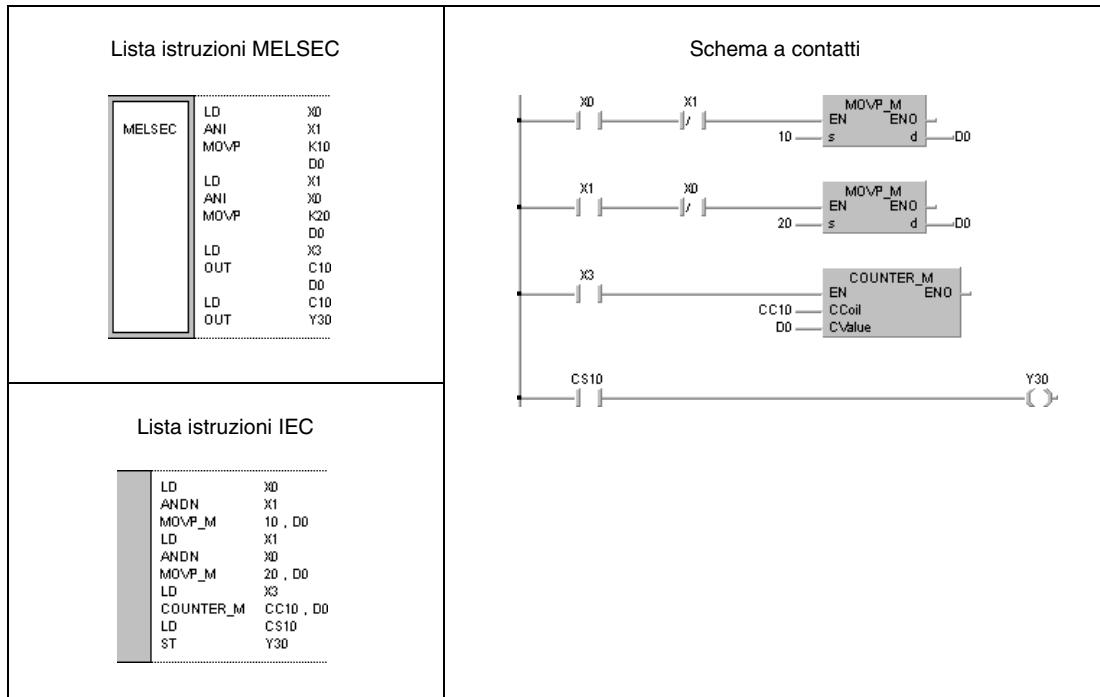
OUT C

Dopo che X0 è stato attivato per dieci volte, il programma che segue eccita Y30 e, se X1 è attivo, diseccita Y30.



**Programma
di esempio 2** OUT C

Il programma che segue imposta il valore di conteggio di C10 con 10 (D0=10) con il fronte di salita di X0, e con 20 (D0=20) con il fronte di salita di X1, Se X3 è attivo, il contatore inizia a contare ed attiva Y30 quando raggiunge il valore di impostazione contenuto in D0.



5.3.4 OUT F

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

Operandi utilizzabili																	Definizione cifra	Numero di passi	Indice	Segnale di errore	Segnale di errore				
Operandi a bit							Operandi a word (16-bit)							Costanti		Puntatore						Livello			
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K						H (16#)	P	I	N
d						●																● ¹	● ²		

¹ In generale, 1 passo. Eccezione: 3 passi per la programmazione di relé interni o spie di errore come operando per l'istruzione OUT. Consultare la sezione "Programmazione con CPU AnA, AnAS e AnU" del manuale di programmazione per il corrispondente numero di passi.

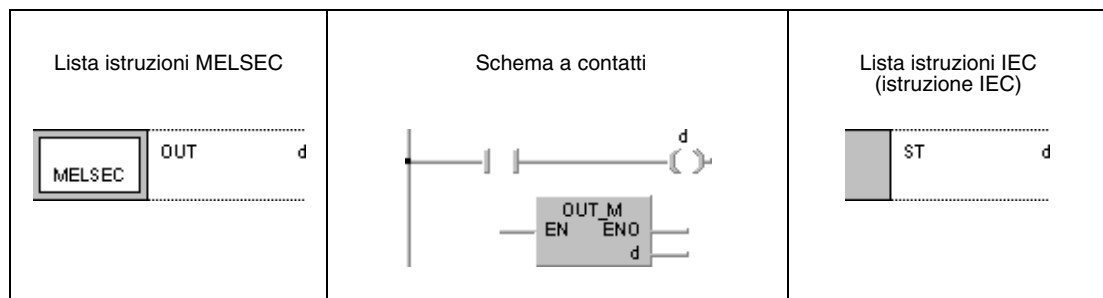
² L'indicizzazione è disponibile solo con le CPU AnA, AnAS o AnU.

Operandi MELSEC Q

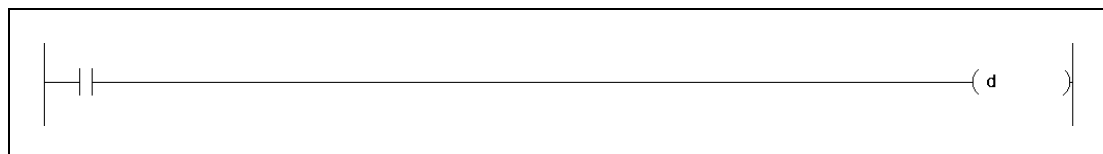
Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word							
d	● ¹	—	—	—	—	—	—	—	—	—	4

¹ solo F

GX IEC Developer:



GX Developer:



Variabili

Dati impostati	Significato	Tipo dati
d	Numero della spia di errore da attivare.	bit (solo F)

Funzioni **Uscite per spie di errore****OUT F Spia di errore (serie Q e System Q)**

Se la condizione di ingresso di una istruzione OUT F è attiva, la spia di errore viene attivata e vengono eseguite le seguenti operazioni:

Il numero della spia di errore viene visualizzato sul display a LED della CPU (Q3A e Q4AR), e viene acceso il LED "USER".

I numeri delle spie di errore attivate sono memorizzati nei registri speciali da SD64 a SD79.

Il valore in SD63 viene incrementato di 1.

Se il registro speciale SD63 contiene il valore 16 (16 numeri di spie di errore memorizzati), nessun numero ulteriore viene memorizzato nel campo di registri da SD64 a SD79.

Se una spia di errore viene disattivata con una istruzione OUT, la visualizzazione sul display a LED, la condizione del LED "USER" ed il contenuto dei registri speciali SD63-SD79 rimangono inalterati.

Spie di errore, registri e visualizzazioni vengono cancellati con l'istruzione RST F.

OUT F Spia di errore (serie A)

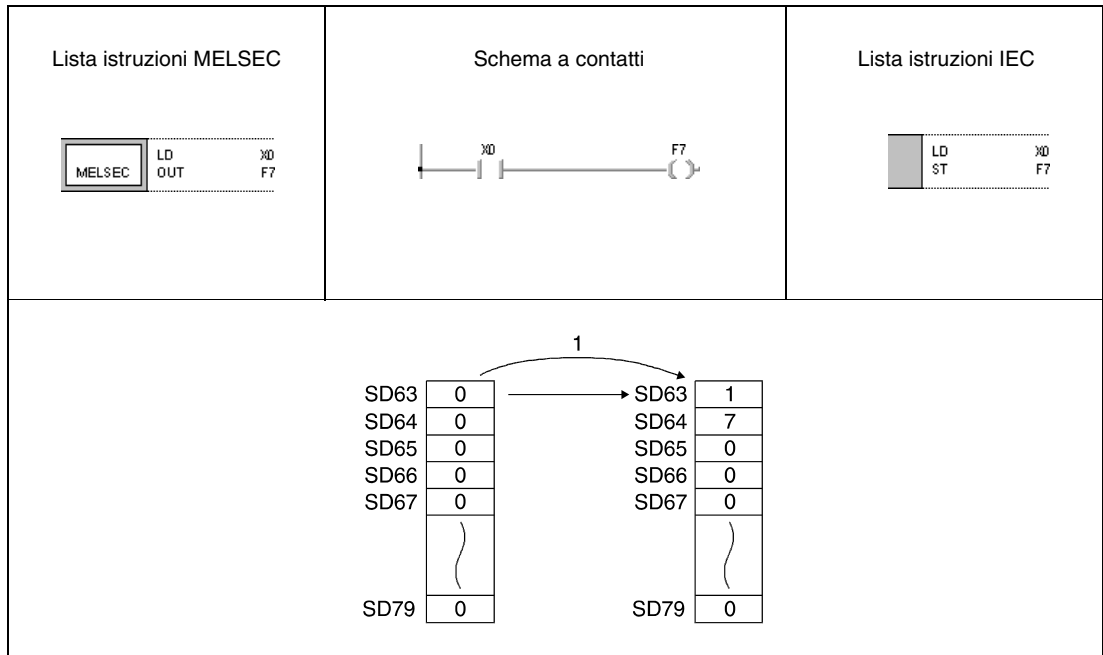
Se un programma attiva una spia di errore (F), il LED ERROR e il display a LED sul modulo CPU si accendono. Il numero di spie di errore attive è contenuto in un registro speciale. Fare riferimento al Manuale di programmazione, parte 1, per ulteriori dettagli.

Le spie di errore non devono essere attivate con una istruzione OUT, perché in questo caso la visualizzazione sul display a LED non corrisponde alla condizione del contatto dell'istruzione di uscita. Per evitare questo, le spie di errore dovrebbero essere impostate con una istruzione SET. L'attivazione di spie di errore con l'istruzione OUT porta anche alla cancellazione della spia se la condizione di ingresso viene disattivata. Il display a LED visualizza lo stato del LED ERROR, mentre il contenuto dei registri speciali viene conservato.

Esempio di programma (Serie Q)

OUT F

Se X0 è attivo, il programma seguente attiva la spia di errore F7. Il numero 7 viene memorizzato nei registri da SD64 a SD79. Il valore del registro SD63 viene incrementato di 1 (1 spia di errore memorizzata).



¹ X0 è attivo

5.3.5 SET

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Segnale di riporto	Segnale di errore									
Operandi a bit						Operandi a word (16-bit)						Costanti		Puntatore							Livello								
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H (16#)	P	I	N			M9012	M9010 M9011					
d	●	●	●	●	●																				1	● ¹	● ²		

¹ Il numero di passi è 3 se relé interni, relé di collegamento o spie di errore (M, B, F) vengono attivati con l'istruzione SET, oppure se vengono disattivati relé interni o qualsiasi operando a word.

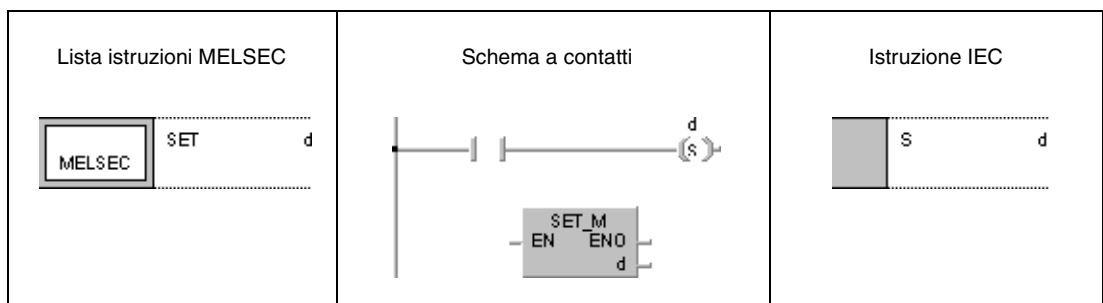
² L'indicizzazione è disponibile solo con le CPU AnA, AnAS o AnU.

Operandi MELSEC Q

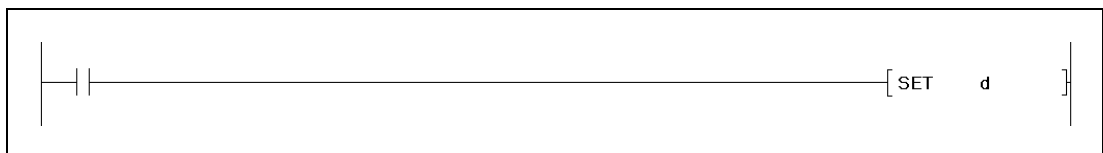
Operandi utilizzabili										Segnale di errore	Numero di passi	
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti K, H (16#)	Altro				
Bit	Word		Bit	Word				BL	DY			
d	●	●	●	●	●	—	—	●	●	—	1	● ¹

¹ 1 passo se si usano operandi interni, 2 passi se si usano uscite ad accesso diretto DY o blocchi SFC (BL), 3 passi se si usa qualsiasi altro operando (compresi file registri con accesso per numero di serie), 4 passi usando timer (T) o contatori (C).

GX IEC Developer:



GX Developer:



Variabili

Dati impostati	Significato	Tipo dati
d	Numero dell'operando a bit (contatto di uscita) da attivare oppure indicazione bit di un operando a word.	bit

Funzioni

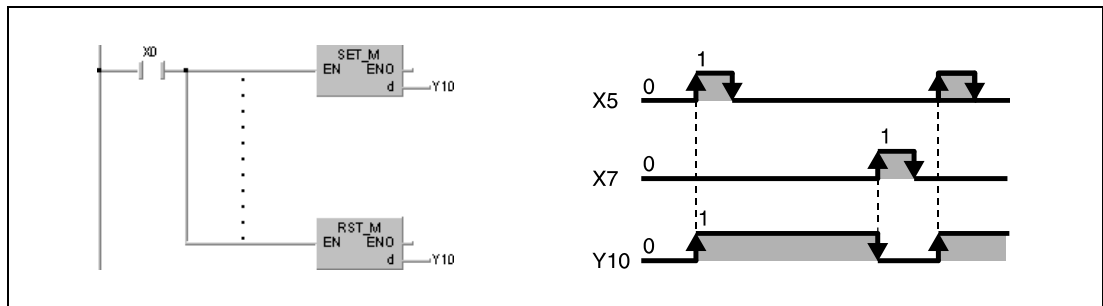
Attivazione operandi

SET Istruzione di attivazione

L'istruzione SET è composta da un comando SET seguito dal numero (indirizzo) dell'operando da attivare.

Dopo l'attivazione della condizione di ingresso dell'istruzione SET, l'operando indicato da d viene attivato, oppure il bit indicato di un operando a word viene impostato a 1.

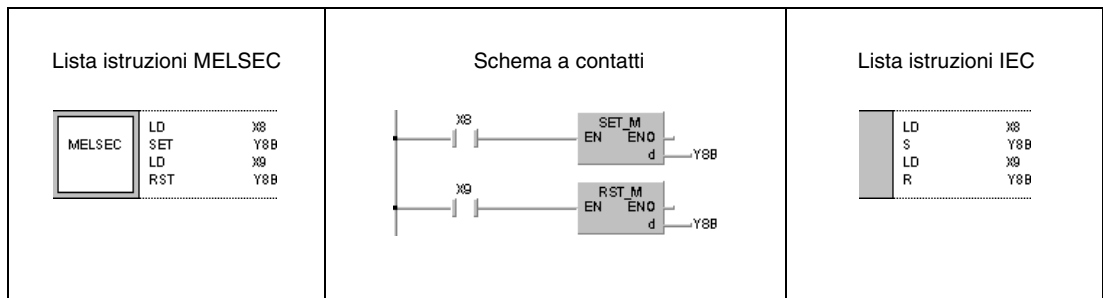
Se la condizione di ingresso viene disattivata, l'operando attivato non viene modificato. L'operando può essere disattivato con l'istruzione RST.



Programma di esempio 1

SET

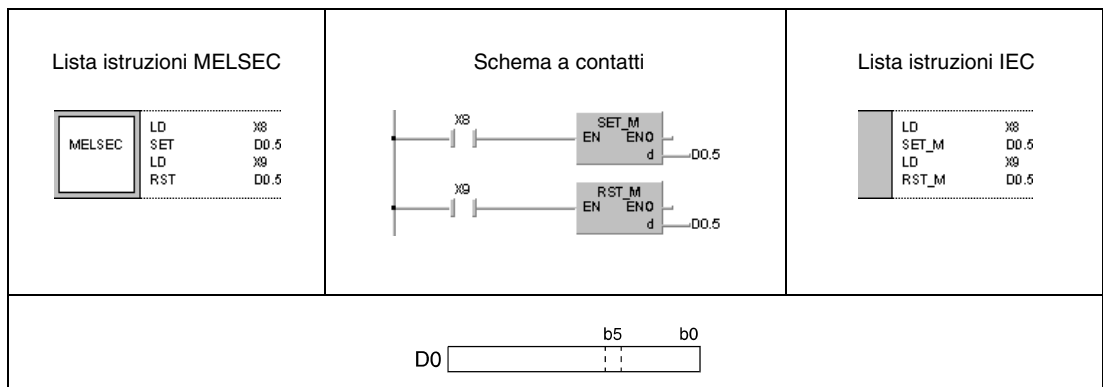
Se X8 è attivo, il programma seguente attiva l'uscita Y8B, Se X9 è attivo, Y8B viene disattivata.



Programma di esempio 2

SET

Se X8 è attivo, il programma seguente imposta il bit 5 (b5) di D0 che passa da 0 a 1. Se X9 viene attivato, questo bit viene disattivato.



5.3.6 RST

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

	Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Segnale di errore M9012	Segnale di errore M9010 M9011										
	Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello															
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z						V	K	H (16#)	P	I	N				
d	●	●	●	●	●		●	●	●	●	●	●	●	●	●											1	● ¹	● ²		

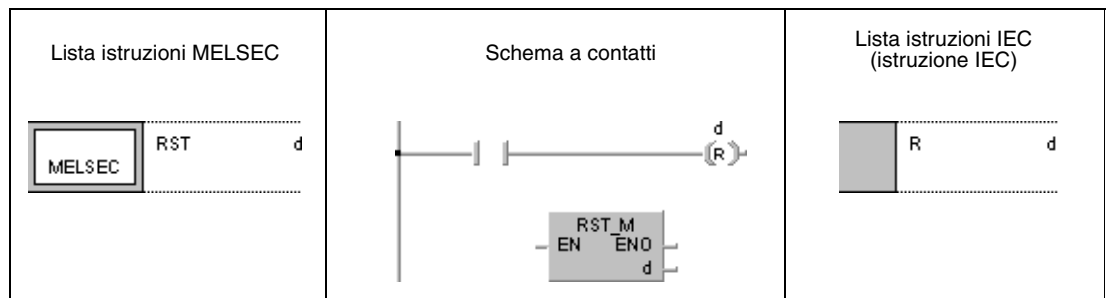
¹ Il numero di passi è 3 se relé interni, relé di collegamento o spie di errore (M, B, F) vengono attivati con l'istruzione SET, oppure se vengono disattivati relé interni o qualsiasi operando a word.

² L'indicizzazione è disponibile solo con le CPU AnA, AnAS o AnU.

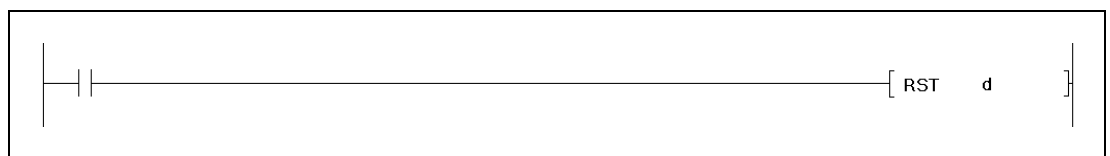
Operandi MELSEC Q

	Operandi utilizzabili									Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro			
	Bit	Word		Bit	Word				BL			DY
d	●	●	●	●	●	●	●	—	—	●	—	2

GX IEC Developer:



GX Developer:



Variabili

Dati impostati	Significato	Tipo dati
d	Numero dell'operando a bit (contatto di uscita) da disattivare oppure indicazione bit di un operando a word.	bit ● ¹

¹ Una funzione speciale dell'istruzione RST_M è la possibilità di azzerare interi operandi a word. In questo caso viene impiegato un minore numero di passi rispetto all'istruzione MOV con la costante K0.

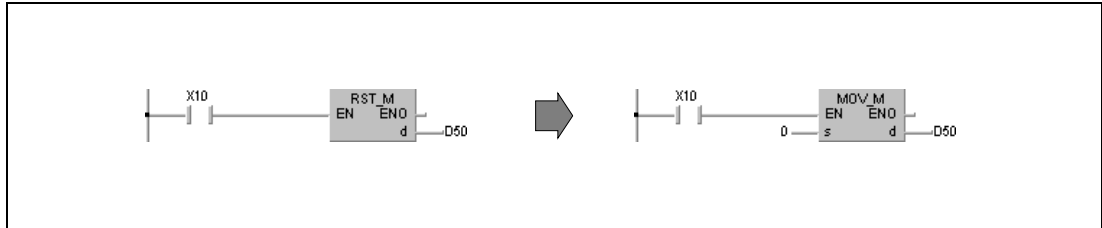
Funzioni Disattivazione operandi

RST Istruzione di disattivazione

L'istruzione RST è composta da un comando RST seguito dal numero (indirizzo) dell'operando da disattivare.

Dopo l'esecuzione dell'istruzione RST i contatti di ingresso e uscita degli operandi a bit vengono disattivati (0), i valori attuali di timer e contatori (T, C) sono impostati a 0 e i contatti corrispondenti disattivati, il bit designato di un operando a word viene azzerato e il contenuto degli operandi a word impostato su 0.

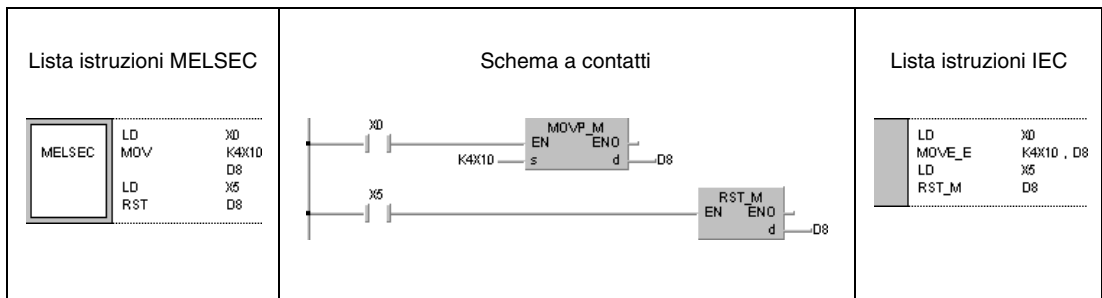
Nello schema seguente, la funzione dell'istruzione RST è identica a quella dell'istruzione MOV sulla destra. X10 è l'ingresso dell'istruzione RST.



Programma di esempio 1

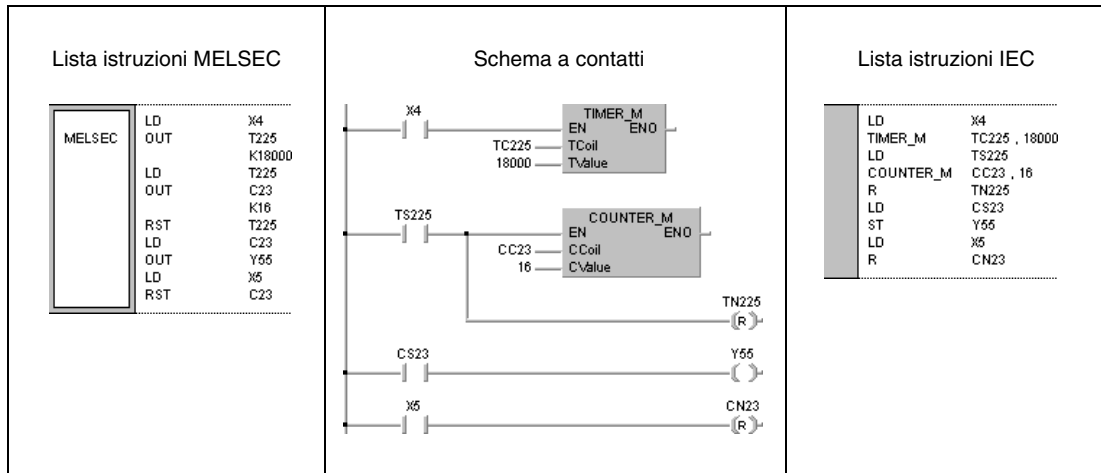
RST

Con il fronte di salita di X0, il programma che segue memorizza il contenuto da X10 a X1F nel registro dati D8. Se X5 viene attivato, il contenuto di D8 viene azzerato.



Programma di esempio 2 RST T, C

Il programma che segue mostra il reset di timer e contatori ritentivi. Nel primo passo di programma viene attivato T225 se X4 è rimasto attivo per 30 minuti (18000 secondi). Nel secondo passo di programma C23 conta il numero di volte che T225 si attiva. Se questo timer viene attivato per 16 volte (valore di impostazione di C223 = 16), l'uscita Y55 viene attivata. Se X5 viene attivato, il contatore viene azzerato.



5.3.7 SET F, RST F

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Segnale di errore	Segnale di errore						
Operandi a bit					Operandi a word (16-bit)					Costanti	Puntatore	Livello													
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z						V	K	H (16#)	P	I	N
d						●																3	●		

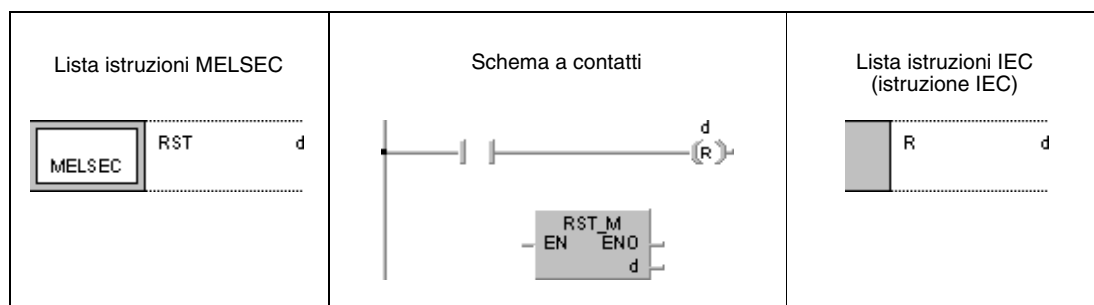
¹ L'indicizzazione è disponibile solo con le CPU AnA, AnAS o AnU.

Operandi MELSEC Q

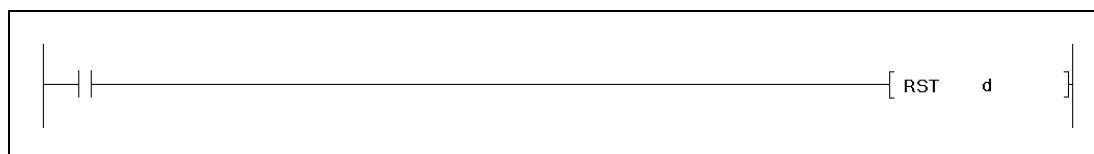
Operandi utilizzabili									Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
Bit	Word		Bit	Word				U		
d	● ¹	—	—	—	—	—	—	—	—	3

¹ solo F.

GX IEC Developer:



GX Developer:



Variabili

Dati impostati	Significato	Tipo dati
d (SET)	Numero della spia di errore da attivare.	bit (solo F)
d (RST)	Numero della spia di errore da disattivare.	bit (solo F)

Funzioni Attivazione e disattivazione di spie di errore (serie Q e System Q)**SET F Istruzione di attivazione**

L'istruzione SET F è composta da un comando SET seguito dal numero (indirizzo) dell'operando da attivare. All'attivazione della condizione di ingresso dell'istruzione SET, l'operando indicato viene attivato. L'istruzione SET genera un impulso per attivare una spia di errore.

Vengono eseguite le seguenti funzioni:

Il numero (indirizzo) della spia di errore viene visualizzato sul display a LED della CPU (Q3A e Q4AR), e viene acceso il LED "USER".

I numeri (indirizzi) delle spie di errore attivate sono memorizzati nei registri speciali da SD64 a SD79.

Il valore in SD63 viene incrementato di 1.

Se il registro speciale SD63 contiene il valore 16 (16 numeri di spie di errore memorizzati), nessun numero ulteriore viene memorizzato nel campo di registri da SD64 a SD79.

RST F Istruzione di disattivazione

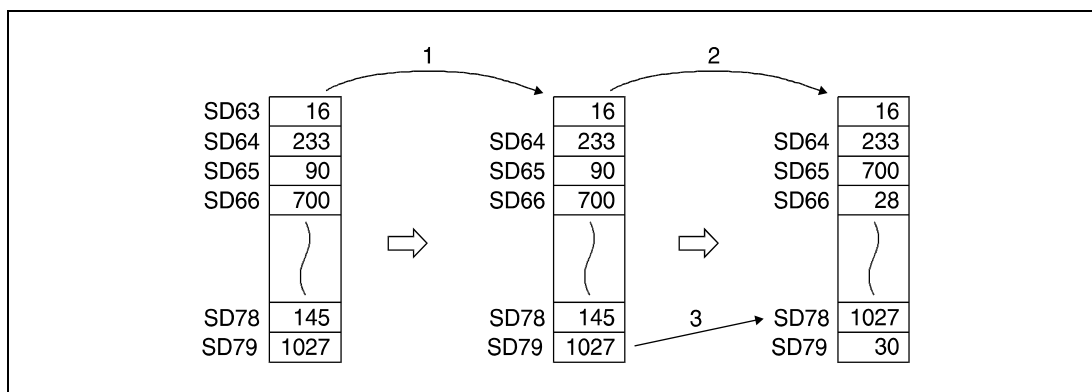
L'istruzione RST F è composta da un comando RST seguito dal numero (indirizzo) dell'operando da disattivare.

Dopo l'attivazione della condizione di ingresso dell'istruzione RST, l'operando corrispondente al numero indicato viene disattivato. Il segnale di uscita per la disattivazione di una spia di errore è un impulso.

Il numero della spia di errore disattivata viene cancellato dai registri SD64 - SD79 ed il valore del registro SD63 viene decrementato di 1. Se il valore di SD63 era 16 e vengono disattivate spie di errore con l'istruzione RST F, è nuovamente possibile attivare nuove spie di errore. Questi numeri di spie di errore vengono inseriti nelle posizioni libere dei registri da SD64 a SD79.

Se il valore del registro speciale SD63 arriva a 0 e tutte le spie di errore sono state disattivate, il display a LED ed il LED "USER" vengono spenti.

Nello schema seguente, F30 viene attivato nel primo passo (1), ma non può essere registrato perché sono già registrati 16 numeri. In un secondo passo (2), F90 viene disattivato. Quindi, in un terzo passo (3), F30 può essere memorizzato in SD79 dato che le altre spie di errore memorizzate vengono fatte scorrere verso l'alto del numero di registri azzerati (SD65).



Attivazione e disattivazione di spie di errore (serie A)

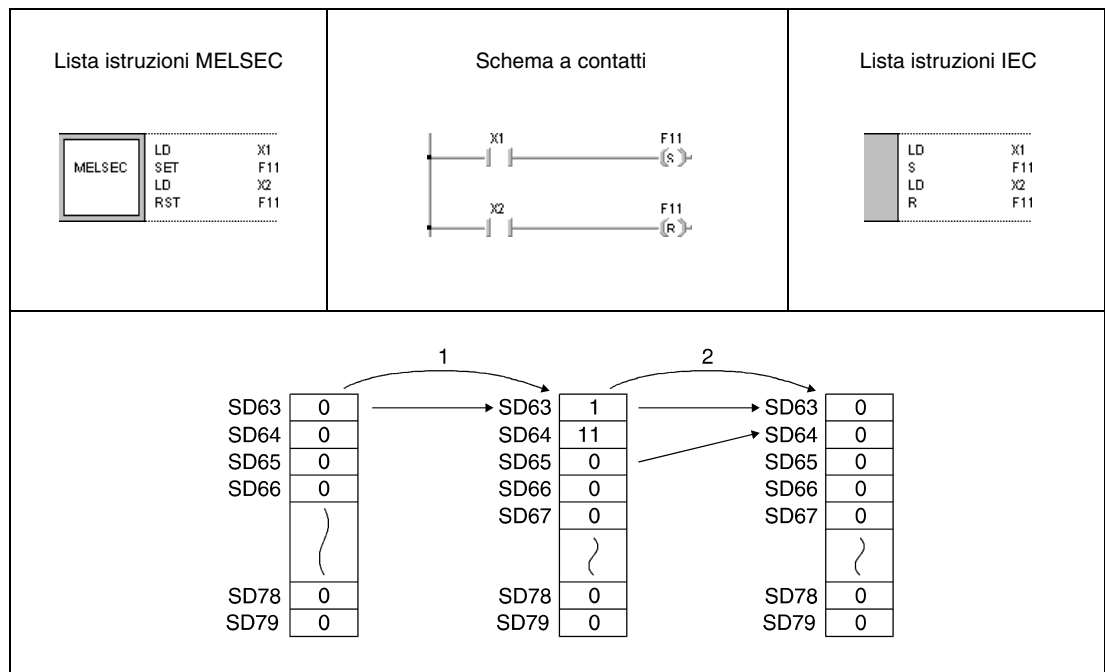
SET F/ RST F Istruzione attivazione / disattivazione

Se una spia di errore F viene attivata o disattivata tramite una istruzione SET/RST, la visualizzazione LED, il LED della condizione di errore sulla CPU ed il contenuto del registro speciale collegato cambiano. Le spie di errore sono attivate o disattivate con segnali impulsivi.

Esempio di programma

SET F/ RST F (seria Q e System Q)

Se X1 è attivo, il programma seguente attiva la spia di errore F11, Il numero 11 viene memorizzato nei registri da SD64 a SD79 ed il valore SD63 viene incrementato di 1 (1). Successivamente, se X2 viene attivato, la spia di errore F11 viene disattivata. Il numero 11 viene cancellato dai registri da SD64 a SD79 ed il valore SD63 viene decrementato di 1 (2).



5.3.8 PLS, PLF

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

Operandi utilizzabili																	Definizione cifra	Numero di passi	Indice	Segnale di errore	Segnale di errore					
Operandi a bit							Operandi a word (16-bit)							Costanti		Puntatore						Livello				
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K						H (16#)	P	I	N	M9012
d	●	●	●	●	●	●																	3	● ¹	● ²	

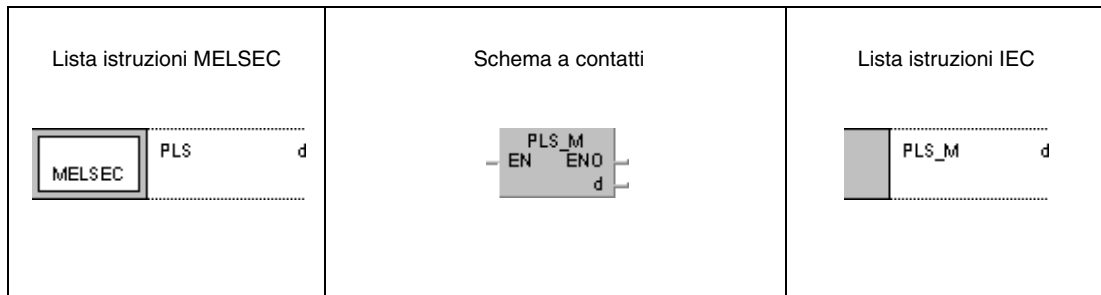
¹ Consultare la sezione "Programmazione con CPU AnA, AnAS e AnU" del manuale di programmazione per il corrispondente numero di passi.

² L'indicizzazione è disponibile solo con le CPU AnA, AnAS o AnU.

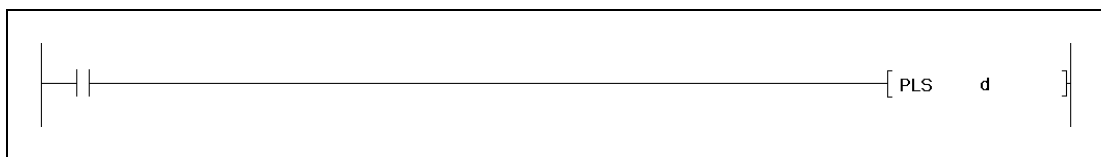
Operandi MELSEC Q

Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word				DY			
d	●	●	●	●	●	—	—	●	—	—	2

GX IEC Developer:



GX Developer:



Variabili

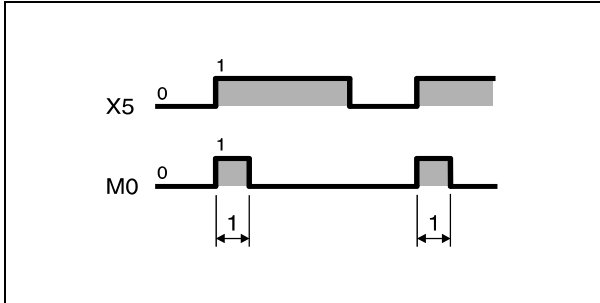
Dati impostati	Significato	Tipo dati
d	Operando il cui segnale di uscita viene convertito in impulso.	bit

Funzioni **Uscita su fronte di salita e discesa,**

PLS Attivazione uscita su fronte di salita

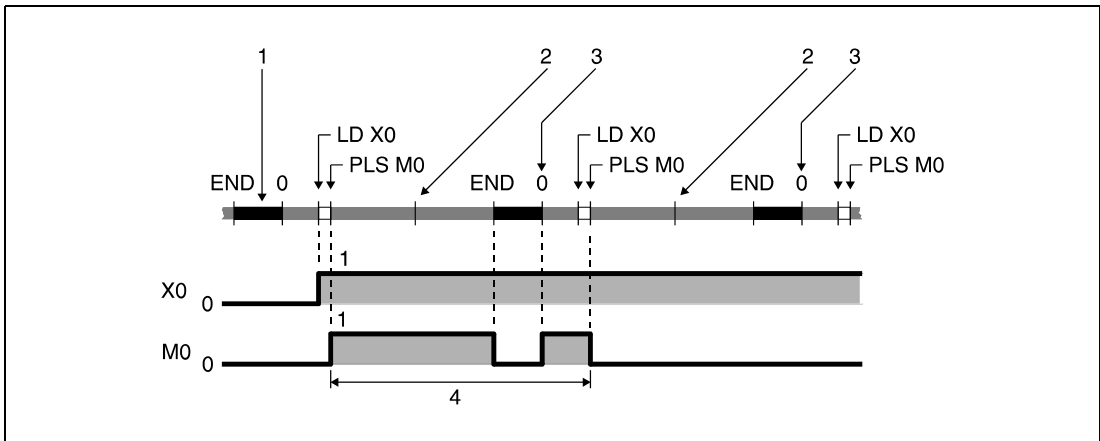
L'istruzione PLS è composta da un comando PLS seguito dal numero (indirizzo) dell'operando da attivare.

L'istruzione PLS (impulso) attiva un operando per una scansione del programma in corrispondenza del fronte di salita della condizione d'ingresso. Se l'operando indicato è già attivo, questo verrà disattivato per una scansione del programma.



¹ Una scansione

Se il commutatore RUN/STOP della CPU viene spostato su STOP mentre è in corso l'esecuzione di una istruzione PLS, questa non viene eseguita di nuovo dopo che il commutatore è stato riportato su RUN, anche se la condizione di ingresso è ancora attiva.



¹ Elaborazione END

² Commutatore RUN/STOP della CPU commutato da RUN a STOP

³ Commutatore RUN/STOP della CPU commutato da STOP a RUN

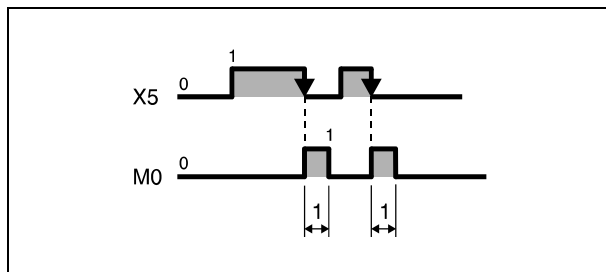
⁴ Una scansione di PLS M0

Se l'istruzione PLS si riferisce a un relé a memoria, e l'alimentazione viene tolta con il relé a memoria eccitato, al ripristino dell'alimentazione il relé a memoria viene attivato per una scansione.

PLF Attivazione uscita su fronte di discesa

L'istruzione PLF è composta da un comando PLF seguito dal numero (indirizzo) dell'operando d da attivare.

L'istruzione PLF attiva un operando per una scansione del programma in corrispondenza del fronte di discesa della condizione d'ingresso. Se l'operando indicato è già attivo, questo verrà disattivato per una scansione del programma.



¹ Una scansione

Se il commutatore RUN/STOP della CPU viene spostato su STOP mentre è in corso l'esecuzione di una istruzione PLF, questa non viene eseguita di nuovo dopo che il commutatore è stato riportato su RUN, anche se la condizione di ingresso è ancora attiva.

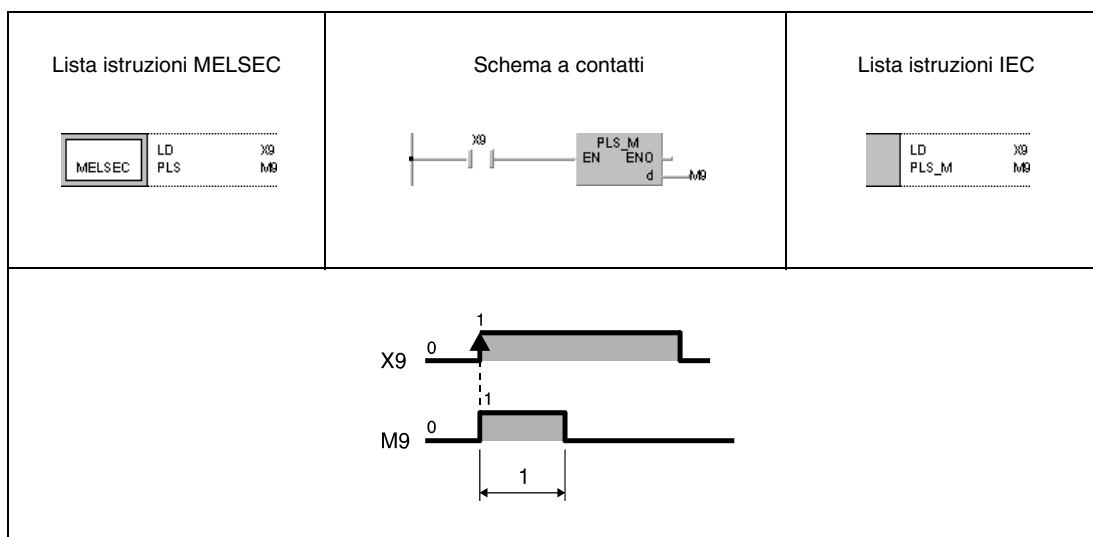
NOTA

L'operando d indicato da una istruzione PLS o PLF rimane attivo per più di una scansione del programma se una istruzione CJ o simile ha saltato l'istruzione PLS o PLF, non eseguendo quella parte del programma.

Programma di esempio 1

PLS

Con il fronte di salita di X9, il programma seguente attiva il relé interno M9 per una scansione del programma.

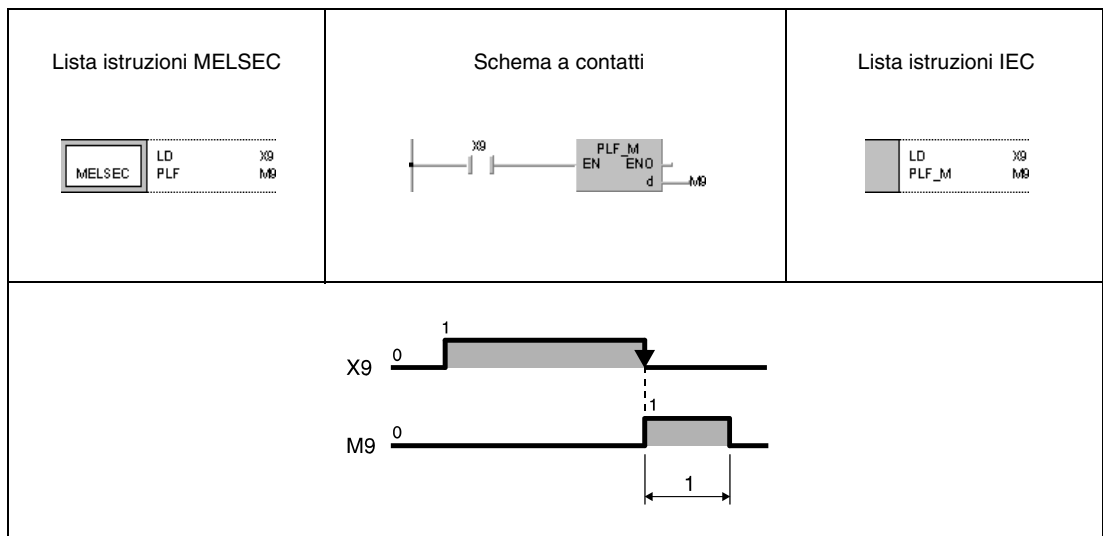


¹ Una scansione

Programma di esempio 2

PLF

Con il fronte di discesa di X9, il programma seguente attiva il relé interno M9 per una scansione del programma.



¹ Una scansione

5.3.9 FF

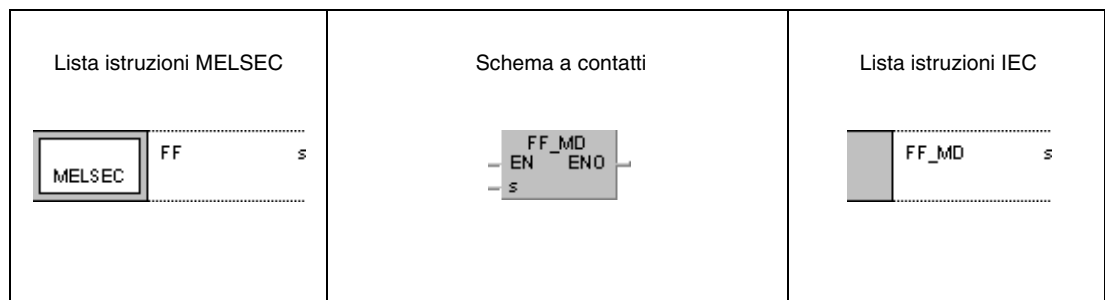
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

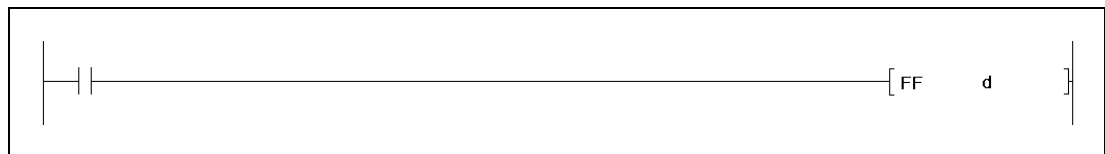
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro DY
	Bit	Word		Bit	Word						
d	●	●	●	●	●	●	—	—	●	—	2

GX IEC Developer:



GX Developer:



Variabili

Dati impostati	Significato	Tipo dati
d	Numero dell'operando a bit, o del bit indicato dell'operando a word, da invertire.	bit

Funzioni **Inversione uscita operando a bit**

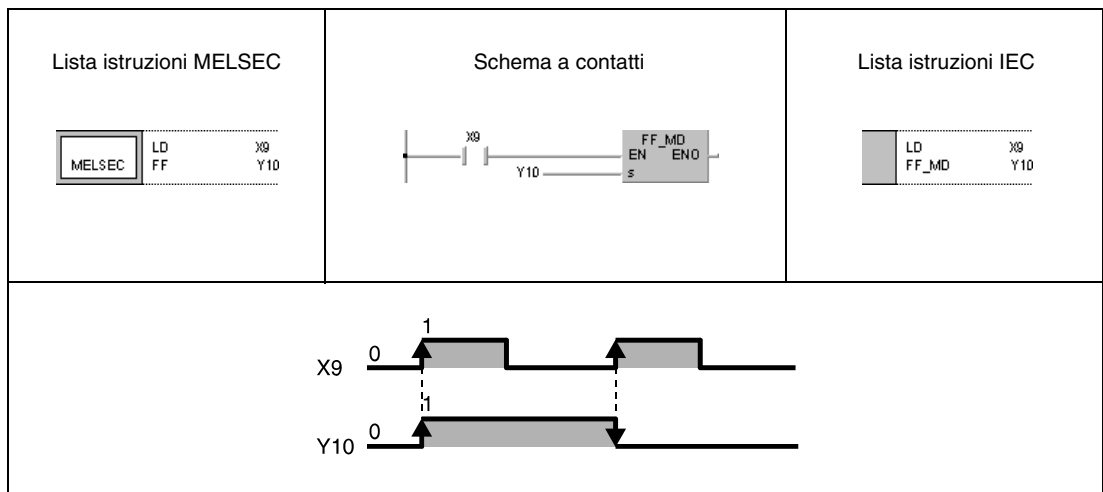
FF **Inversione di operando di uscita a bit**

L'istruzione FF inverte la condizione di operazione dell'operando indicato da d, in corrispondenza del fronte di salita all'ingresso dell'istruzione FF. L'operando può essere un operando a bit o un bit specificato di un operando a word. Se la condizione dell'operando di uscita è attiva (1), dopo l'inversione questa verrà disattivata (0). Se la condizione dell'operando di uscita non è attiva (0), dopo l'inversione questa verrà attivata (1).

Programma di esempio 1

FF

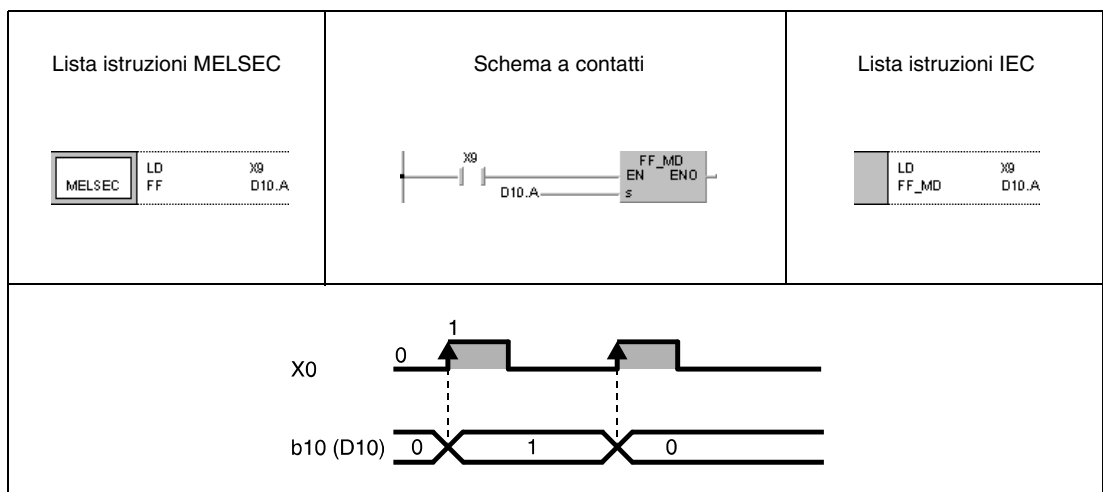
Con il fronte di salita di X9, il programma che segue inverte la condizione di uscita di Y10.



Programma di esempio 2

FF

Con il fronte di salita di X9, il programma che segue inverte il bit 10 (b10) di D10.



Funzioni Inversione uscita operando a bit (serie A)

Note generali

La funzione dell'istruzione CHK varia in funzione del modo di funzionamento. Nella modalità di controllo diretto I/O (tranne CPU AnA e A2C), l'istruzione CHK esegue un controllo di errore. Usando una CPU AnS o AnN in modo controllo I/O con rinfresco, l'istruzione CHK inverte la condizione di funzionamento di un operando di uscita (flip-flop).

CHK Inversione uscita operando a bit

Una istruzione CHK completa è composta dal comando CHK, da un operando d1 di cui si vuole invertire la condizione di uscita e da un operando dummy d2.

Se la condizione di ingresso dell'istruzione CHK è attiva, la condizione di funzionamento dell'operando indicato dall'istruzione CHK viene invertita. Dopo aver disattivato e riattivato nuovamente la condizione di ingresso, l'operando indicato viene riportato alla sua condizione iniziale.

Anche se d2 indica solo un operando dummy, è necessario specificarlo (vedi tabella dei operandi utilizzabili). Se d2 specifica un operando a bit, la cifra deve essere indicata tramite K1 - K4. Si può indicare qualsiasi valore, dato che i dati non sono significativi. L'operando d2 può essere liberamente utilizzato per altri scopi.

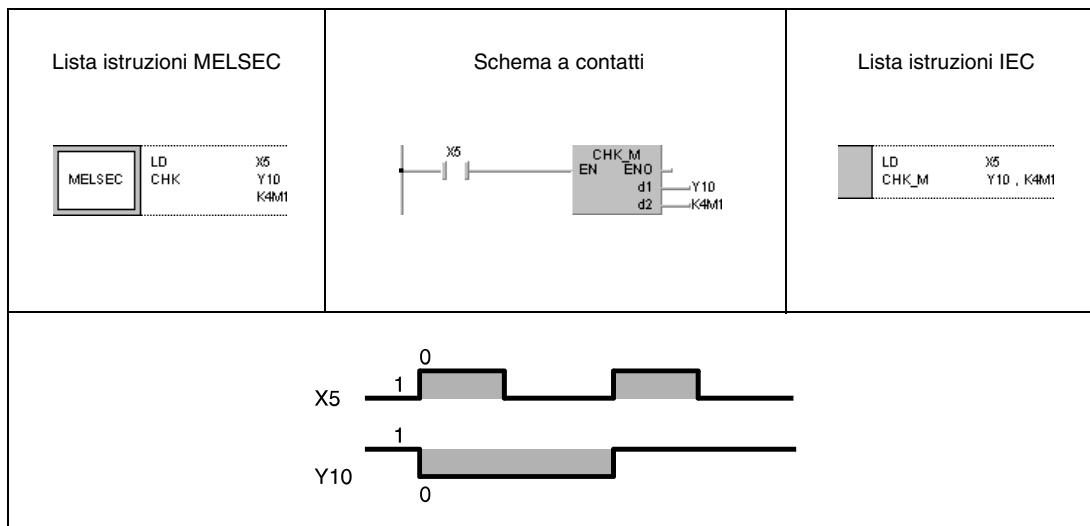
L'istruzione CHK descritta viene eseguita solo in modalità rinfresco.

L'inversione della condizione di funzionamento di un operando di uscita deve essere mantenuta almeno per la durata di una scansione di programma.

Esempio di programma

CHK

Con il fronte di salita di X5, il programma che segue inverte la condizione di uscita di Y10.



5.3.11 DELTA, DELTAP

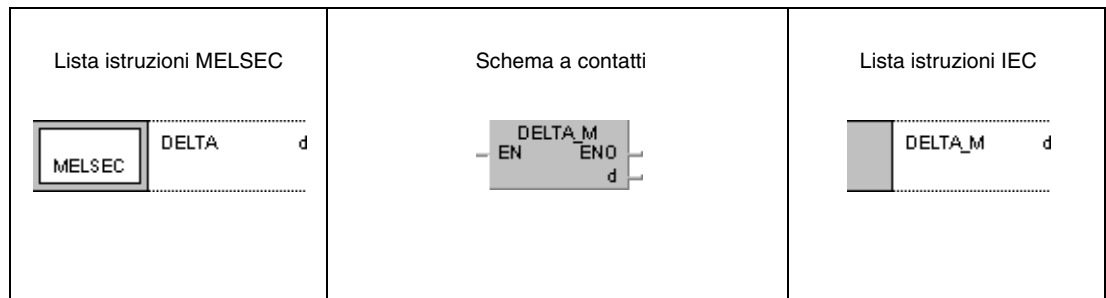
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

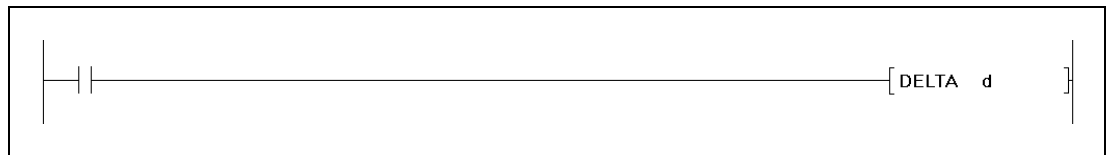
Operandi MELSEC Q

d	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						DY
	—	—	—	—	—	—	—	—	●	SM0	2

GX IEC Developer:



GX Developer:



Variabili

Dati impostati	Significato	Tipo dati
d	Numero dell'uscita ad accesso diretto su cui si deve generare un impulso.	bit ● ¹

¹ solo uscite ad accesso diretto

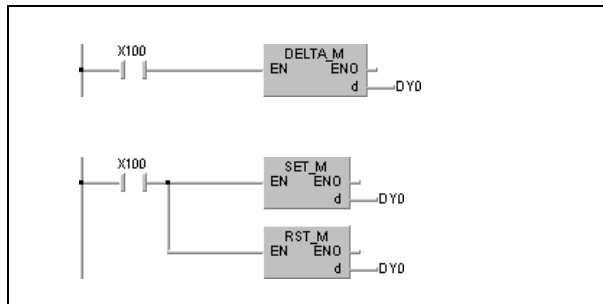
Funzioni **Generazione di impulsi su uscite ad accesso diretto**

DELTA Conversione di contatti in impulsi

L'istruzione DELTA genera un impulso sull'uscita ad accesso diretto (DY) indicata da d, cioè l'uscita viene attivata solo per un tempo determinato.

Se l'uscita indicata dall'istruzione DELTA è DY0, la funzione eseguita è identica a quella dell'istruzione SET/RST (vedi schema).

L'istruzione DELTA(P) viene usata con i comandi per esecuzione sul fronte di salita in unità funzione speciali.



Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il numero dell'uscita indicata da d supera il campo delle uscite. (codice di errore: 4101).

Esempio di programma

DELTAP

Con il fronte di salita di X20, il programma seguente esegue il preset di CH1 dell'unità di uscita AD61 montata nello slot 0 del telaio di base. Il valore 0 di preset viene memorizzato negli indirizzi 1 e 2 del buffer di memoria di AD61. L'istruzione DELTAP emette l'istruzione di preset su DY11.

<p style="text-align: center;">Lista istruzioni MELSEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;">MELSEC</td> <td style="padding: 2px;">LD</td> <td style="padding: 2px;">X20</td> </tr> <tr> <td></td> <td style="padding: 2px;">DMOV</td> <td style="padding: 2px;">K0 U0V1</td> </tr> <tr> <td></td> <td style="padding: 2px;">DELTAP</td> <td style="padding: 2px;">DY11</td> </tr> </table>	MELSEC	LD	X20		DMOV	K0 U0V1		DELTAP	DY11	<p>Schema a contatti</p>	<p style="text-align: center;">Lista istruzioni IEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">LD</td> <td style="padding: 2px;">X20</td> </tr> <tr> <td style="padding: 2px;">MOV_P_M</td> <td style="padding: 2px;">0, U0V1</td> </tr> <tr> <td style="padding: 2px;">MOV_P_M</td> <td style="padding: 2px;">0, U0V2</td> </tr> <tr> <td style="padding: 2px;">DELTAP_M</td> <td style="padding: 2px;">DY11</td> </tr> </table>	LD	X20	MOV_P_M	0, U0V1	MOV_P_M	0, U0V2	DELTAP_M	DY11
MELSEC	LD	X20																	
	DMOV	K0 U0V1																	
	DELTAP	DY11																	
LD	X20																		
MOV_P_M	0, U0V1																		
MOV_P_M	0, U0V2																		
DELTAP_M	DY11																		

5.4 Istruzioni di scorrimento

5.4.1 SFT, SFTP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

d	Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Segnale di errore	Segnale di errore					
	Operandi a bit						Operandi a word (16-bit)						Costanti		Puntatore							Livello				
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V				K	H (16#)	P	I	N	M9012	M9010 M9011
	●	●	●	●	●	●																	3 ¹	● ²		

¹ Consultare la sezione "Programmazione con CPU AnA, AnAS e AnU" del manuale di programmazione per il corrispondente numero di passi.

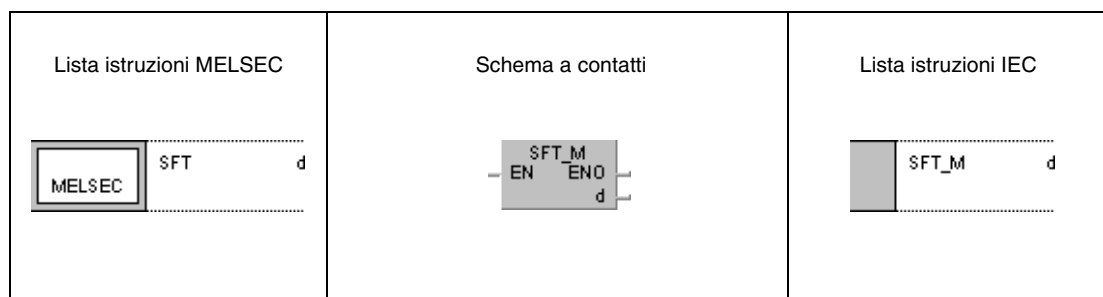
² L'indicizzazione è disponibile solo con le CPU AnA, AnAS o AnU.

Operandi MELSEC Q

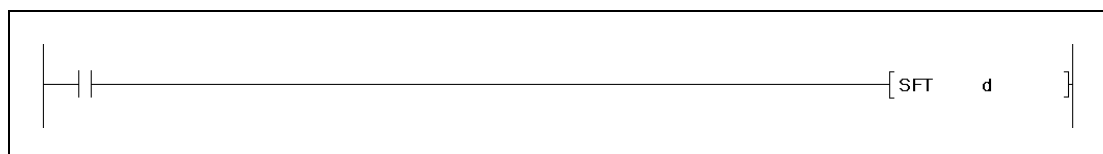
d	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti DY			Altro
	Bit	Word		Bit	Word						
	● ¹	● ¹	● ¹	● ¹	● ¹	● ¹	—	—	●	—	2

¹ Tranne T e C

GX IEC Developer:



GX Developer:



Variabili

Dati impostati	Significato	Tipo dati
d	Numero dell'operando che deve scorrere.	bit

Funzioni Istruzioni di scorrimento

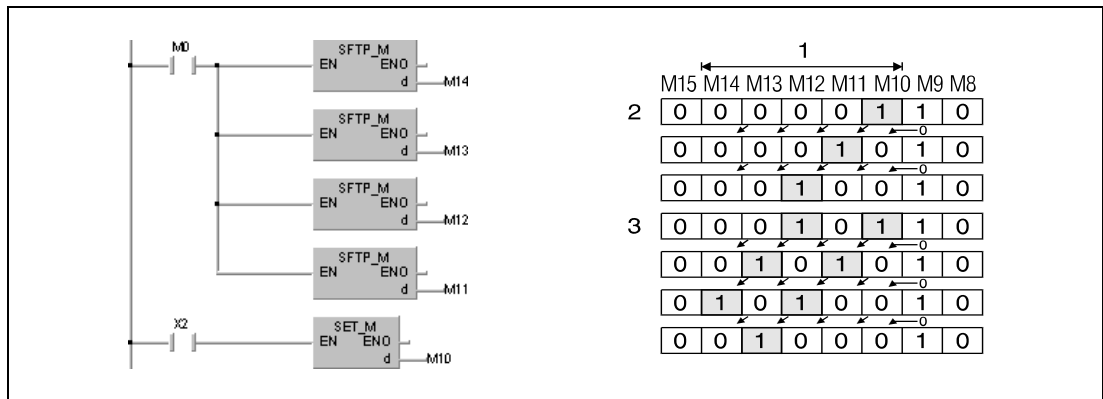
SFT Scorrimento di operandi a bit

L'istruzione SFT scorre di un bit un operando. Gli operandi vengono fatti scorrere dall'istruzione SFT solo se la condizione di ingresso è soddisfatta (fronte di salita).

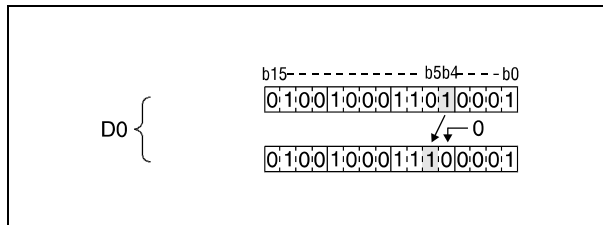
L'istruzione fa scorrere lo stato di un operando (indicato da d-1) nell'indirizzo di destinazione d. Lo stato dell'operando con indirizzo d-1 viene azzerato. Il numero di indirizzi di operando che vengono fatti scorrere può essere impostato con l'istruzione SET.

Se diverse istruzioni SFT vengono programmate in sequenza, il programma inizia lo scorrimento dall'operando con il numero più alto.

Il programma seguente attiva il relé interno M10 se X2 è attivo (2,3). Lo stato di M10 (1) viene fatto scorrere nel campo di scorrimento (1) dall'istruzione SFT P.



Se vengono fatti scorrere bit di un operando a word, la condizione (0/1) del bit d-1 viene spostata nel bit d. Il bit d-1 viene azzerato al termine dell'istruzione SFT. Nella figura che segue, viene fatto scorrere il bit 5 (b5) di D0. Il bit 4 (b4) viene azzerato al termine dell'istruzione.



Esempio di programma SFT

Con il fronte di salita di X8, il programma seguente fa scorrere lo stato di Y57 in Y5B. Y57 viene attivato con il fronte di salita di X7.

<p style="text-align: center;">Lista istruzioni MELSEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;">MELSEC</td> <td style="padding: 2px;">LDP</td> <td style="padding: 2px;">X8</td> </tr> <tr> <td></td> <td style="padding: 2px;">SFT</td> <td style="padding: 2px;">Y5B</td> </tr> <tr> <td></td> <td style="padding: 2px;">SFT</td> <td style="padding: 2px;">Y5A</td> </tr> <tr> <td></td> <td style="padding: 2px;">SFT</td> <td style="padding: 2px;">Y59</td> </tr> <tr> <td></td> <td style="padding: 2px;">SFT</td> <td style="padding: 2px;">Y58</td> </tr> <tr> <td></td> <td style="padding: 2px;">LDP</td> <td style="padding: 2px;">X7</td> </tr> <tr> <td></td> <td style="padding: 2px;">SET</td> <td style="padding: 2px;">Y57</td> </tr> </table>	MELSEC	LDP	X8		SFT	Y5B		SFT	Y5A		SFT	Y59		SFT	Y58		LDP	X7		SET	Y57	<p style="text-align: center;">Schema a contatti</p>	<p style="text-align: center;">Lista istruzioni IEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">LD</td> <td style="padding: 2px;">X8</td> </tr> <tr> <td style="padding: 2px;">PLS_M</td> <td style="padding: 2px;">M8</td> </tr> <tr> <td style="padding: 2px;">LD</td> <td style="padding: 2px;">M8</td> </tr> <tr> <td style="padding: 2px;">SFT_M</td> <td style="padding: 2px;">Y5B</td> </tr> <tr> <td style="padding: 2px;">SFT_M</td> <td style="padding: 2px;">Y5A</td> </tr> <tr> <td style="padding: 2px;">SFT_M</td> <td style="padding: 2px;">Y59</td> </tr> <tr> <td style="padding: 2px;">SFT_M</td> <td style="padding: 2px;">Y58</td> </tr> <tr> <td style="padding: 2px;">LD</td> <td style="padding: 2px;">X7</td> </tr> <tr> <td style="padding: 2px;">PLS_M</td> <td style="padding: 2px;">M7</td> </tr> <tr> <td style="padding: 2px;">LD</td> <td style="padding: 2px;">M7</td> </tr> <tr> <td style="padding: 2px;">S</td> <td style="padding: 2px;">Y57</td> </tr> </table>	LD	X8	PLS_M	M8	LD	M8	SFT_M	Y5B	SFT_M	Y5A	SFT_M	Y59	SFT_M	Y58	LD	X7	PLS_M	M7	LD	M7	S	Y57
MELSEC	LDP	X8																																											
	SFT	Y5B																																											
	SFT	Y5A																																											
	SFT	Y59																																											
	SFT	Y58																																											
	LDP	X7																																											
	SET	Y57																																											
LD	X8																																												
PLS_M	M8																																												
LD	M8																																												
SFT_M	Y5B																																												
SFT_M	Y5A																																												
SFT_M	Y59																																												
SFT_M	Y58																																												
LD	X7																																												
PLS_M	M7																																												
LD	M7																																												
S	Y57																																												

5.5 Istruzioni di controllo primario

5.5.1 MC, MCR

NOTA Queste istruzioni non dovrebbero essere utilizzate con gli editor IEC.

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

	Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Segnale di errore	Segnale di errore				
	Operandi a bit						Operandi a word (16-bit)						Costanti		Puntatore							Livello			
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V				K	H (16#)	P	I	N	M9012
n																					●	3/5	● ¹		
d	●	●	●	●	●	●																●	● ²		

¹ L'indicizzazione è disponibile solo con le CPU AnA, AnAS o AnU.

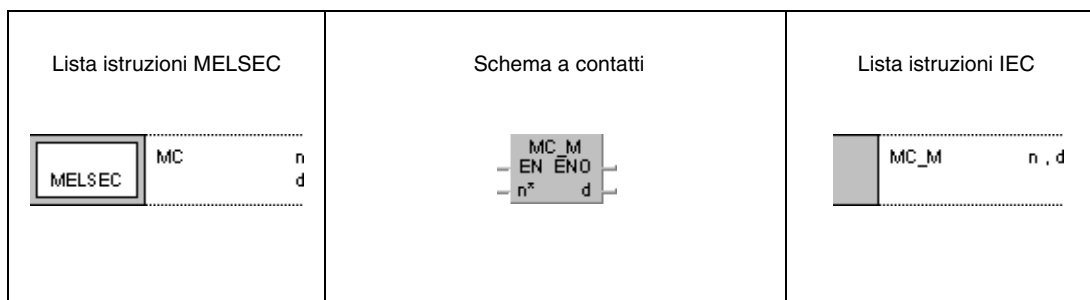
² Il numero di passi dell'istruzione MC è 5 e quello dell'istruzione MCR è 3. Consultare la sezione "Programmazione di una CPU AnA, AnAS e AnU" nel manuale di programmazione, per il relativo numero di passi.

Operandi MELSEC Q

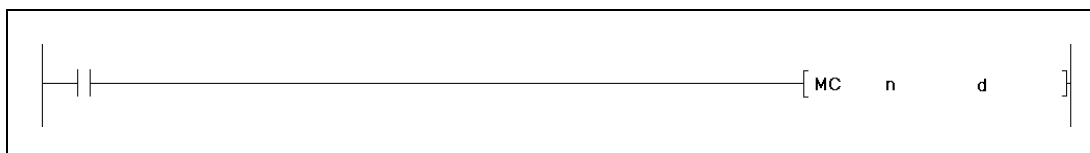
	Operandi utilizzabili									Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro			
	Bit	Word		Bit	Word				N			DY
n	—	—	—	—	—	—	—	—	●	—	1/2	
d	●	●	●	●	●	●	—	—	—	●	● ¹	

¹ Il numero di passi è 2 per l'istruzione MC e 1 per l'istruzione MCR.

GX IEC Developer:



GX Developer:



Variabili

Dati impostati	Significato	Tipo dati
n	Livello di annidamento (serie A ? N0 - N7, serie Q e System Q = N0 - N14).	Annidamento
d	Numero dell'operando che imposta il livello.	bit

Funzioni Attivazione e disattivazione controllo primario

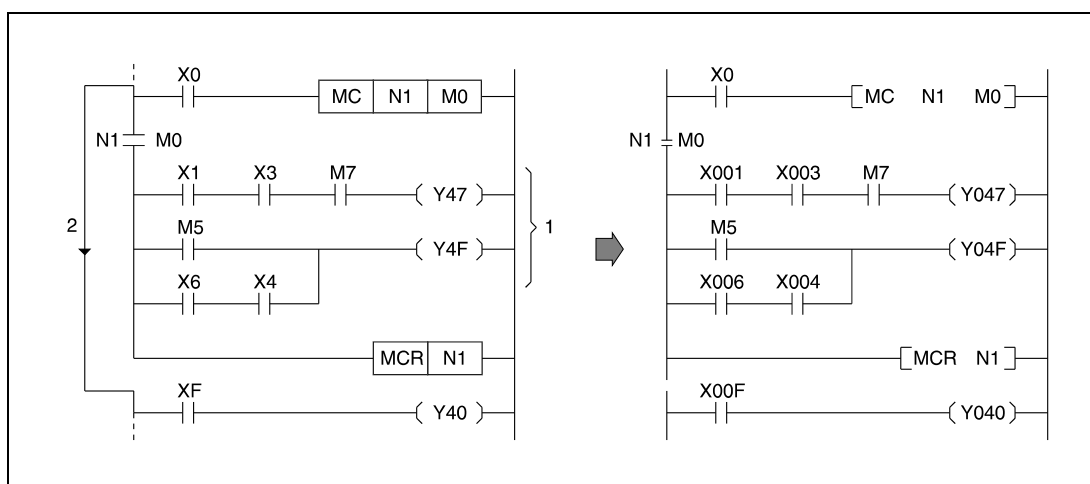
Note generali

L'istruzione MC viene utilizzata per programmi di sequenza con una efficiente commutazione di parti del programma. Quando la condizione di ingresso è attiva, viene eseguita la parte di programma fra la destinazione d e l'istruzione MCR. Le regioni di controllo primario vengono distinte per annidamento (da N0 a N7 per la serie A, e da N0 a N14 per la serie Q e il System Q).

Dato che il software GX IEC Developer non consente una efficace rappresentazione delle istruzioni MC/MCR, vengono qui utilizzati gli schemi a contatti del pacchetto GX Developer.

Lo schema a contatti mostra la funzione dell'istruzione MC. Se l'ingresso X0 non è attivo, la parte di programma del livello 1 (indicata da N1) viene saltata (1). Se X0 è attivo, la parte di programma da N1 all'istruzione MCR viene eseguita (2).

Se si programma con lo schema a contatto, non è necessario inserire i contatti MC sul bus verticale, dato che vengono rappresentati automaticamente.



MC Attivazione di parti definite del programma

L'istruzione MC è l'istruzione di partenza per il controllo primario che consente l'elaborazione di una parte di programma specificata. Se la condizione di ingresso dell'istruzione MC è attiva, gli operandi compresi tra l'istruzione MC e l'istruzione MCR vengono elaborati regolarmente.

Gli operandi compresi fra le istruzioni MC e MCR vengono elaborati anche dopo che la condizione di ingresso di MC non è più attiva. Quindi in questo caso il tempo di scansione del programma non viene diminuito. Quando la condizione di ingresso non è attiva, gli operandi compresi tra le istruzioni MC e MCR vengono elaborati come segue:

Operandi	Elaborazione
Timer da 10 ms Timer da 100 ms	Valore di conteggio azzerato. Contatti di ingresso e uscita aperti (0).
Timer 10 ms ritentivi (solo serie Q e System Q) Timer 100 ms ritentivi Contatore	Valore di impostazione conteggio e stato dei contatti di ingresso mantenuti. Contatto di uscita aperto (0).
Operandi delle istruzioni OUT	Tutte le uscite azzerate.
Operandi delle istruzioni SET, RST e SFT	Stato attuale conservato

NOTA

Se una istruzione che non richiede condizione di ingresso (ad es. FOR/NEXT, EI, DI) viene posta fra le istruzioni MC e MCR, queste istruzioni vengono eseguite dal PLC senza considerare la condizione di ingresso dell'istruzione MC.

Per una istruzione MC è consentito l'uso dello stesso livello di annidamento n, se vengono utilizzati numeri di operandi(indirizzi) diversi.

Dopo l'attivazione dell'istruzione MC, l'operando indicato da d viene attivato. Se questo operando viene utilizzato come condizione di ingresso in altre parti del programma, i contatti vengono elaborati come contatti raddoppiati e aperti e chiusi in parallelo. Quindi, l'operando indicato da d non deve essere utilizzato altrove nel programma.

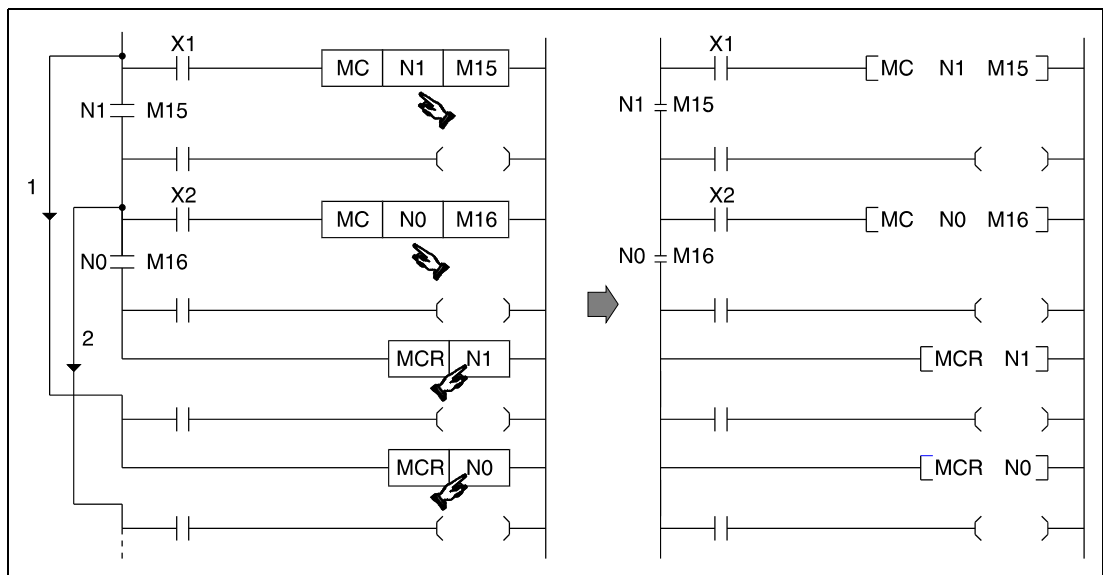
MCR Disattivazione di parti definite del programma

L'istruzione MCR cancella l'istruzione MC e indica la fine della parte di programma soggetta al controllo primario.

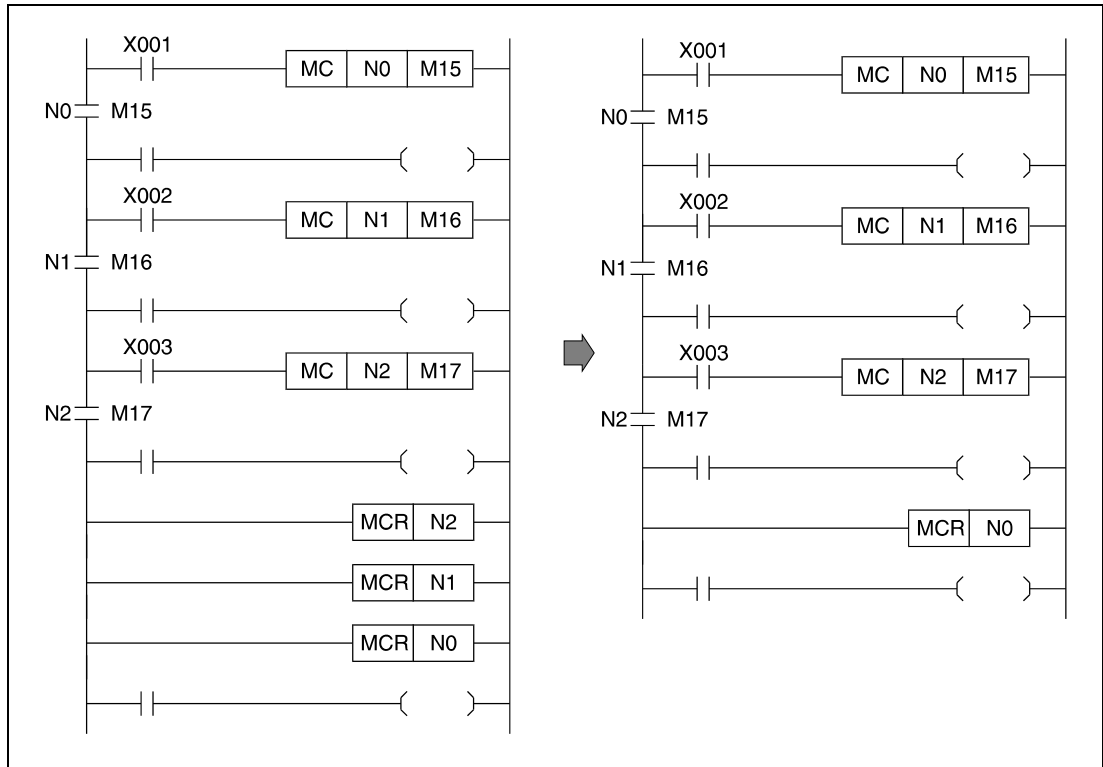
L'istruzione MCR non deve essere attivata tramite un contatto di ingresso.

Note sulla programmazione dei numeri di annidamento (indirizzi):

La serie Q e i System Q offrono 15 livelli di annidamento, da N0 a N14; la serie A offre 8 livelli di annidamento, da N0 a N7. La prima regione soggetta a controllo primario, indicata dall'istruzione MC deve partire dall'indirizzo di annidamento più basso, mentre la prima istruzione MCR deve iniziare con l'indirizzo di annidamento più alto. Se gli indirizzi di annidamento vengono indicati in ordine diverso, i livelli corrispondenti (1, 2) non vengono elaborati correttamente dal PLC. Lo schema seguente mostra questo caso.



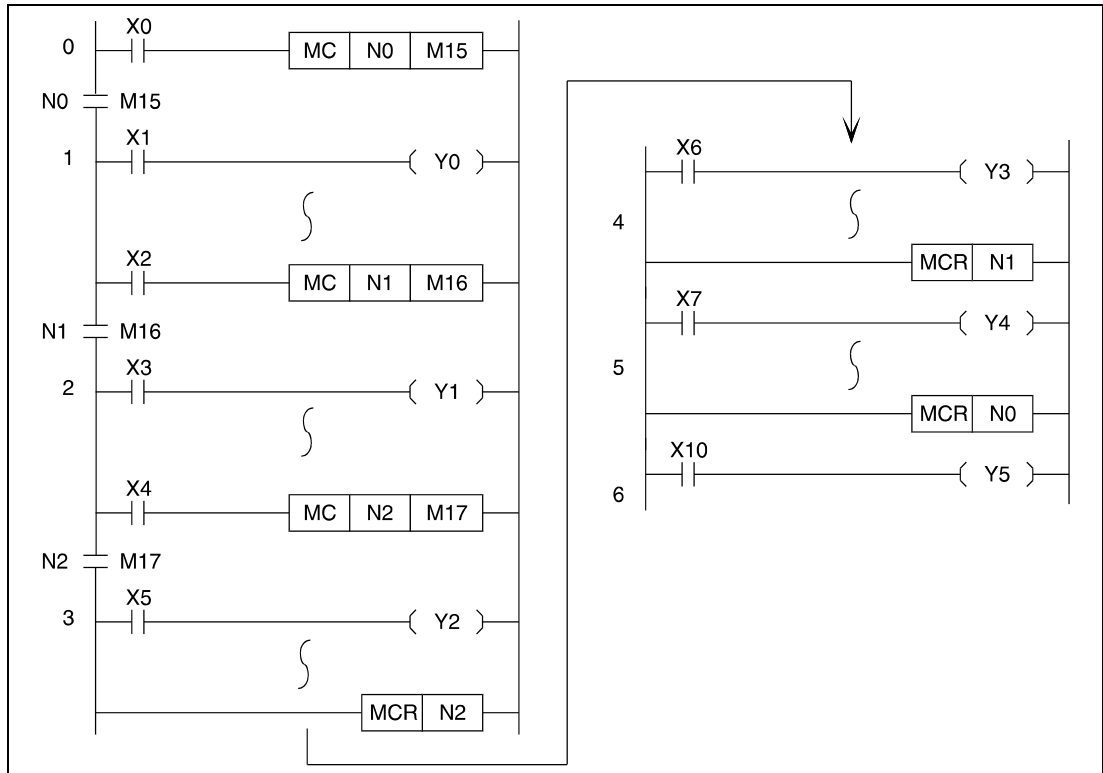
Se diverse istruzioni MCR vengono programmate consecutivamente, il programma può essere abbreviato inserendo solo una istruzione MCR con l'indirizzo di annidamento più basso, per chiudere tutte le parti di programma MC.



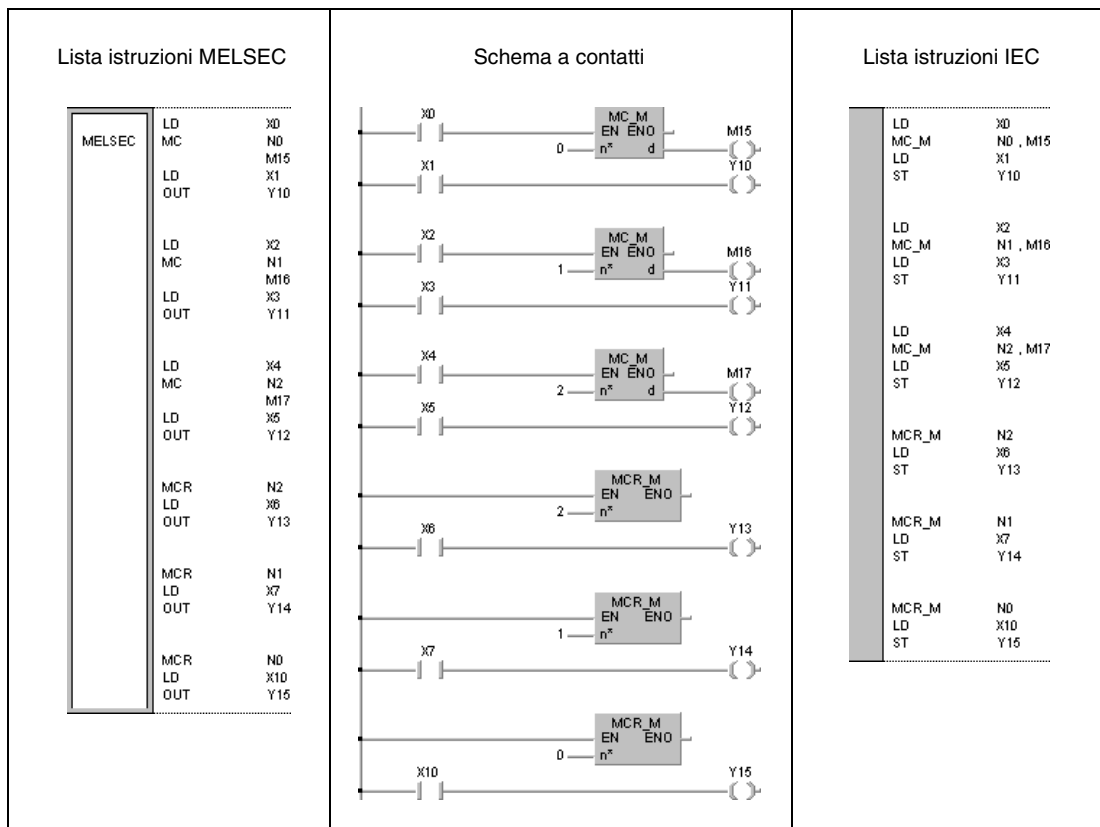
Esempio di programma MC, MCR

L'istruzione MC indica un indirizzo di annidamento N che specifica il livello. Gli indirizzi di annidamento possono essere definiti nel campo da N0 a N14 per la serie Q e il System Q, oppure fra N0 e N7 per la serie A.

Gli indirizzi di annidamento determinano la sequenza di esecuzione delle parti di programma MC. Il programma seguente illustra la designazione di livelli di esecuzione diversi tramite indirizzi di annidamento. Per una migliore comprensibilità, viene mostrato lo schema a contatti di GX Developer:



Viene inoltre mostrato lo schema a contatti di GX IEC Developer.



Funzioni Fine del programma principale

FEND Fine diramazione di programma

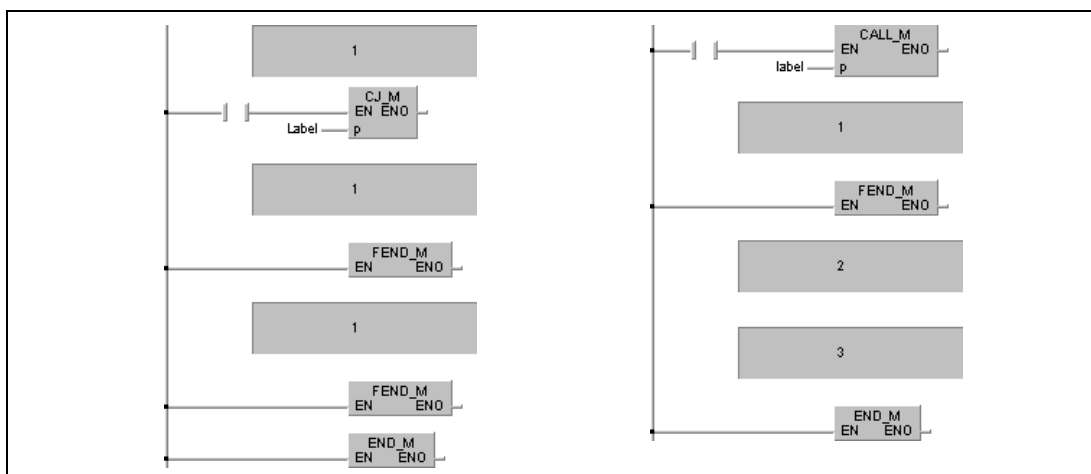
L'istruzione FEND contrassegna la fine di un ramo di programma. Questo ramo può appartenere ad un programma principale o a un sottoprogramma.

Dopo l'esecuzione di una istruzione FEND, il programma salta all'istruzione END. L'esecuzione di processi interni come l'elaborazione di timer o contatori, oppure l'autodiagnosi della CPU inizia nuovamente con il passo 1 del programma.

L'esempio di programma sulla sinistra mostra la terminazione di rami di programma richiamati tramite l'istruzione CJ (salto condizionato).

Dopo l'esecuzione dell'istruzione CJ, la parte di programma richiamata viene eseguita fino alla prima istruzione FEND. Se non viene prima eseguita una istruzione CJ, dopo la prossima istruzione FEND il programma torna al passo 0.

Il programma di esempio sulla destra mostra l'uso dell'istruzione FEND per separare un programma principale da un sottoprogramma o da un programma in interruzione.



1 Programma principale

2 Sottoprogramma

3 Programma di interruzione

NOTA

Nella lista di istruzioni di GX Developer, l'istruzione FEND deve essere programmata esplicitamente dall'utente. Al termine dell'elaborazione di questa unità organizzativa di programma, le successive non verranno eseguite, in quanto successive all'istruzione FEND.

In alternativa a questa programmazione, è possibile utilizzare l'editor IEC. In questo caso l'istruzione FEND viene inserita automaticamente dal compilatore di GX IEC Developer.

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- L'istruzione FEND viene eseguita dopo una istruzione CALL, FCALL, ECALL o EFCALL e prima dell'istruzione RET (serie Q e System Q = codice di errore 4211).
- L'istruzione FEND viene eseguita dopo una istruzione FOR e prima di una istruzione NEXT (serie Q e System Q = codice di errore 4200).
- L'istruzione FEND viene eseguita durante un programma di interruzione e prima di una istruzione IRET (serie Q e System Q = codice di errore 4221).
- L'istruzione FEND viene eseguita dopo una istruzione CHKCIR e prima di una istruzione CHKEND (serie Q e System Q = codice di errore 4230).
- L'istruzione FEND viene eseguita dopo una istruzione IX e prima di una istruzione IXEND (serie Q e System Q = codice di errore 4231).

5.6.2 END

NOTA Queste istruzioni non dovrebbero essere utilizzate con gli editor IEC.

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●


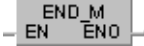
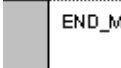
Operandi MELSEC A

Operandi utilizzabili														Definizione cifra	Numero di passi	Indice	Segnale di errore	Segnale di errore							
Operandi a bit						Operandi a word (16-bit)						Costanti							Puntatore	Livello					
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1				Z	V	K	H (16#)	P	I	N	M9012	M9010 M9011
																							1		●

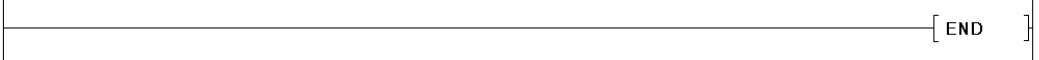
Operandi MELSEC Q

Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word				U			
—	—	—	—	—	—	—	—	—	—	SM0	1

GX IEC Developer:

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
 END		

GX Developer:


--

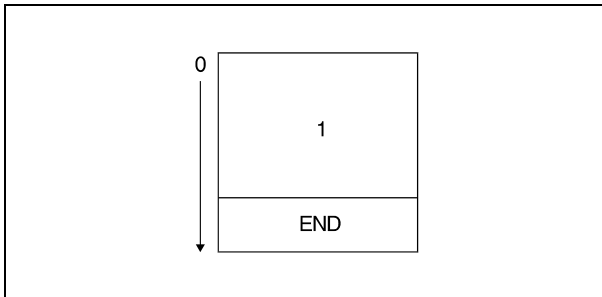
Variabili

Dati impostati	Significato	Tipo dati
—	—	—

Funzioni **Terminazione del programma sequenza**

END **Terminazione del programma sequenza**

L'istruzione END contrassegna la fine di un programma. L'esecuzione dell'istruzione END comporta un salto indietro al passo di programma 0.

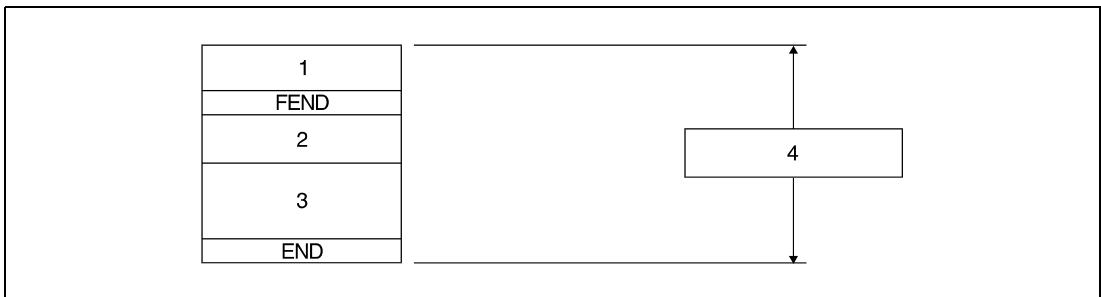


¹ Programma sequenza

L'istruzione END non può essere utilizzata in un sottoprogramma. Un sottoprogramma deve terminare con l'istruzione FEND.

Se l'istruzione END non esiste nel programma, viene restituito un codice di errore all'avviamento del programma e la sua esecuzione viene sospesa dal PLC. Senza l'istruzione END si verifica un errore di esecuzione anche se la capacità di un sottoprogramma viene impostata da parametri.

Lo schema seguente mostra la programmazione corretta delle istruzioni END e FEND:



- ¹ Programma principale
- ² Sottoprogramma
- ³ Programma di interruzione
- ⁴ Programma sequenza

NOTA *L'istruzione FEND viene inserita automaticamente da GX IEC Developer e GX Developer.*

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- La destinazione di salto di una istruzione CJ, SCJ o JMP si trova oltre l'istruzione END.
- Viene richiamato un sottoprogramma o un programma di interruzione posto dopo l'istruzione END.
- L'istruzione END viene eseguita dopo una istruzione CALL, FCALL, ECALL o EFCALL e prima dell'istruzione RET (serie Q e System Q = codice di errore 4211).
- L'istruzione END viene eseguita dopo una istruzione FOR e prima di una istruzione NEXT (serie Q e System Q = codice di errore 4200).
- L'istruzione END viene eseguita durante un programma di interruzione e prima di una istruzione IRET (serie Q e System Q = codice di errore 4221).
- L'istruzione END viene eseguita dopo una istruzione CHKCIR e prima di una istruzione CHKEND (serie Q e System Q = codice di errore 4230).
- L'istruzione END viene eseguita dopo una istruzione IX e prima di una istruzione IXEND (serie Q e System Q = codice di errore 4231).

5.7 Istruzioni miscellanee

5.7.1 STOP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

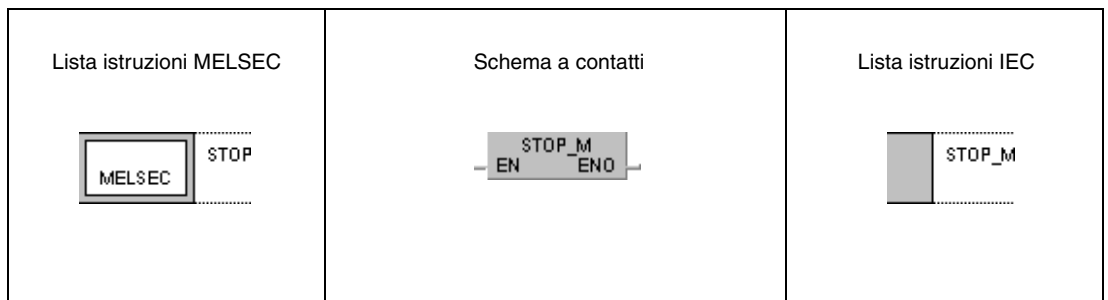
Operandi MELSEC A

Operandi utilizzabili														Definizione cifra	Numero di passi	Indice	Segnale di errore	Segnale di errore						
Operandi a bit				Operandi a word (16-bit)						Costanti	Punta-tore	Livello												
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1						Z	V	K	H (16#)	P	I
																					1		M9012	M9010 M9011

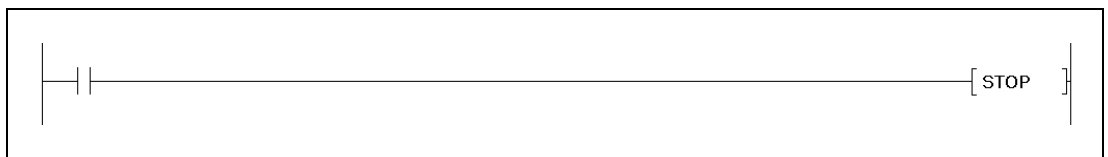
Operandi MELSEC Q

Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word				U			
—	—	—	—	—	—	—	—	—	—	SM0	1

GX IEC Developer:



GX Developer:



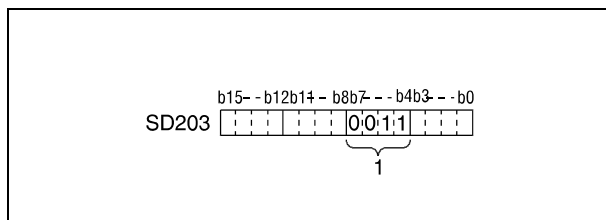
Variabili

Dati impostati	Significato	Tipo dati
—	—	—

Funzioni **Arresto programma sequenza****STOP** **Istruzione di arresto**

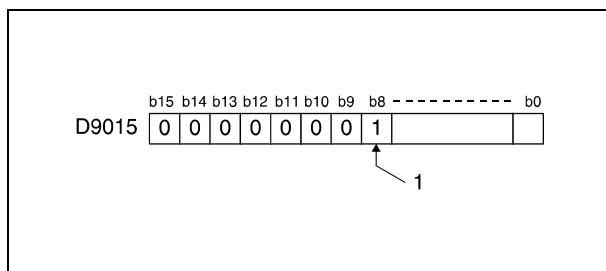
Se la condizione d'ingresso dell'istruzione STOP è attiva, tutte le uscite (Y) sono azzerate e tutte le operazioni del PLC vengono terminate. L'istruzione STOP svolge la stessa funzione della posizione STOP della chiavetta RUN/STOP della CPU.

A seguito dell'esecuzione dell'istruzione STOP in una CPU serie Q o System Q, i bit dal quinto all'ottavo (da b4 a b7) del registro speciale SD203 contengono il valore binario 3.



¹ Valore binario 3

Dopo l'esecuzione dell'istruzione STOP in una CPU serie A, il nono bit (b8) del registro speciale D9015 viene attivato (1).



¹ Il bit è attivo (1)

Per riavviare il funzionamento del PLC, il commutatore RUN/STOP deve essere portato su STOP e poi di nuovo su RUN.

Portando la levetta RESET su LATCH CLEAR dopo l'esecuzione di una istruzione STOP, il contenuto del buffer di memoria non viene modificato. Per azzerare il buffer di memoria, è necessario portare prima il commutatore RUN/STOP su STOP, poi la levetta RESET su LATCH CLEAR.

Errori di esecuzione

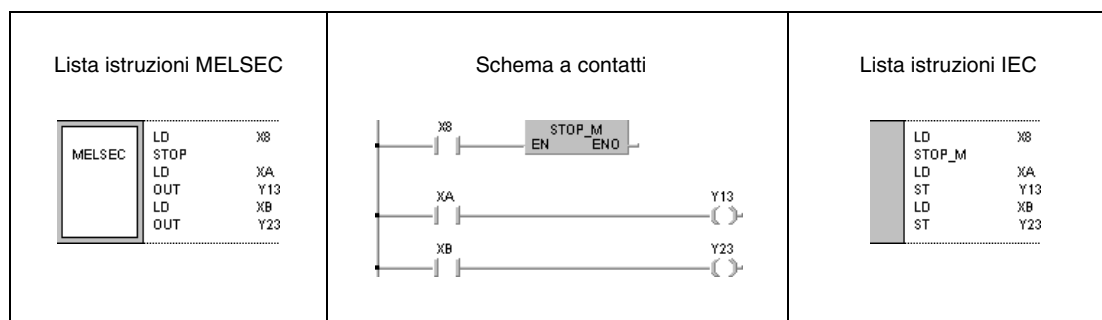
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- L'istruzione END viene eseguita dopo una istruzione CALL, FCALL, ECALL o EFCALL e prima dell'istruzione RET (serie Q e System Q = codice di errore 4211).
- L'istruzione END viene eseguita dopo una istruzione FOR e prima di una istruzione NEXT (serie Q e System Q = codice di errore 4200).
- L'istruzione END viene eseguita durante un programma di interruzione e prima di una istruzione IRET (serie Q e System Q = codice di errore 4221).
- L'istruzione END viene eseguita dopo una istruzione CHKCIR e prima di una istruzione CHKEND (serie Q e System Q = codice di errore 4230).
- L'istruzione END viene eseguita dopo una istruzione IX e prima di una istruzione IXEND (serie Q e System Q = codice di errore 4231).

Esempio di programma

STOP

Se X8 è attivo, il programma seguente termina. Tutti i passi di programma successivi sono eseguiti dopo aver portato il commutatore RUN/STOP su STOP e poi di nuovo su RUN.



5.7.2 NOP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●


Operandi MELSEC A

Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Segnale di errore M9012	Segnale di errore M9010 M9011						
Operandi a bit						Operandi a word (16-bit)						Costanti		Puntatore							Livello					
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V						K	H (16#)	P	I	N	
																						1				

Operandi MELSEC Q

Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word							
—	—	—	—	—	—	—	—	—	—	—	1

GX IEC Developer:

Lista istruzioni MELSEC 	Schema a contatti	Lista istruzioni IEC
--	-------------------	----------------------

GX Developer:

--

Variabili

Dati impostati	Significato	Tipo dati
—	—	—

Funzioni **Passo di programma nessuna operazione**

NOP **Passo di programma nessuna operazione**

L'istruzione NOP è una istruzione senza funzioni, che non influenza nessun'altra operazione o parte di programma. L'istruzione NOP crea un passo di programma vuoto che può essere sostituito da altre istruzioni durante lo sviluppo di un nuovo programma.

L'istruzione NOP è specialmente utile nei casi seguenti:

Per creare spazio per il debug di programmi sequenza.

Per cancellare una istruzione (sovrascrivendola) senza modificare il numero di passi.

Per cancellare temporaneamente una istruzione per una modifica successiva.

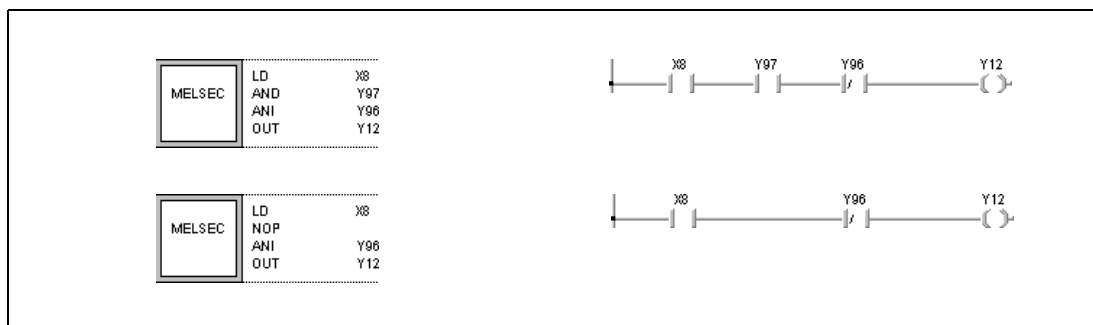
NOTA

Dopo aver completato l'editing del programma, le istruzioni NOP dovrebbero essere cancellate (dove possibile) per abbreviare il tempo di scansione del programma.

Programma di esempio 1

NOP

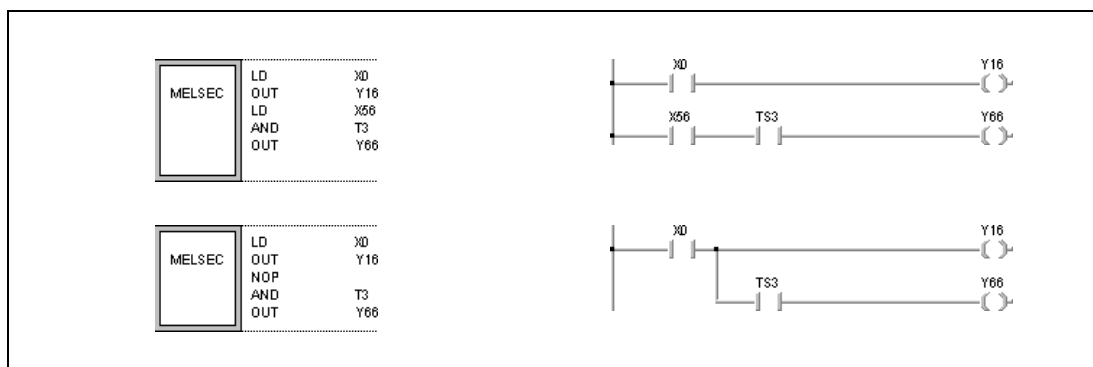
Il programma seguente contiene una istruzione NOP che sostituisce il collegamento AND a scopo di debug.



Programma di esempio 2

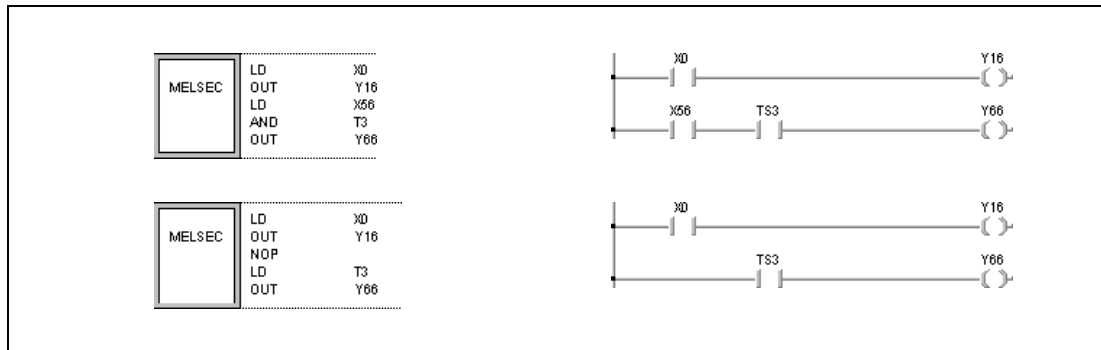
NOP

Il programma di esempio seguente contiene una istruzione NOP che sostituisce una istruzione LD.



Programma di esempio 3 NOP

Il programma di esempio seguente contiene una istruzione NOP che sostituisce una istruzione LD.



NOTA

Usare estrema cautela nella sostituzione di contatti di ingresso (LD, LDI) con istruzioni NOP, dato che la struttura logica del programma viene modificata considerevolmente.

6 Istruzioni applicative, parte 1

Le istruzioni applicative, parte 1, comprendono istruzioni che elaborano dati numerici a 16 e 32-bit, dati in virgola mobile e dati a stringhe di caratteri. Normalmente, queste istruzioni di base eseguono operazioni di confronto e operazioni aritmetiche.

Istruzione	Significato
Istruzioni per operazioni di confronto	Confronto di dati fra loro (ad es. =, >, ≥)
Istruzioni per operazioni aritmetiche	Somma, sottrazione, moltiplicazione, divisione, incremento e decremento di dati BIN e BCD, dati in virgola mobile e blocchi dati BIN Collegamento stringhe di caratteri
Istruzioni di conversione dati	Conversione fra tipi di dati (ad es. BCD -> BIN, BIN -> BCD)
Istruzioni per trasferimento dati	Trasmette un insieme di dati definiti
Istruzioni di salto di programma	Comandi di salto di programma
Istruzioni per controllo esecuzione del programma	Abilita e disabilita gli interrupt di programma
Istruzione di rinfresco	Rinfresca operandi a bit, collegamenti e interfacce di I/O
Altre istruzioni utili	Conteggio avanti e indietro di ingressi single ended o differenziali, timer ad apprendimento, timer per funzioni speciali, controllo rotazione tavola girevole con scelta percorso più breve, segnale di rampa, misura densità impulsi, uscita impulsiva a ciclo fisso, modulazione larghezza impulso, ingresso matrice

6.1 Istruzioni per operazioni di confronto

Le istruzioni per operazioni di confronto eseguono confronti fra valori di dati (ad es. uguale =, maggiore >, minore <). La programmazione delle istruzioni di confronto è simile alle istruzioni di base corrispondenti:

LD, LDI ⇒ LD=, LDD=

AND, ANI ⇒ AND=, ANDD=

OR, ORI ⇒ OR=, ORD=

Funzione	MELSEC Istruzione con Editor MELSEC	MELSEC Istruzione con Editor IEC	Funzione	MELSEC Istruzione con Editor MELSEC	MELSEC Istruzione con Editor IEC
= uguale	LD=	LD_EQ_M	≤ minore o uguale	LD<=	LD_LE_M
	AND=	AND_EQ_M		AND<=	AND_LE_M
	OR=	OR_EQ_M		OR<=	OR_LE_M
	LDD=	LDD_EQ_M		LDD<=	LDD_LE_M
	ANDD=	ANDD_EQ_M		ANDD<=	ANDD_LE_M
	ORD=	ORD_EQ_M		ORD<=	ORD_LE_M
	LDE=	LD_EEQ_M		LDE<=	LD_ELE_M
	ANDE=	AND_EEQ_M		ANDE<=	AND_ELE_M
	ORE=	OR_EEQ_M		ORE<=	OR_ELE_M
	LD\$=	LD_STRING_EQ_M		LD\$<=	LD_STRING_LE_M
	AND\$=	AND_STRING_EQ_M		AND\$<=	AND_STRING_LE_M
	OR\$=	OR_STRING_EQ_M		OR\$<=	OR_STRING_LE_M
	BKCOMP=	BKCOMP_EQ_M		BKCOMP<=	BKCOMP_LE_M
	BKCOMP=P	BKCOMP_EQP_M		BKCOMP<=P	BKCOMP_LEP_M
≠ diverso	LD<>	LD_NE_M	< minore	LD<	LD_LT_M
	AND<>	AND_NE_M		AND<	AND_LT_M
	OR<>	OR_NE_M		OR<	OR_LT_M
	LDD<>	LDD_NE_M		LDD<	LDD_LT_M
	ANDD<>	ANDD_NE_M		ANDD<	ANDD_LT_M
	ORD<>	ORD_NE_M		ORD<	ORD_LT_M
	LDE<>	LD_ENE_M		LDE<	LD_ELT_M
	ANDE<>	AND_ENE_M		ANDE<	AND_ELT_M
	ORE<>	OR_ENE_M		ORE<	OR_ELT_M
	LD\$<>	LD_STRING_NE_M		LD\$<	LD_STRING_LT_M
	AND\$<>	AND_STRING_NE_M		AND\$<	AND_STRING_LT_M
	OR\$<>	OR_STRING_NE_M		OR\$<	OR_STRING_LT_M
	BKCOMP<>	BKCOMP_NE_M		BKCOMP<	BKCOMP_LT_M
	BKCOMP<>P	BKCOMP_NEP_M		BKCOMP<P	BKCOMP_LTP_M

Funzione	MELSEC Istruzione con Editor MELSEC	MELSEC Istruzione con Editor IEC	Funzione	MELSEC Istruzione con Editor MELSEC	MELSEC Istruzione con Editor IEC
> maggiore	LD>	LD_GT_M	≥ maggiore o uguale	LD>=	LD_GE_M
	AND>	AND_GT_M		AND>=	AND_GE_M
	OR>	OR_GT_M		OR>=	OR_GE_M
	LDD>	LDD_GT_M		LDD>=	LDD_GE_M
	ANDD>	ANDD_GT_M		ANDD>=	ANDD_GE_M
	ORD>	ORD_GT_M		ORD>=	ORD_GE_M
	LDE>	LD_EGT_M		LDE>=	LD_EGE_M
	ANDE>	AND_EGT_M		ANDE>=	AND_EGE_M
	ORE>	OR_EGT_M		ORE>=	OR_EGE_M
	LD\$>	LD_STRING_GT_M		LD\$>=	LD_STRING_GE_M
	AND\$>	AND_STRING_GT_M		AND\$>=	AND_STRING_GE_M
	OR\$>	OR_STRING_GT_M		OR\$>=	OR_STRING_GE_M
	BKCMP>	BKCMP_GT_M		BKCMP>=	BKCMP_GE_M
	BKCMP>P	BKCMP_GTP_M		BKCMP>=P	BKCMP_GEP_M

NOTA

Per le istruzioni relative alle operazioni di confronto, dei comandi di confronto con la stessa funzionalità sono disponibili nella libreria standard IEC di GX IEC Developer.

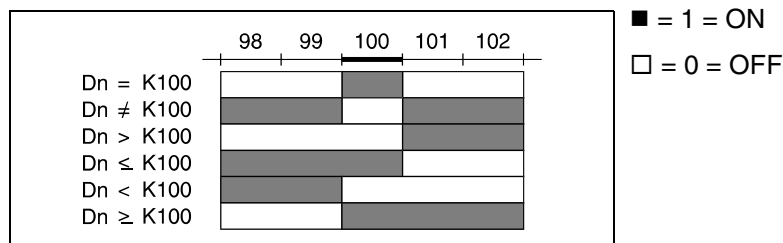
Comando IEC

Funzione	Comando IEC	Significato
=	EQ	Uguale
<>	NE	Diverso
<=	LE	Minore o uguale
<	LT	Minore
>=	GE	Maggiore o uguale
>	GT	Maggiore

Utilizzare comandi IEC con gli editor IEC

Condizione di esecuzione

La figura seguente mostra le condizioni di esecuzione per la diverse istruzioni di confronto.



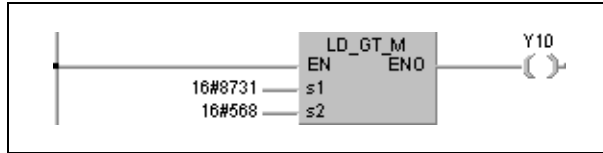
NOTA

Ai fini del confronto, le istruzioni di confronto leggono tutti i tipi dati indicati come valori BIN negativi. Le istruzioni di confronto considerano qualsiasi dato indicato come dato binario.

Il risultato dell'operazione di confronto $16\#8731 > 16\#7999$ è FALSE (0), anche se il risultato atteso sarebbe TRUE (1). I valori sono convertiti in dati BIN, e quindi il bit 15 (b15) è impostato a 1. Se il bit 15 è a 1, il valore diventa negativo.

Programma di esempio 1

Confronto di due valori BCD a due cifre:

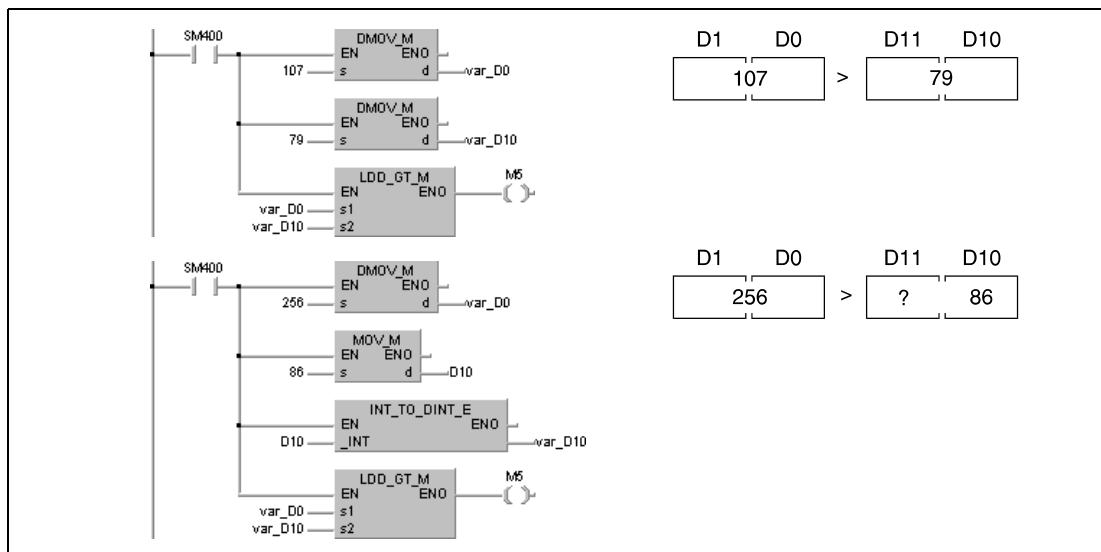


8731_H viene considerato come -30927 e 568_H come 1384 . L'operazione di confronto è quindi $-30927 > 1384$ e Y10 non si attiva.

Nelle operazioni di confronto con dati a 32 bit, il valore numerico d'ingresso deve essere determinato da una istruzione a 32 bit, ad esempio DMOV. L'istruzione non viene eseguita correttamente se il valore viene determinato da una istruzione a 16 bit come ad esempio MOV, dato che l'istruzione a 32 bit si applica sempre al dato contenuto in n e (n+1).

Programma di esempio 2

Istruzioni di confronto con dati a 32 bit:



L'esempio mostra due operazioni di confronto con dati a 32 bit. Il primo programma eccita M5, dato che entrambi i valori sono determinati dall'istruzione a 32 bit DMOV.

Il secondo programma non ha un risultato definito, dato che il valore nei byte più pesanti è indeterminato.

NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.1.1 =, < >, >, < =, <, > =

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

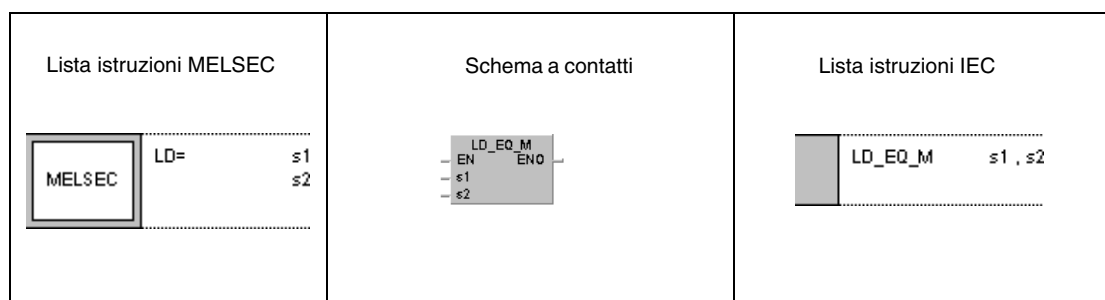
	Operandi utilizzabili														Definizione cifra	Numero di passi	Indice	Rapporto	Errore								
	Operandi a bit				Operandi a word (16-bit)				Costanti		Puntatore		Livello														
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1						Z	V	K	H (16#)	P	I	N	
s1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●					K1 ↓ K4	5/7	●		●	
s2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●					●	1			●

¹ Il numero di passi è 7, se la funzione di indicizzazione non è attiva, la cifra indicata di un operando a bit non è K4 e l'indirizzo di partenza di un operando a bit non è multiplo di 8 (o 16 per le CPU A3H, A3M, AnA, AnAS e AnU). Consultare la sezione "Programmazione con CPU AnA, AnAS e AnU" di questo manuale di programmazione per il corrispondente numero di passi.

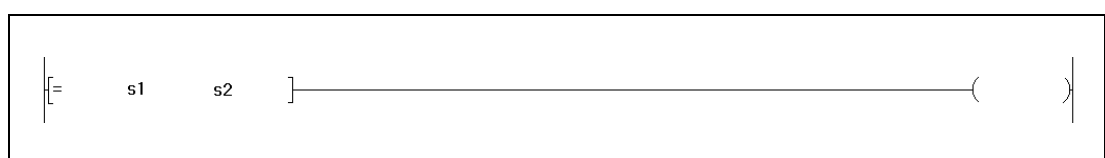
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operando interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	●	●	●	●	●	●	●	●	—	—	3
s2	●	●	●	●	●	●	●	●	—	—	

GX IEC Developer:



GX Developer:



Variabili

Dati impostati	Significato	Tipo dati
s1	Dato di confronto o operando che contiene il dato di confronto	BIN 16-bit
s2		

Funzioni Confronto dati BIN a 16-bit

=, <, >, <=, <, >= Istruzioni per operazioni di confronto

Una istruzione di confronto è composta dall'istruzione vera e propria e dai due operandi s1 e s2 che devono essere confrontati.

Il risultato dell'operazione di confronto è trattato come un contatto NO (normalmente aperto).

I risultati delle operazioni di confronto per le singole istruzioni sono riportati di seguito:

Simbolo istruzione	Risultato operazioni di confronto	
	1	0
=	s1 = s2	s1 ≠ s2
<>	s1 ≠ s2	s1 = s2
>	s1 > s2	s1 ≤ s2
<=	s1 ≤ s2	s1 > s2
<	s1 < s2	s1 ≥ s2
>=	s1 ≥ s2	s1 < s2

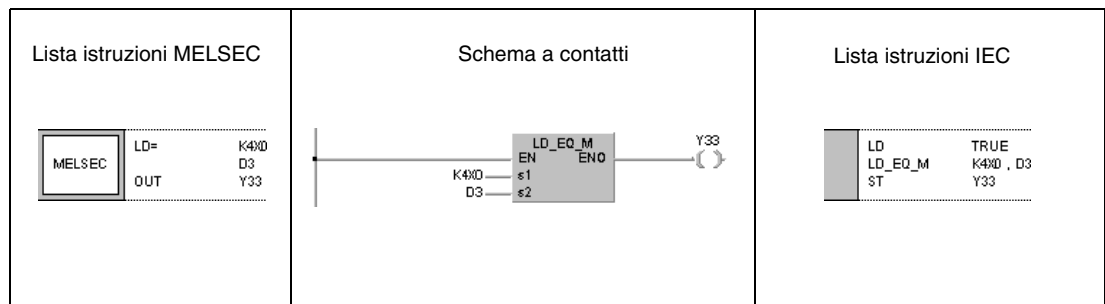
Ai fini del confronto, le istruzioni di confronto leggono tutti i tipi dati indicati come valori BIN negativi.

Il risultato dell'operazione di confronto 16#8000 > 16#7999 è FALSE (0), anche se il risultato atteso sarebbe TRUE (1). I valori sono convertiti in dati BIN, e quindi il bit 15 (b15) è impostato a 1. Se il bit 15 è a 1, il valore diventa negativo.

Programma di esempio 1

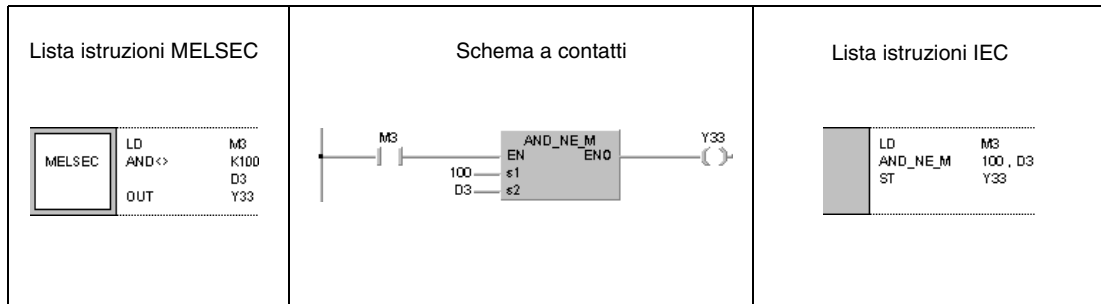
Istruzioni per confronto su uguale

Il programma che segue confronta i dati da X0 a XF con il dato in D3. Eccita Y33 se i dati sono uguali.

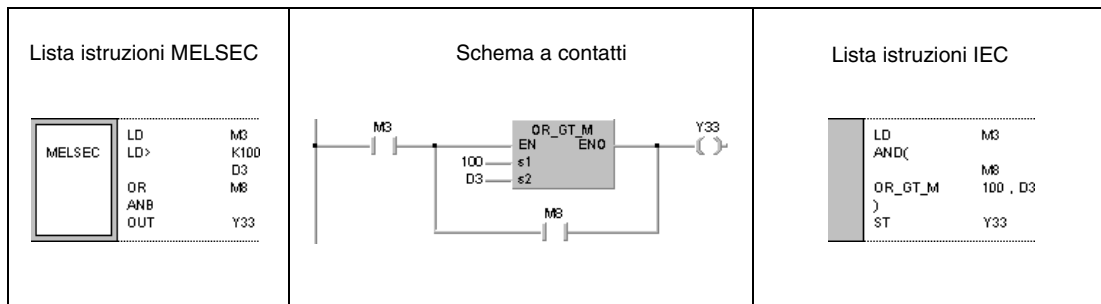


Programma di esempio 2 Istruzioni per confronto <> (diverso)

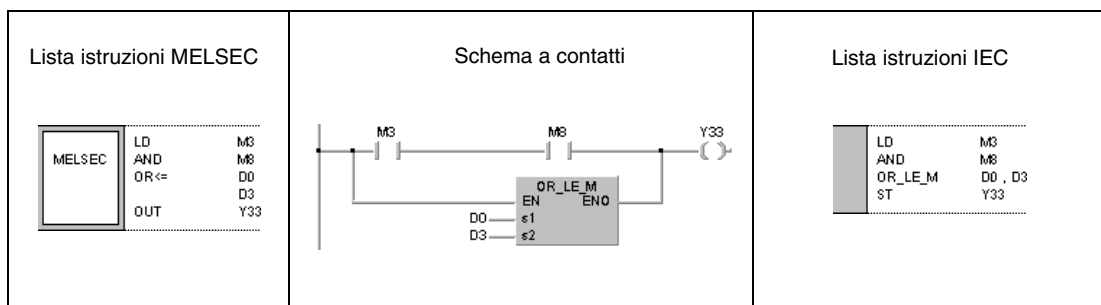
Il programma che segue confronta il dato BIN 100 con il dato in D3. Eccita Y33 se il dato in D3 è diverso da 100.

**Programma di esempio 3** Istruzioni per confronto > (diverso)

Il programma che segue confronta il dato BIN 100 con il dato in D3. Eccita Y33 se il dato in D3 è inferiore a 100 e M3 è ON. Y33 si eccita anche se M8 e M3 sono ON.

**Programma di esempio 4** Istruzioni per confronto su <=

Il programma che segue confronta il dato in D0 con il dato in D3. Eccita Y33 se il dato in D3 è inferiore o uguale a quello in D3. Y33 si eccita anche se M8 e M3 sono ON.



6.1.2 D=, D<>, D>, D<=, D<, D>=

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

Operandi															Definizione cifra	Numero di passi	Indice	Rapporto	Errore								
Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello													
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z						V	K	H (16#)	P	I	N		
s1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●					K1 ↓ K8	11	●			●
s2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●										

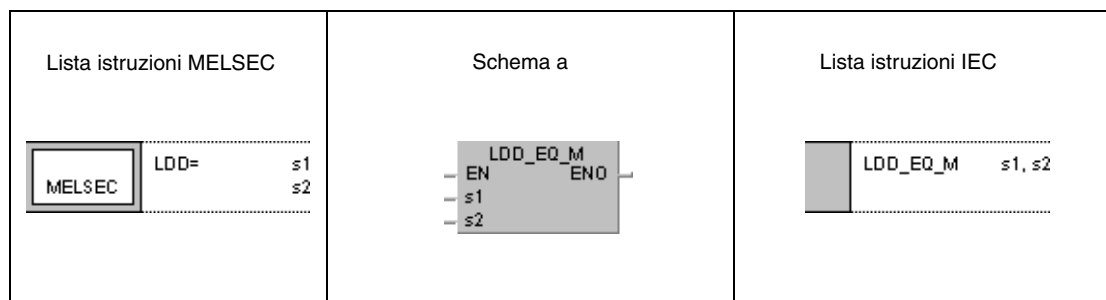
¹ Consultare la sezione "Programmazione con CPU AnA, AnAS e AnU" di questo manuale di programmazione per il corrispondente numero di passi.

Operandi MELSEC Q

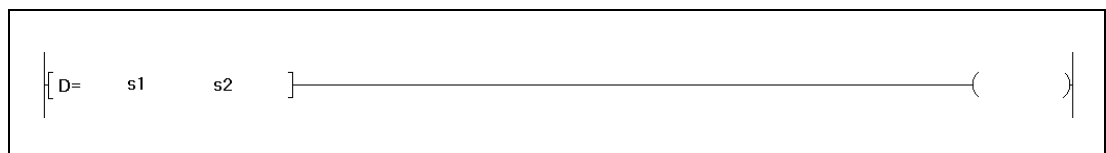
Operandi										Segnale di errore	Numero di passi
Operando interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word							
s1	●	●	●	●	●	●	●	●	—	—	3 ¹⁾
s2	●	●	●	●	●	●	●	●	—	—	

¹ Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.
 Se viene usata una CPU QnA oppure una CPU a processore singolo del System Q: 3
 Se viene usata una CPU multi processore del System Q con operandi interni a word (tranne file registri ZR): 5
 costanti: 5
 Operandi a bit con numero operando multiplo di 16, in cui la cifra indicata è K8, e che non usa indicizzazione: 5
 Se viene usata una CPU multiprocessore del System Q con operandi diversi da quelli menzionati: 3

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s1	Dato di confronto o operando che contiene il dato di confronto.	BIN 32-bit
s2		

Funzioni **Confronto dati BIN a 32-bit****D=, D<>, D>, D<=, D<, D>= istruzioni per operazioni di confronto**

Una istruzione di confronto a 32 bit è composta dall'istruzione vera e propria e dai due operandi s1 e s2 che devono essere confrontati.

Il risultato dell'operazione di confronto è trattato come un contatto NO (normalmente aperto). Il confronto viene eseguito su dati a 32 bit.

I risultati delle operazioni di confronto per le singole istruzioni sono riportati di seguito:

Simbolo istruzione	Risultato operazioni di confronto	
	1	0
D=	s1 = s2	s1 ≠ s2
D<>	s1 ≠ s2	s1 = s2
D>	s1 > s2	s1 ≤ s2
D<=	s1 ≤ s2	s1 > s2
D<	s1 < s2	s1 ≥ s2
D>=	s1 ≥ s2	s1 < s2

NOTA

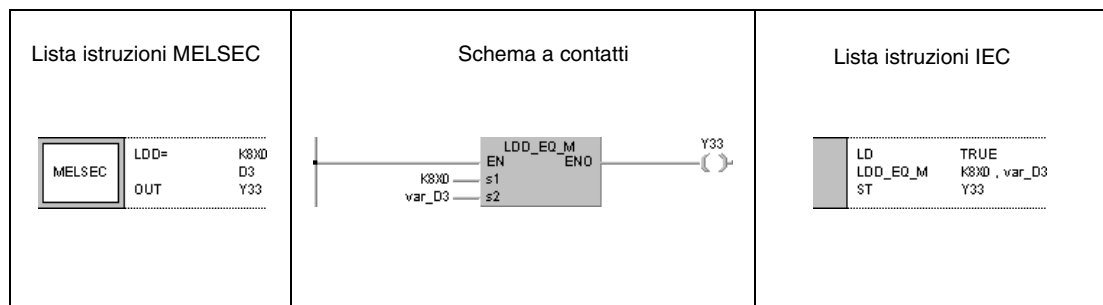
Ai fini del confronto, le istruzioni di confronto leggono tutti i tipi dati indicati come valori BIN negativi.

Il risultato dell'operazione di confronto $16\#8000 > 16\#7999$ è FALSE (0), anche se il risultato atteso sarebbe TRUE (1). I valori sono convertiti in dati BIN, e quindi il bit 15 (b15) è impostato a 1. Se il bit 15 è a 1, il valore diventa negativo.

Programma di esempio 1

Istruzioni per confronto su uguale D=

Il programma che segue confronta i dati da X0 a X1F con il dato in D3 e D4. Eccita Y33 se i dati sono uguali.



Programma di esempio 2

Istruzioni per confronto D<> (diverso)

Il programma che segue confronta il dato BIN 38000 con il dato in D3 e D4. Eccita Y33 se M3 è ON ed il dato in D3 e D4 è diverso da 38000.

<p>Lista istruzioni MELSEC</p> <pre> MELSEC LD M3 ANDD<> K38000 OUT D3 OUT Y33 </pre>	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p> <pre> LD M3 ANDD_NE_M 38000 , var_D3 ST Y33 </pre>
--	--------------------------	---

Programma di esempio 3

Istruzioni per confronto D> (diverso)

Il programma che segue confronta il dato BIN -80000 con il dato in D3 e D4. Eccita Y33 se M3 è ON ed il dato in D3 e D4 è inferiore a -80000. Y33 si eccita anche se M3 e M8 sono ON.

<p>Lista istruzioni MELSEC</p> <pre> MELSEC LD M3 LDD> K-80000 OR D3 ANB M8 OUT Y33 </pre>	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p> <pre> LD M3 AND(M8 ORD_GT_M K-80000 , var_D3) ST Y33 </pre>
---	--------------------------	---

Programma di esempio 4

Istruzioni per confronto su uguale D<=

Il programma che segue confronta il dato in D0 e D1 con il dato in D3 e D4. Y33 si eccita se il dato in D3 e D4 è maggiore o uguale a quello in D0 e D1. Y33 si eccita anche se M8 e M3 sono ON.

<p>Lista istruzioni MELSEC</p> <pre> MELSEC LD M3 AND M8 ORD<= D0 OUT D3 OUT Y33 </pre>	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p> <pre> LD M3 AND M8 ORD_LE_M var_D0 , var_D3 ST Y33 </pre>
--	--------------------------	---

NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per ulteriori informazioni, vedi il capitolo 3.5.2 di questo manuale.

6.1.3 E=, E<>, E>, E<=, E<, E>=**CPU**

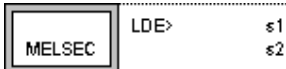
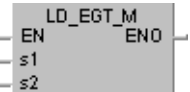
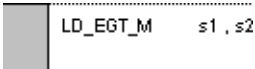
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

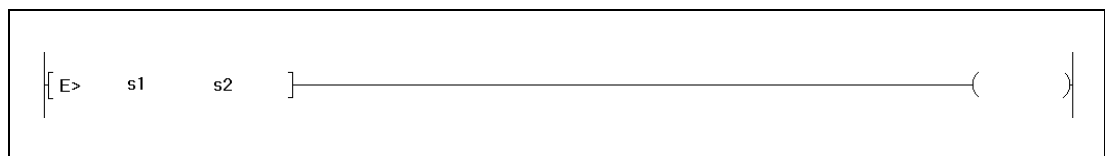
¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

	Operandi								Segnale di errore	Numero di passi	
	Operando interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti E			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	●	●	—	●	—	3	
s2	—	●	●	—	●	●	—	●	—		

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
		

GX Developer**Variabili**

Dati impostati	Significato	Tipo dati
s1	Dato di confronto o operando che contiene il dato di confronto.	Numero reale
s2		

Funzioni **Confronto dati in virgola mobile**

E=, E<>, E>, E<=, E<, E>= istruzioni per operazioni di confronto

Una istruzione di confronto per dati in virgola mobile è composta dall'istruzione vera e propria e dai due operandi s1 e s2 che devono essere confrontati.

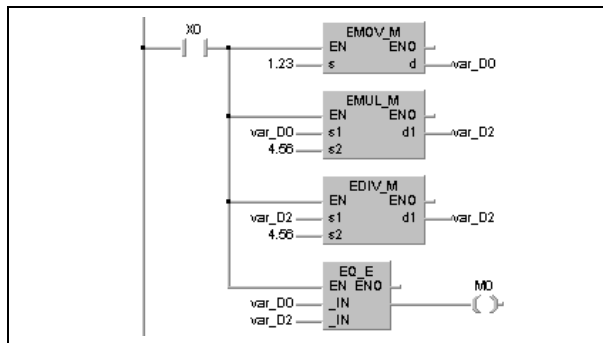
Il risultato dell'operazione di confronto è trattato come un contatto NO (normalmente aperto). Il confronto viene eseguito su dati in virgola mobile.

I risultati delle operazioni di confronto per le singole istruzioni sono riportati di seguito:

Simbolo istruzione	Risultato operazioni di confronto	
	1	0
E=	s1 = s2	s1 ≠ s2
E<>	s1 ≠ s2	s1 = s2
E>	s1 > s2	s1 ≤ s2
E<=	s1 ≤ s2	s1 > s2
E<	s1 < s2	s1 ≥ s2
E>=	s1 ≥ s2	s1 < s2

NOTA

In alcuni casi compaiono degli errori di arrotondamento ed i valori in virgola mobile che erano uguali prima dell'operazione di confronto non lo sono più dopo. Nell'esempio seguente M0 non viene attivato (ON):



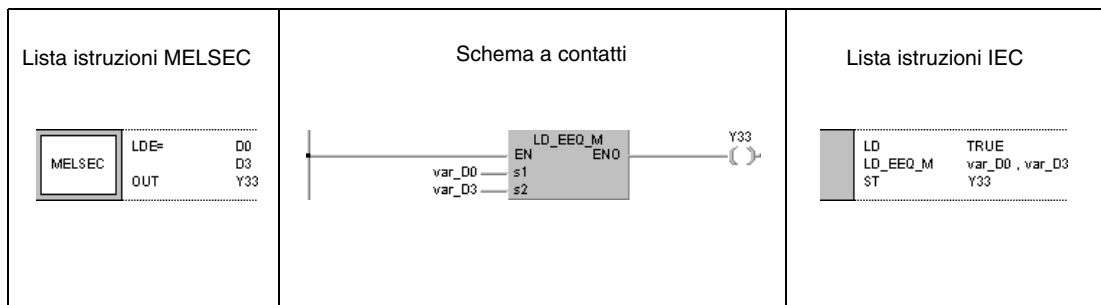
NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per ulteriori informazioni, vedi il capitolo 3.5.2 di questo manuale.

Programma di esempio 1

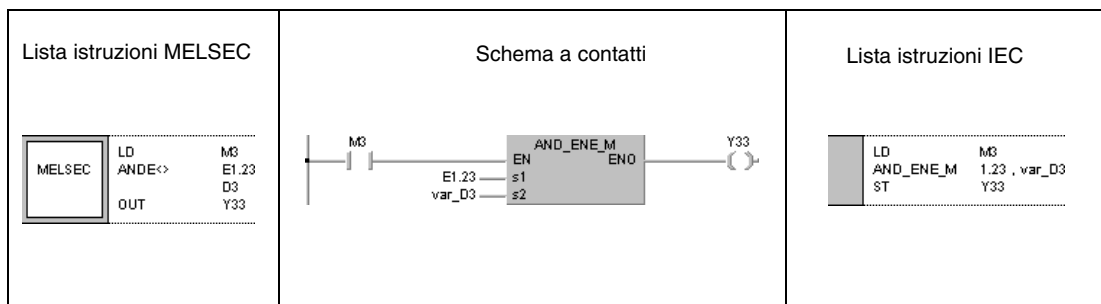
Istruzioni per confronto su uguale E=

Il programma che segue confronta il dato virgola mobile in D0 e D1 con il dato virgola mobile in D3 e D4. Eccita Y33 se i dati sono uguali.

**Programma di esempio 2**

Istruzioni per confronto E<> (diverso)

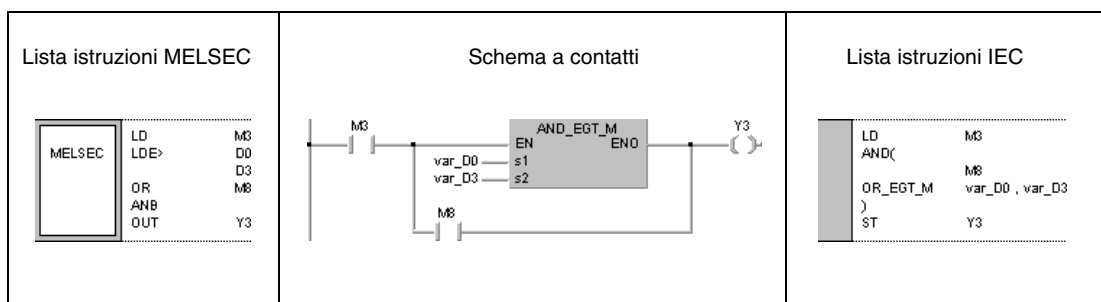
Il programma seguente confronta il numero in virgola mobile 1,23 con il numero in virgola mobile in D3 e D4. Eccita Y33 se M3 è ON ed il dato in D3 e D4 è diverso da 1.23.

**Programma di esempio 3**

Istruzioni per confronto E>

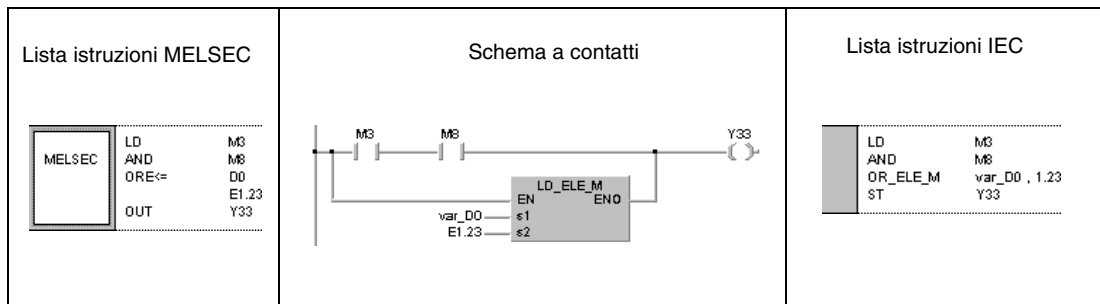
Il programma che segue confronta il dato virgola mobile in D0 e D1 con il dato virgola mobile in D3 e D4. Eccita Y3 se M3 è ON e il dato in D3 e D4 è inferiore al dato in D0 e D1.

Y3 si eccita anche se M3 e M8 sono ON.



Programma di esempio 4 Istruzioni per confronto E<=

L'esempio seguente confronta il numero in virgola mobile in D0 e D1 con il numero in virgola mobile 1,23. Eccita Y33 se il dato in D0 e D1 è inferiore o uguale a 1,23. Y33 si eccita anche se M8 e M3 sono ON.

**NOTA**

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per ulteriori informazioni, vedi il capitolo 3.5.2 di questo manuale.

6.1.4 \$ =, \$ < >, \$ >, \$ < =, \$ <, \$ > =

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi
MELSEC Q

	Operandi								Segnale di errore	Numero di passi	
	Operando interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti \$			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	●	—	SM0	3
s2	—	●	●	—	—	—	—	●	—		

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
<div style="border: 1px solid black; padding: 5px; width: fit-content;"> MELSEC </div> LD\$> s1 s2		LD_STRING_GT_M s1, s2

GX
Developer

[\$> s1 s2] ————— ()

Variabili

Dati impostati	Significato	Tipo dati
s1	Dato di confronto o operando che contiene il dato di confronto.	Stringa caratteri
s2		

Funzioni Confronto dati fra stringhe di caratteri

\$=, \$<>, \$>, \$<=, \$<, \$>= Istruzioni per operazioni di confronto

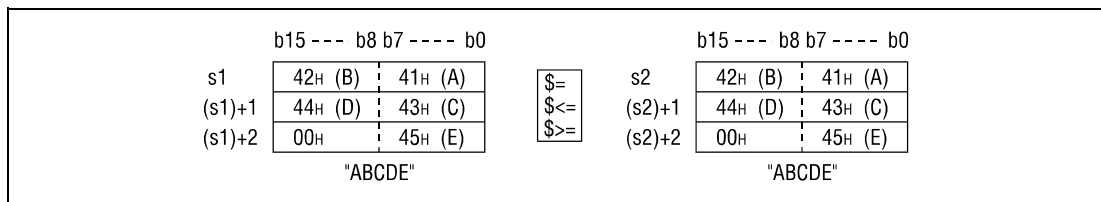
Una istruzione di confronto per dati a stringa di caratteri è composta dall'istruzione vera e propria e dai due operandi s1 e s2 che devono essere confrontati.

Il risultato dell'operazione di confronto è trattato come un contatto NO (normalmente aperto).

Il confronto viene eseguito sui dati della stringa di caratteri in codice ASCII, carattere per carattere, iniziando dal primo carattere della stringa.

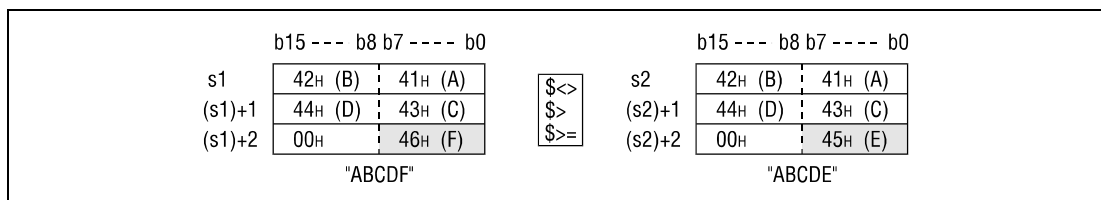
Le stringhe di caratteri s1 e s2 comprendono tutti i caratteri compresi fra il numero di operando specificato e l'operando che contiene il codice "00H".

Se tutti i caratteri della stringa soddisfano il confronto, il risultato per le istruzioni \$=, \$<=, \$>= è 1.



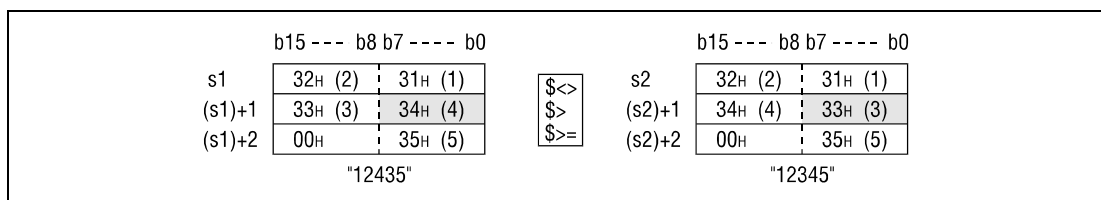
Se le stringhe di caratteri sono diverse, la stringa che contiene il codice di carattere maggiore viene considerata come maggiore.

Qui sotto, il risultato del confronto per le operazioni \$<>, \$>, \$>= è 1.



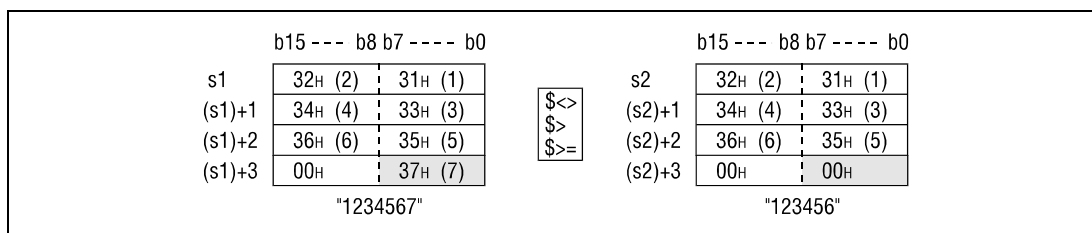
Se le stringhe di caratteri sono diverse, il primo codice di carattere diverso determina se la stringa è maggiore o minore.

Qui sotto, il risultato del confronto per le operazioni \$<>, \$>, \$>= è 1.



Se le stringhe di caratteri hanno lunghezze diverse, il dato con la stringa più lunga viene considerato come maggiore.

Qui sotto, il risultato del confronto per le operazioni \$ < >, \$ >, \$ > = è 1.



Errori di esecuzione

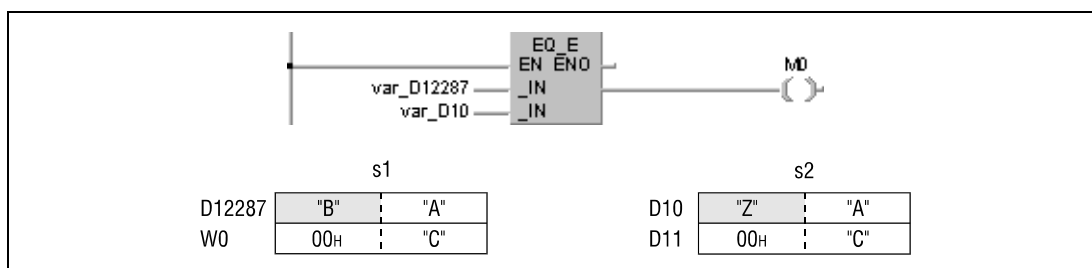
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il codice "00H" non esiste nel campo operandi utile per s1 e s2 (codice di errore: 4101).

NOTA

Le istruzioni per il confronto su stringhe di caratteri controllano anche il campo degli operandi.

Nei casi in cui una stringa di caratteri ecceda il campo dell'operando, i dati delle due stringhe vengono confrontati all'interno del campo utile, rilevando le differenze eventuali. Il risultato dell'operazione di confronto viene emesso senza segnalazione di errore.



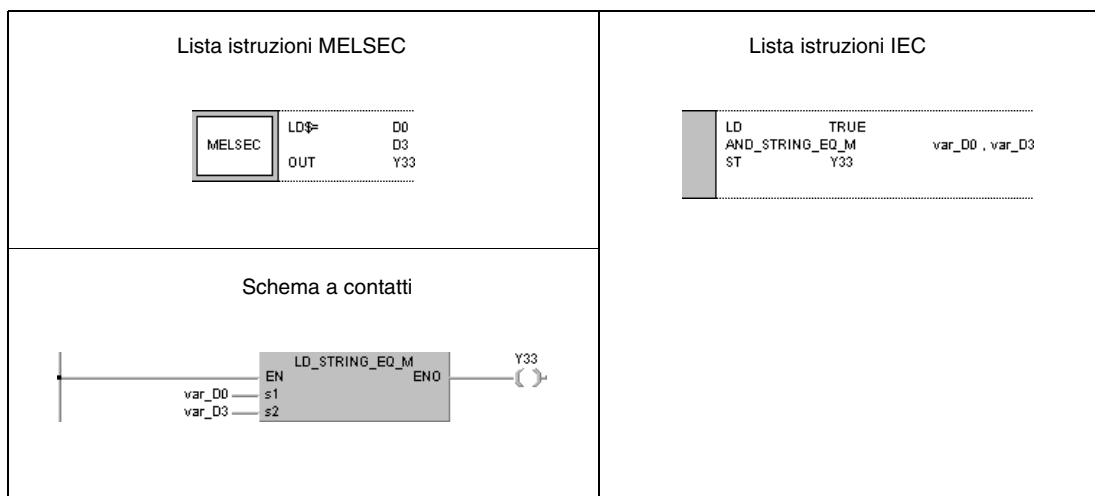
Nell'esempio precedente, la stringa s1 supera il campo dell'operando e i 16 bit più significativi (D12288) sono rinominati in W0. Tuttavia il risultato del confronto è 0, perché il secondo carattere di s1 viene rilevato come diverso da quello di s2. In questo caso non viene restituito alcun codice di errore relativo al campo dell'operando.

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

Programma di esempio 1

Istruzioni per confronto \$=

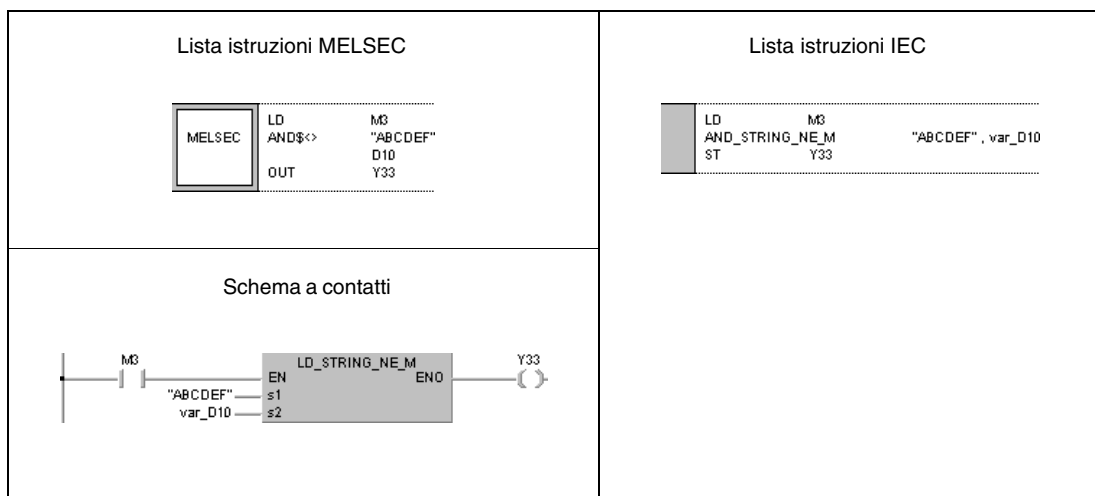
Il programma che segue confronta il dato a stringa in D0 con il dato a stringa in D3. Eccita Y33 se i dati sono uguali.



Programma di esempio 2

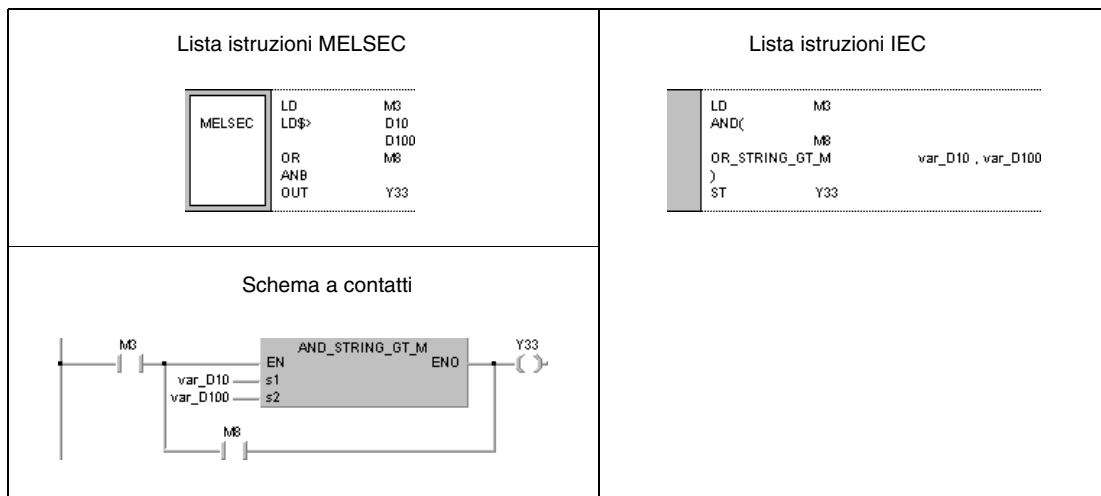
Istruzioni per confronto \$<> (diverso)

Il programma seguente confronta la stringa "ABCDEF" con il dato a stringa contenuto in D10. Eccita Y33 se i dati sono diversi.

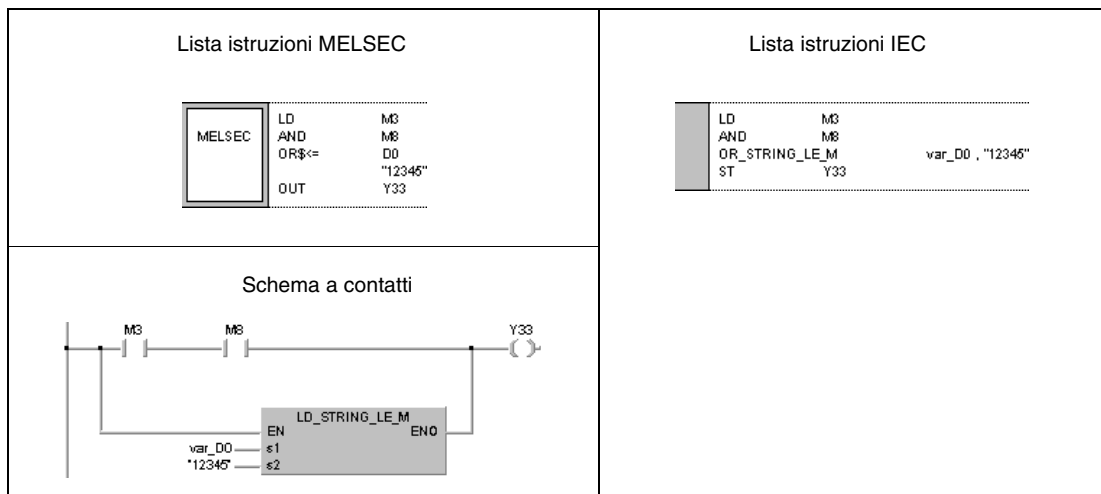


Programma di esempio 3 Istruzioni per confronto \$> (diverso)

Il programma che segue confronta il dato a stringa in D10 con il dato a stringa in D100. Eccita Y33 se il dato a stringa di caratteri in D10 è il maggiore.

**Programma di esempio 4** Istruzioni per confronto su \$<=

Il programma che segue confronta il dato a stringa in D0 con la stringa di caratteri "12345". Eccita Y33 se il dato in D0 è inferiore o uguale a "12345".

**NOTA**

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.1.5 BKCMP, BKCMPP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

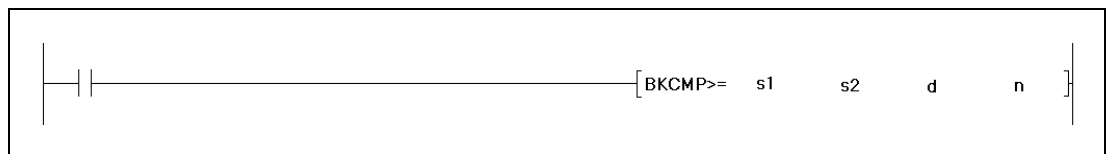
Operandi MELSEC Q

	Operandi								Segnale di errore	Numero di passi	
	Operando interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	●	—	SM0	5
s2	—	●	●	—	—	—	—	—	—		
d	●	●	●	—	—	—	—	—	—		
n	●	●	●	●	●	●	●	●	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p>MELSEC</p> </div> <p>BKCMP>= s1 s2 d n</p>	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p>BKCMP_GE_M s1, s2, n, d</p> </div>
--	--------------------------	--

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s1	Dato di confronto o operando che contiene il dato di confronto	BIN 16-bit
s2	Dato di confronto o operando che contiene il dato di confronto	BIN 16-bit
d	Primo numero di operando che contiene il risultato dell'operazione di confronto	Bit
n	Numero dei blocchi dati da confrontare	BIN 16-bit

Funzioni Confronto blocchi dati BIN

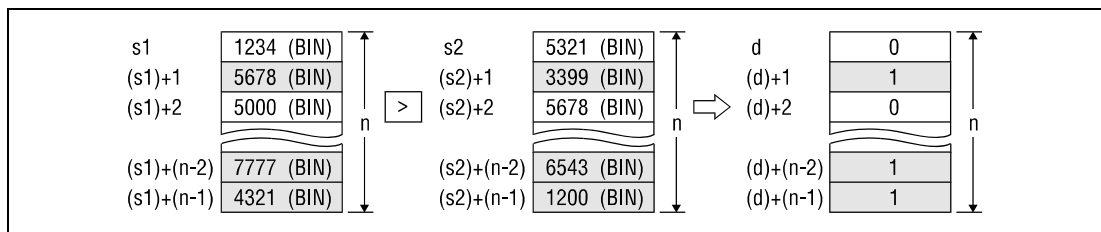
BKCMP Istruzioni per operazioni di confronto

Le istruzioni per operazioni di confronto su blocchi dati BIN sono composte dall'istruzione stessa, dai due operandi s1 e s2 da confrontare, da un operando d in cui memorizzare il risultato, e dal numero di blocchi da confrontare.

Confronta l'ennesimo blocco BIN a 16 bit in s1 con l'ennesimo blocco BIN a 16 bit in s2, iniziando con il primo numero operando. Il risultato del confronto di ciascun blocco è memorizzato in d.

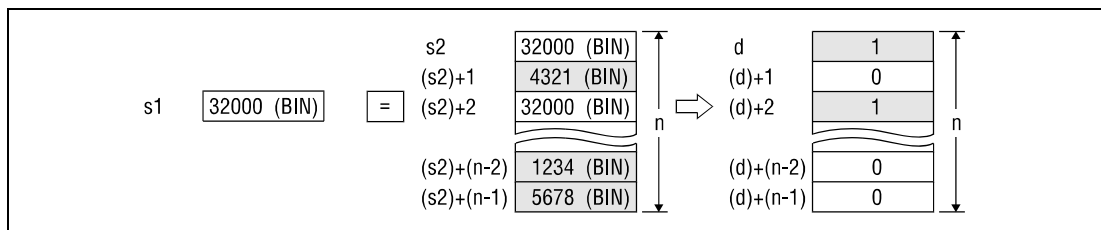
Se il risultato del confronto del blocco è 1, 1 viene memorizzato in d.

Se il risultato del confronto del blocco è 0, 0 viene memorizzato in d.



L'operazione di confronto avviene in unità da 16 bit.

La costante indicata da s1 deve essere un numero BIN a 16 bit compreso fra -32768 e 32767.



I risultati delle operazioni di confronto per le singole istruzioni sono riportati di seguito:

Simbolo istruzione	Risultati operazione di confronto su ennesimo blocco da 16 bit	
	1	0
BKCMP=	s1 = s2	s1 ≠ s2
BKCMP<>	s1 ≠ s2	s1 = s2
BKCMP>	s1 > s2	s1 ≤ s2
BKCMP<=	s1 ≤ s2	s1 > s2
BKCMP<	s1 < s2	s1 ≥ s2
BKCMP>=	s1 ≥ s2	s1 < s2

Se tutti i risultati dei confronti memorizzati in d sono a 1, il segnale di confronto blocco SM704 viene impostato a 1.

Se l'operando indicato da d è già a 1, l'operando non cambia. Per questa ragione, se i dati contenuti in s1 e s2 vengono modificati e viene eseguita l'istruzione BKCMP_P, l'operando indicato da d deve essere precedentemente azzerato.

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il blocco dati BIN in s1, s2 o d supera il campo operandi ammesso (codice di errore: 4101).
- Il campo operandi da [s1 a (s1) + (n-1)] si sovrappone al campo operandi da [d a (d) + (n-1)] (codice di errore: 4101).
- Il campo operandi da [s2 a (s2) + (n-1)] si sovrappone al campo operandi da [d a (d) + (n-1)] (codice di errore: 4101).
- Il campo operandi da [s1 a (s1) + (n-1)] si sovrappone al campo operandi da [s2 a (s2) + (n-1)] (codice di errore: 4101).

Programma di esempio 1

Istruzione di confronto BKCMP=P

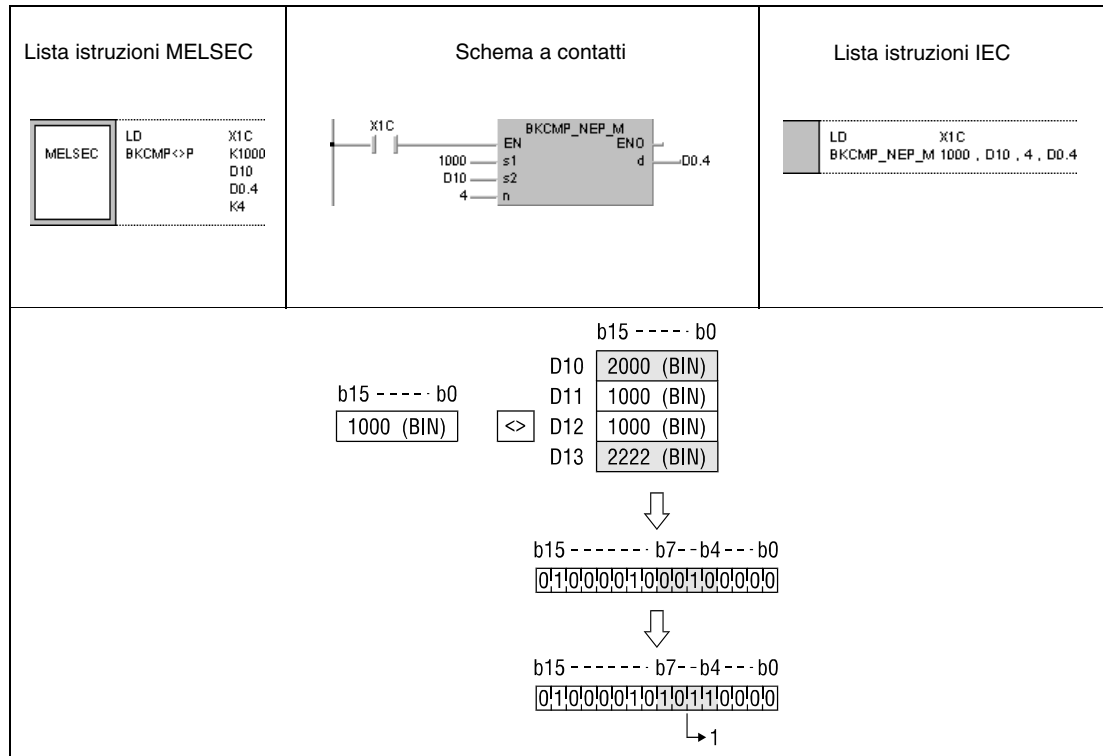
Sul fronte di salita di X20, il programma che segue confronta il blocco dati BIN in D100 con il blocco dati BIN in R0. I risultati del confronto vengono memorizzati a partire da M10. Il numero di blocchi da confrontare (4) è contenuto in D0

<p>Lista istruzioni MELSEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; text-align: center;">MELSEC</td> <td style="width: 15%; text-align: center;">LD</td> <td style="width: 15%; text-align: center;">X20</td> <td style="width: 15%; text-align: center;">D100</td> <td style="width: 15%; text-align: center;">R0</td> <td style="width: 15%; text-align: center;">M10</td> <td style="width: 15%; text-align: center;">D0</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </table>	MELSEC	LD	X20	D100	R0	M10	D0								<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; text-align: center;">LD</td> <td style="width: 15%; text-align: center;">X20</td> <td style="width: 15%; text-align: center;">D100</td> <td style="width: 15%; text-align: center;">R0</td> <td style="width: 15%; text-align: center;">D0</td> <td style="width: 15%; text-align: center;">M10</td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </table>	LD	X20	D100	R0	D0	M10																																								
MELSEC	LD	X20	D100	R0	M10	D0																																																								
LD	X20	D100	R0	D0	M10																																																									
<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 25%;"></td> <td style="width: 25%; text-align: center;">b15 ----- b0</td> <td style="width: 25%; text-align: center;">b15 ----- b0</td> <td style="width: 25%;"></td> </tr> <tr> <td style="text-align: center;">D100</td> <td style="border: 1px solid black; text-align: center;">1000 (BIN)</td> <td style="text-align: center;">R0</td> <td style="border: 1px solid black; text-align: center;">1000 (BIN)</td> </tr> <tr> <td style="text-align: center;">D101</td> <td style="border: 1px solid black; text-align: center;">2000 (BIN)</td> <td style="text-align: center;">R1</td> <td style="border: 1px solid black; text-align: center;">2000 (BIN)</td> </tr> <tr> <td style="text-align: center;">D102</td> <td style="border: 1px solid black; text-align: center;">3000 (BIN)</td> <td style="text-align: center;">R2</td> <td style="border: 1px solid black; text-align: center;">5000 (BIN)</td> </tr> <tr> <td style="text-align: center;">D103</td> <td style="border: 1px solid black; text-align: center;">4000 (BIN)</td> <td style="text-align: center;">R3</td> <td style="border: 1px solid black; text-align: center;">4000 (BIN)</td> </tr> <tr> <td style="text-align: center;">D0</td> <td style="border: 1px solid black; text-align: center;">4</td> <td style="text-align: center;">=</td> <td style="text-align: center;">⇒</td> </tr> <tr> <td></td> <td></td> <td></td> <td style="text-align: center;">M10</td> </tr> <tr> <td></td> <td></td> <td></td> <td style="border: 1px solid black; text-align: center;">1</td> </tr> <tr> <td></td> <td></td> <td></td> <td style="text-align: center;">M11</td> </tr> <tr> <td></td> <td></td> <td></td> <td style="border: 1px solid black; text-align: center;">1</td> </tr> <tr> <td></td> <td></td> <td></td> <td style="text-align: center;">M12</td> </tr> <tr> <td></td> <td></td> <td></td> <td style="border: 1px solid black; text-align: center;">0</td> </tr> <tr> <td></td> <td></td> <td></td> <td style="text-align: center;">M13</td> </tr> <tr> <td></td> <td></td> <td></td> <td style="border: 1px solid black; text-align: center;">1</td> </tr> </table>								b15 ----- b0	b15 ----- b0		D100	1000 (BIN)	R0	1000 (BIN)	D101	2000 (BIN)	R1	2000 (BIN)	D102	3000 (BIN)	R2	5000 (BIN)	D103	4000 (BIN)	R3	4000 (BIN)	D0	4	=	⇒				M10				1				M11				1				M12				0				M13				1
	b15 ----- b0	b15 ----- b0																																																												
D100	1000 (BIN)	R0	1000 (BIN)																																																											
D101	2000 (BIN)	R1	2000 (BIN)																																																											
D102	3000 (BIN)	R2	5000 (BIN)																																																											
D103	4000 (BIN)	R3	4000 (BIN)																																																											
D0	4	=	⇒																																																											
			M10																																																											
			1																																																											
			M11																																																											
			1																																																											
			M12																																																											
			0																																																											
			M13																																																											
			1																																																											

Programma di esempio 2

Istruzione di confronto BKCMP<>P

Con il fronte di salita di X1C, il programma che segue confronta la costante K1000 con il blocco dati che inizia da D10. Il numero di blocchi (4) da confrontare è definito dalla costante K4. I risultati del confronto sono memorizzati nei bit da b4 a b7 di D0.

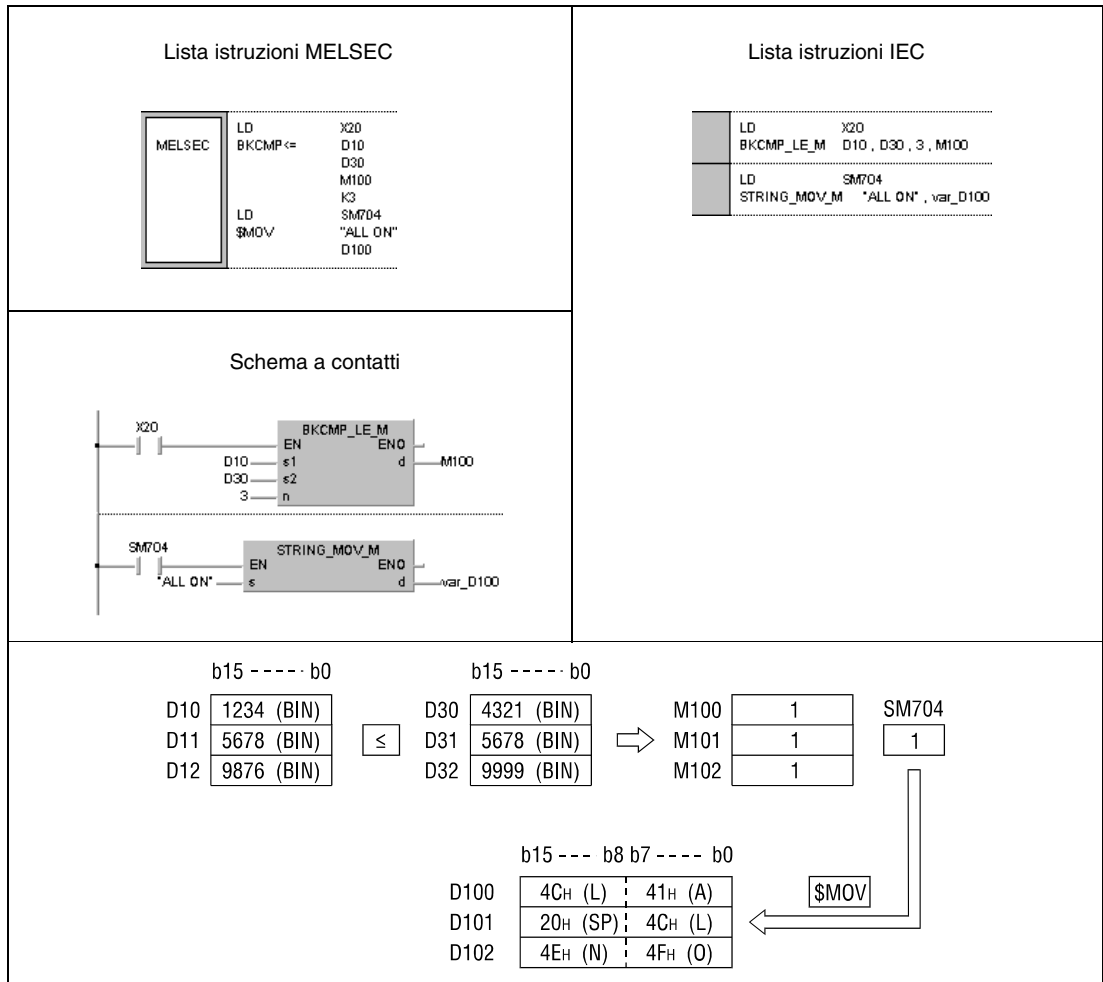


¹ Entrambi questi bit non cambiano di stato dopo l'esecuzione dell'istruzione.

Programma di esempio 3

Istruzioni di confronto BKCMP<=

Per tutto il tempo che X20 è a 1, il programma che segue confronta il blocco dati che inizia da D10 con il blocco dati che inizia da D30. Il numero di blocchi (3) da confrontare è definito dalla costante K3. I risultati del confronto vengono memorizzati a partire da M100. Se tutti i risultati del confronto memorizzati in M100 sono a 1, il segnale di confronto blocchi SM704 viene impostato a 1 e la stringa di caratteri "ALL ON" viene trasferita in D100.



NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.2 Istruzioni per operazioni aritmetiche

Le istruzioni per operazioni aritmetiche effettuano calcoli semplici, quali somme, sottrazioni, moltiplicazioni e divisioni.

Il numero totale delle istruzioni per operazioni aritmetiche è 54 (serie Q e System Q) e 40 (serie A).

Funzione	BIN		BCD	
	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
+ Somma	+	PLUS_M, PLUS_3_M	B+	BPLUS_M, BPLUS_3_M
	+P	PLUSP_M, PLUSP_3_M	B+P	BPLUSP_M, BPLUSP_3_M
	D+	DPLUS_M, DPLUS_3_M	DB+	DBPLUS_M, DBPLUS_3_M
	D+P	DPLUSP_M, DPLUSP_3_M	DB+P	DBPLUSP_M, DBPLUSP_3_M
- Sottrazione	-	MINUS_M, MINUS_3_M	B-	BMINUS_M, BMINUS_3_M
	-P	MINUSP_M, MINUSP_3_M	B-P	BMINUSP_M, BMINUSP_3_M
	D-	DMINUS_M, DMINUS_3_M	DB-	DBMINUS_M, DBMINUS_3_M
	D-P	DMINUSP_M, DMINUSP_3_M	DB-P	DBMINUSP_M, DBMINUSP_3_M
× Moltiplicazione	×	MULTI_3_M	B×	BMULTI_M
	×P	MULTIP_3_M	B×P	BMULTIP_M
	D×	DMULTI_3_M	DB×	DBMULTI_M
	D×P	DMULTIP_3_M	DB×P	DBMULTIP_M
/ Divisione	/	DIVID_3_M	B/	BDIVID_M
	/P	DIVIDP_3_M	B/P	BDIVIDP_M
	D/	DDIVID_3_M	DB/	DBDIVID_M
	D/P	DDIVIDP_3_M	DB/P	DBDIVIDP_M
+1 Incremento	INC	INC_M		
	INCP	INCP_M		
	DINC	DINC_M		
	DINCP	DINCP_M		
-1 Decremento	DEC	DEC_M		
	DECP	DECP_M		
	DDEC	DDEC_M		
	DDECP	DDECP_M		

NOTA Utilizzare comandi IEC con gli editor IEC

Funzione	Dati virgola mobile		Blocco dati BIN	
	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
+ Somma	E+	EPLUS_M, EPLUS_3_M	BK+	BKPLUS_M
	E+P	EPLUSP_M, EPLUSP_3_M	BK+P	BKPLUSP_M
- Sottrazione	E-	EMINUS_M, EMINUS_3_M	BK-	BKMINUS_M
	E-P	EMINUSP_M, EMINUSP_3_M	BK-P	BKMINUSP_M
× Moltiplicazione	E×	EMUL_M		
	E×P	EMULP_M		
/ Divisione	E/	EDIV_M		
	E/P	EDIVP_M		

Funzione	Dati a stringa di caratteri	
	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
+ Somma	\$+	STRING_PLUS_M, STRING_PLUS_3_M
	\$+P	STRING_PLUSP_M, STRING_PLUSP_3_M

NOTA *Utilizzare comandi IEC con gli editor IEC*

Le istruzioni per operazioni aritmetiche su dati in virgola mobile, blocchi dati BIN e stringhe di caratteri sono disponibili solo sulla serie Q e System Q.

Istruzioni per operazioni aritmetiche su dati BIN

Se il risultato della somma supera il valore BIN 32767 (2147483647 per una istruzione a 32 bit), viene generato un valore negativo (overflow).

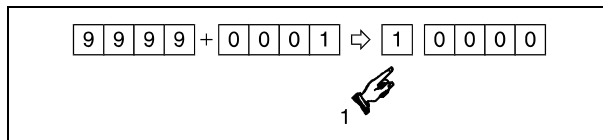
Se il risultato della sottrazione scende sotto il valore BIN -32768 (-2147483647 per una istruzione a 32 bit), viene generato un valore positivo (underflow).

Il calcolo di valori positivi e negativi è indicato di seguito:

$5 + 8 = 13$
 $5 - 8 = -3$
 $5 \times 3 = 15$
 $-5 \times 3 = -15$
 $-5 \times (-3) = 15$
 $5 / 3 = 1$ con resto 2
 $-5 / 3 = -1$ con resto -2
 $5 / (-3) = -1$ con resto 2
 $-5 / (-3) = 1$ con resto -2

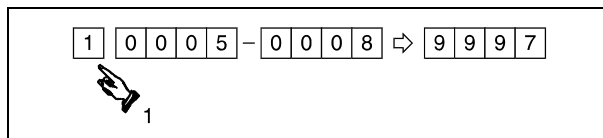
Istruzioni per operazioni aritmetiche su dati BCD

Se il risultato della somma supera 9999 (99999999 per una istruzione a 32 bit), i bit più significativi vengono ignorati (overflow). In questo caso la segnalazione di riporto non viene attivata.



¹ Riporto ignorato

Se il risultato della sottrazione è inferiore a 0000 (underflow), il riporto viene considerato come segue:



² Riporto

6.2.1 +, +P, -, -P

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

	Operandi															Definizione cifra	Numero di passi	Indice	Rapporto	Errore	
	Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello						
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z						V
s	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●				
d		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●					
s1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●				
s2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●				
d1		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●					

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale di programmazione per il corrispondente numero di passi.

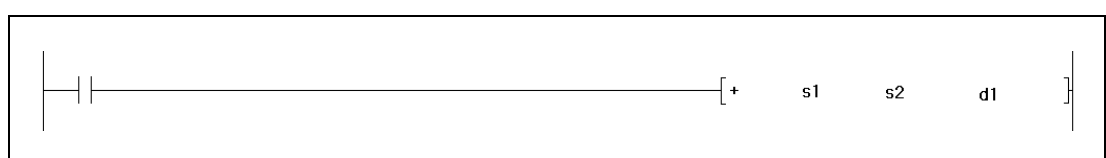
Operandi MELSEC Q

	Operandi								Segnale di errore	Numero di passi	
	Operando interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	●	●	—	—	3
d	●	●	●	●	●	●	●	—	—	—	
s1	●	●	●	●	●	●	●	●	—	—	4
s2	●	●	●	●	●	●	●	●	—	—	
d1	●	●	●	●	●	●	●	—	—	—	

GX IEC Developer

<p>Lista istruzioni MELSEC</p>	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p>
--------------------------------	--------------------------	-----------------------------

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Addendo o sottraendo, oppure primo numero di operando che contiene addendo o sottraendo	BIN 16-bit
d	Addendo o minuendo, oppure primo numero di operando in cui viene memorizzata la somma o differenza	
s1	Addendo o minuendo, oppure primo numero di operando contenente addendo o minuendo	
s2	Addendo o sottraendo, oppure primo numero di operando che contiene addendo o sottraendo	
d1	Primo numero dell'operando in cui viene memorizzato il risultato	

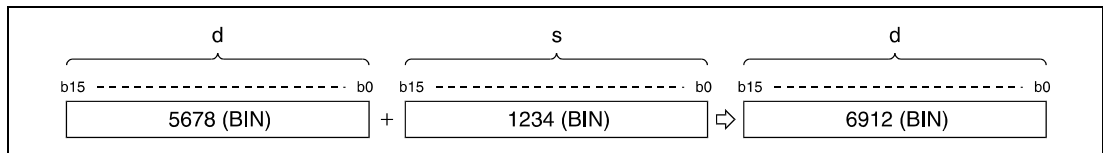
Funzioni

Operazioni di somma e sottrazione dati BIN a 16 bit

+ Somma BIN (16-bit)

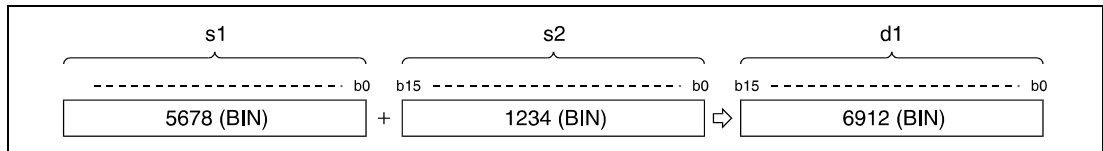
● Variante 1:

Il dato BIN a 16 bit in d viene sommato al dato BIN a 16 bit in s. Il risultato della somma viene memorizzato in d.



● Variante 2:

Il dato BIN a 16 bit in s1 viene sommato al dato BIN a 16 bit in s2. Il risultato della somma è memorizzato in d1.



I dati BIN a 16 bit indicati da s, d, s1, s2 e d1 devono essere nel campo da -32768 a 32767.

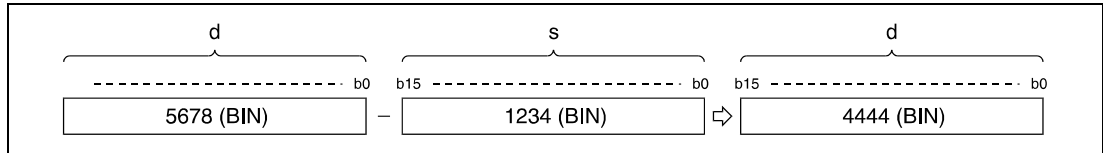
Il bit più significativo (b15) di ciascun operando determina se il dato in s, d, s1 o d1 è positivo (bit = 0) o negativo (bit = 1).

Se vengono scavalcati il bit meno significativo (b0) o quello più significativo (b15), il riporto non viene segnalato.

- Sottrazione BIN (16-bit)

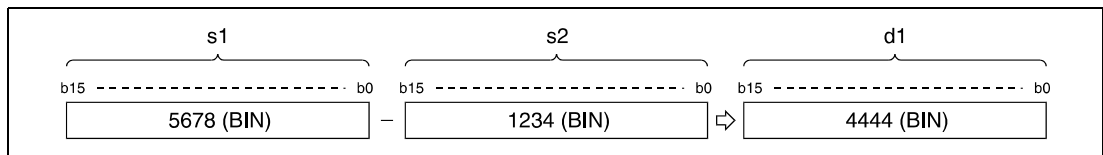
● Variante 1:

Il dato BIN a 16 bit in s viene sottratto dal dato BIN a 16 bit in d. Il risultato della sottrazione viene memorizzato in d.



● Variante 2:

Il dato BIN a 16 bit in s2 viene sottratto dal dato BIN a 16 bit in s1. Il risultato è memorizzato in d1.



I dati BIN a 16 bit indicati da s, d, s1, s2 e d1 devono essere nel campo da -32768 a 32767.

Il bit più significativo (b15) di ciascun operando determina se il dato in s, d, s1 o d1 è positivo (bit = 0) o negativo (bit = 1).

Se vengono scavalcati il bit meno significativo (b0) o quello più significativo (b15), il riporto non viene segnalato.

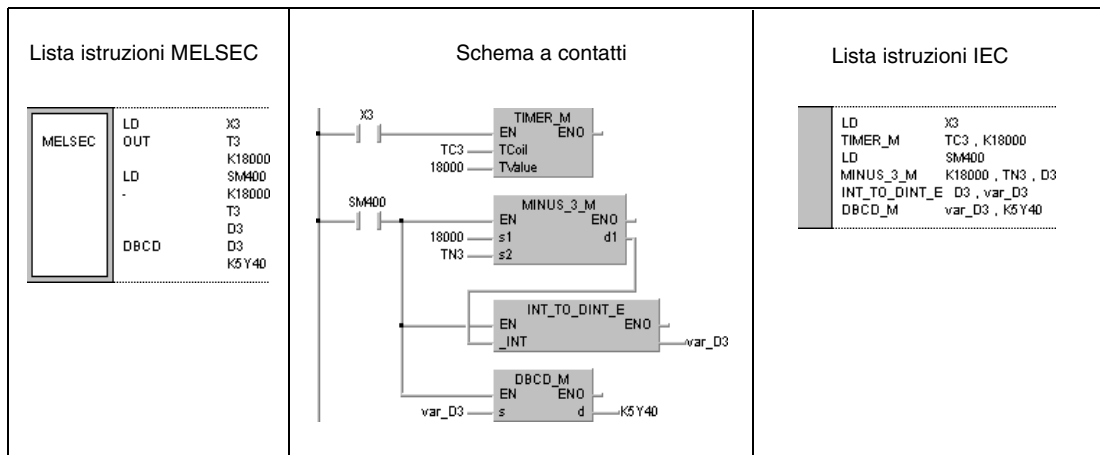
Programma di esempio 1

+P

Il programma che segue somma il dato in D3 con quello in D0 sul fronte di salita di X5. Il risultato viene memorizzato da Y38 a Y3F.

Lista istruzioni MELSEC	Schema a	Lista istruzioni IEC
<pre> MELSEC LD X5 +P D3 D0 K2Y38 </pre>		<pre> LD X5 PLUSP_3_M D3, D0, K2Y38 </pre>

Programma di esempio 2 - Il programma seguente emette la differenza fra il valore nominale e quello attuale del timer T3 in codice BCD sulle uscite da Y40 a Y53.

**NOTA**

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.2.2 D+, D+P, D-, D-P

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

	Operandi																Definizione cifra	Numero di passi	Indice	Rapporto	Errore						
	Operandi a bit								Operandi a word (16-bit)													Costanti		Puntatore		Livello	
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V						K	H (16#)	P	I	N	M9012
s	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●							9			
d		●	●	●	●	●	●	●	●	●	●	●	●	●	●									11	●		
s1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●								●		
s2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●										
d1		●	●	●	●	●	●	●	●	●	●	●	●	●	●												

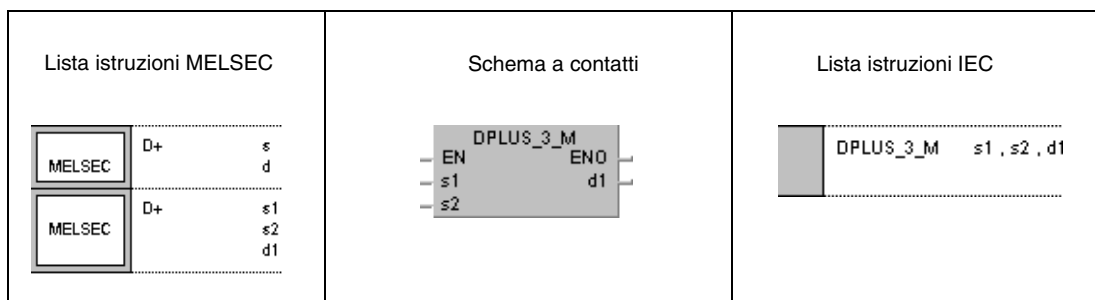
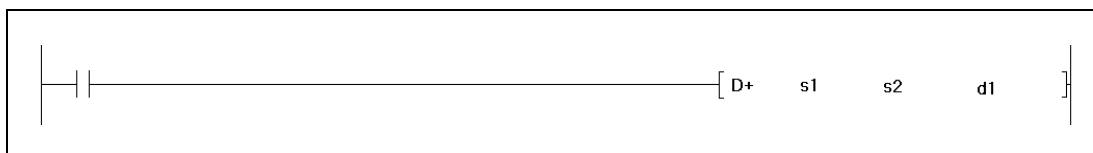
¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale di programmazione per il corrispondente numero di passi.

Operandi MELSEC Q

	Operandi								Segnale di errore	Numero di passi	
	Operando interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	●	●	—	—	3 ¹⁾
d	●	●	●	●	●	●	●	—	—	—	
s1	●	●	●	●	●	●	●	●	—	—	4 ²⁾
s2	●	●	●	●	●	●	●	●	—	—	
d1	●	●	●	●	●	●	●	—	—	—	

¹ Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.
 Se viene usata una CPU QnA oppure una CPU a processore singolo del System Q: 3
 Se viene usata una CPU multiprocessore del System Q con operandi interni a word (tranne file registri ZR): 5
 costanti: 5
 Operandi a bit con numero operando multiplo di 16, in cui la cifra indicata è K8, e che non usa indicizzazione: 5
 Se viene usata una CPU multiprocessore del System Q con operandi diversi da quelli menzionati: 3

² Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.
 Se viene usata una CPU QnA oppure una CPU a processore singolo del System Q: 4
 Se viene usata una CPU multiprocessore del System Q con operandi interni a word (tranne file registri ZR): 6
 costanti: 6
 Operandi a bit con numero operando multiplo di 16, in cui la cifra indicata è K8, e che non usa indicizzazione: 6
 Se viene usata una CPU multiprocessore del System Q con operandi diversi da quelli menzionati: 4

GX IEC Developer**GX Developer****Variabili**

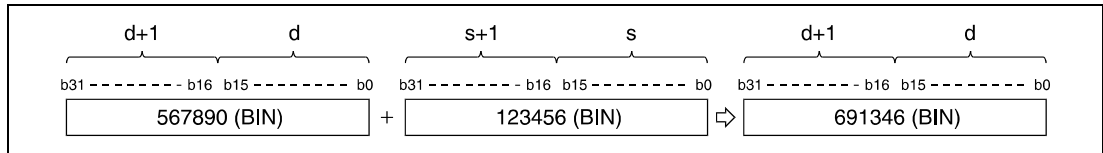
Dati impostati	Significato	Tipo dati
s	Addendo o sottraendo, oppure primo numero di operando che contiene addendo o sottraendo	BIN 32-bit
d	Addendo o minuendo, oppure primo numero di operando contenente addendo o minuendo	
s1	Addendo o minuendo, oppure primo numero di operando contenente addendo o minuendo	
s2	Addendo o sottraendo, oppure primo numero di operando che contiene addendo o sottraendo	
d1	Primo numero dell'operando in viene memorizzato il risultato	

Funzioni Operazioni di somma e sottrazione dati BIN a 32 bit

D+ Somma BIN (32-bit)

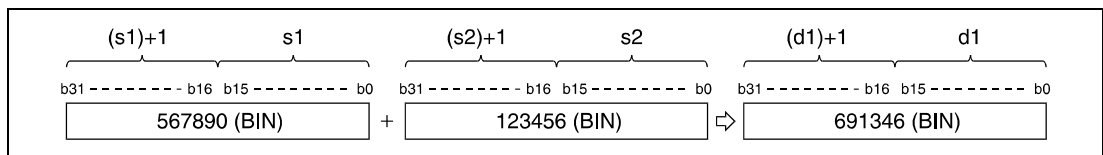
● Variante 1:

Il dato BIN a 32 bit in d viene sommato al dato BIN a 32 bit in s. Il risultato della somma viene memorizzato in d.



● Variante 2:

Il dato BIN a 32 bit in s1 viene sommato al dato BIN a 32 bit in s2. Il risultato della somma è memorizzato in d1.



I dati BIN a 32 bit indicati da s, d, s1, s2 e d1 devono essere nel campo da -2147483648 a 2147483647.

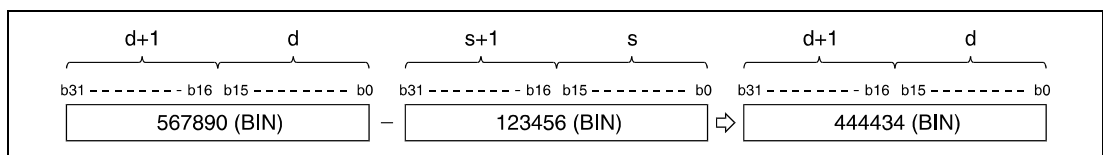
Il bit più significativo (b31) di ciascun operando determina se il dato in s, d, s1 o d1 è positivo (bit = 0) o negativo (bit = 1).

Se vengono scavalcati il bit meno significativo (b0) o quello più significativo (b31), il riporto non viene segnalato.

D- Somma BIN (32-bit)

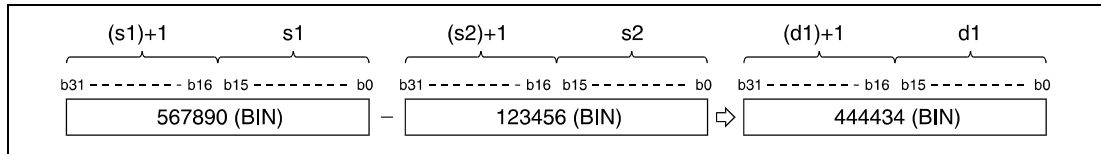
● Variante 1:

Il dato BIN a 32 bit in s viene sottratto dal dato BIN a 32 bit in d. Il risultato della sottrazione viene memorizzato in d.



● Variante 2:

Il dato BIN a 32 bit in s2 viene sottratto dal dato BIN a 32 bit in s1. Il risultato è memorizzato in d1.



I dati BIN a 32 bit indicati da s, d, s1, s2 e d1 devono essere nel campo da -2147483648 a 2147483647.

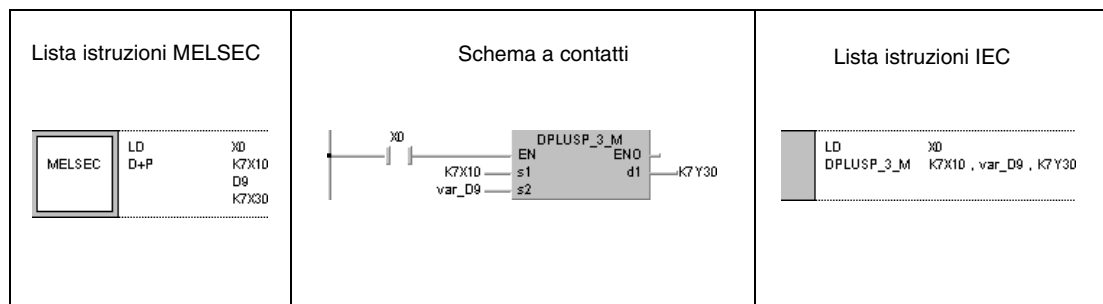
Il bit più significativo (b31) di ciascun operando determina se il dato in s, d, s1 o d1 è positivo (bit = 0) o negativo (bit = 1).

Se vengono scavalcati il bit meno significativo (b0) o quello più significativo (b31), il riporto non viene segnalato.

Programma di esempio 1

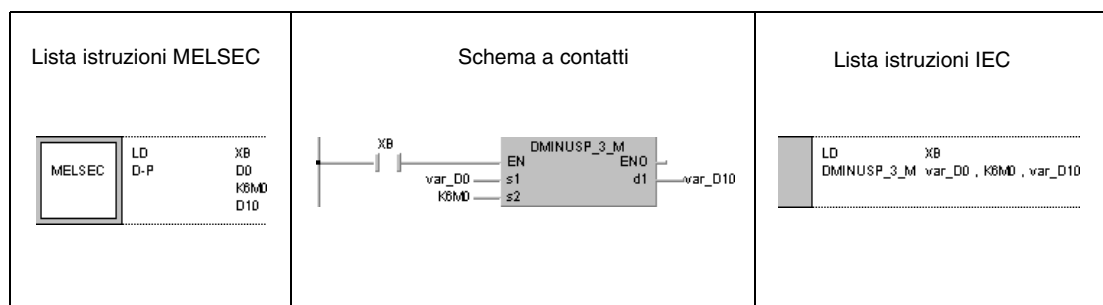
D+P

Sul fronte di salita di X0, il programma seguente somma i dati contenuti da X10 a X2B con il contenuto di D9 e D10. Il risultato viene memorizzato da Y30 a Y4B.

**Programma di esempio 2**

D-P

Sul fronte di salita di XB, il programma seguente sottrae il dato contenuto da M0 a M23 dal dato in D0 e D1. Il risultato viene memorizzato in D10 e D11.

**NOTA**

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.2.3 x, xP, /, /P

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

	Operandi															Definizione cifra	Numero di passi	Indice	Ripporto	Errore						
	Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello											
	X	Y	M	L	S	B	F	T	C	D	W	R	AO	A1	Z						V	K	H (16#)	P	I	N
s1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●					K1	7	●		
s2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●					K4	1			●
d1		●	●	●	●	●	●	●	●	●	●	●	●	●	●											

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale di programmazione per il corrispondente numero di passi.

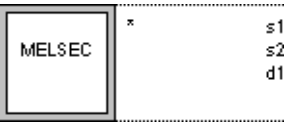
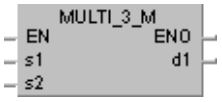
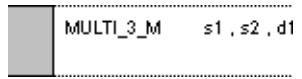
Operandi MELSEC Q

	Operandi								Segnale di errore	Numero di passi	
	Operando interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale Registro indice U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	●	●	●	●	●	●	●	●	—	SM0	4 ¹⁾
s2	●	●	●	●	●	●	●	●	—		
d1	●	●	●	●	●	●	—	—	—		

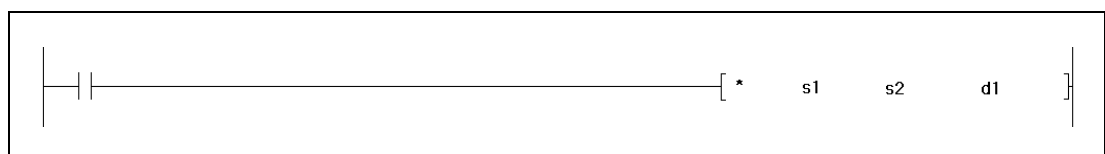
¹ Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.

- Se viene usata una CPU QnA: 4
- Se viene usata una CPU del System Q con operandi interni a word (tranne file registri ZR): 3
- costanti: 3
- Operandi a bit con numero operando multiplo di 16, in cui la cifra indicata è K8, e che non usa indicizzazione: 3
- Se viene usata una CPU del System Q con operandi diversi da quelli menzionati: 4

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
--	--	---

GX Developer



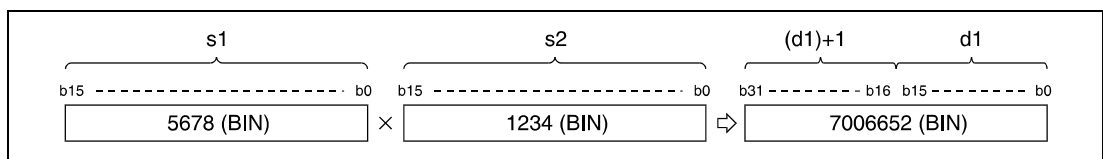
Variabili

Dati impostati	Significato	Tipo dati
s1	Moltiplicatore o dividendo, oppure primo numero dell'operando che contiene moltiplicatore o dividendo	BIN 16-bit
s2	Moltiplicatore o divisore, oppure primo numero dell'operando che contiene moltiplicatore o divisore	BIN 16-bit
d1	Primo numero dell'operando in cui viene memorizzato il risultato dell'operazione di moltiplicazione o divisione	BIN 32-bit

Funzioni

Moltiplicazione e divisione dati BIN a 16 bit**x Moltiplicazione BIN (16-bit)**

Il dato BIN a 16 bit in s1 viene moltiplicato con il dato BIN a 16 bit in s2. Il risultato è memorizzato in d1.



Se d1 indica un operando a bit, il risultato viene memorizzato a partire dai bit meno significativi.

Esempio:

K1: 4 bit meno significativi (da b0 a b3)

K4: 16 bit meno significativi (da b0 a b15)

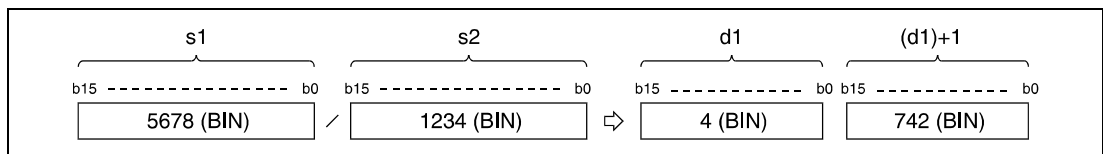
K8: 32 bit (da b0 a b31)

I dati BIN a 16 bit indicati da s1 e s2 devono essere nel campo da -32768 a 32767.

Il bit più significativo (b15 o b31 per d1) di ciascun operando determina se il dato in s1, s2 o d1 è positivo (bit = 0) o negativo (bit = 1).

/ Divisione BIN (16-bit)

Il dato BIN a 16 bit in s1 viene diviso per il dato BIN a 16 bit in s2. Il risultato è memorizzato in d1.



Se viene usato un operando a word, il risultato dell'operazione viene memorizzato su 32 bit, suddiviso fra quoziente e resto. Il quoziente è contenuto nei 16 bit meno significativi. Il resto è contenuto nei 16 bit più significativi.

Se viene usato un operando a bit, vengono utilizzati solo i 16 bit relativi al quoziente.

I dati BIN a 16 bit indicati da s1 e s2 devono essere nel campo da -32768 a 32767.

Il bit più significativo (b15 o b31 per d1) di ciascun operando determina se il dato in s1, s2, d1 o (d1)+1 è positivo (bit = 0) o negativo (bit = 1).

Errori di esecuzione

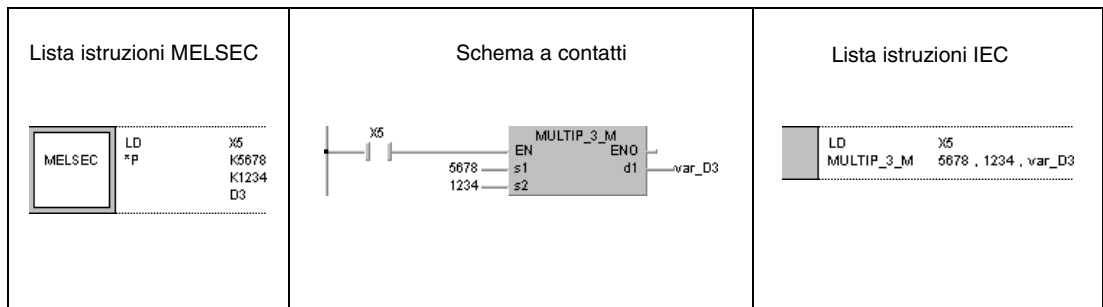
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- L'operando d1 indica A1 o V (serie A).
- Divisione per zero (serie Q e System Q = codice di errore 4100).

Programma di esempio 1

xP

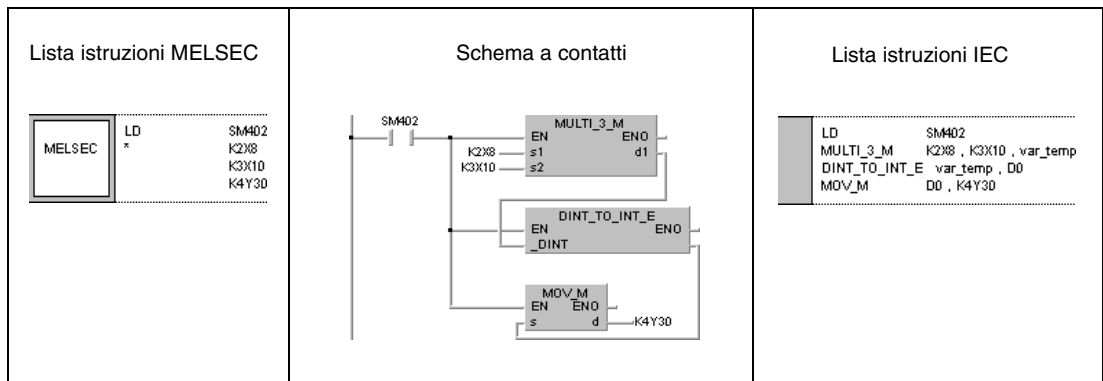
Con il fronte di salita di X5, il programma seguente moltiplica 5678 e 1234. Il risultato viene memorizzato in D3 e D4.



Programma di esempio 2

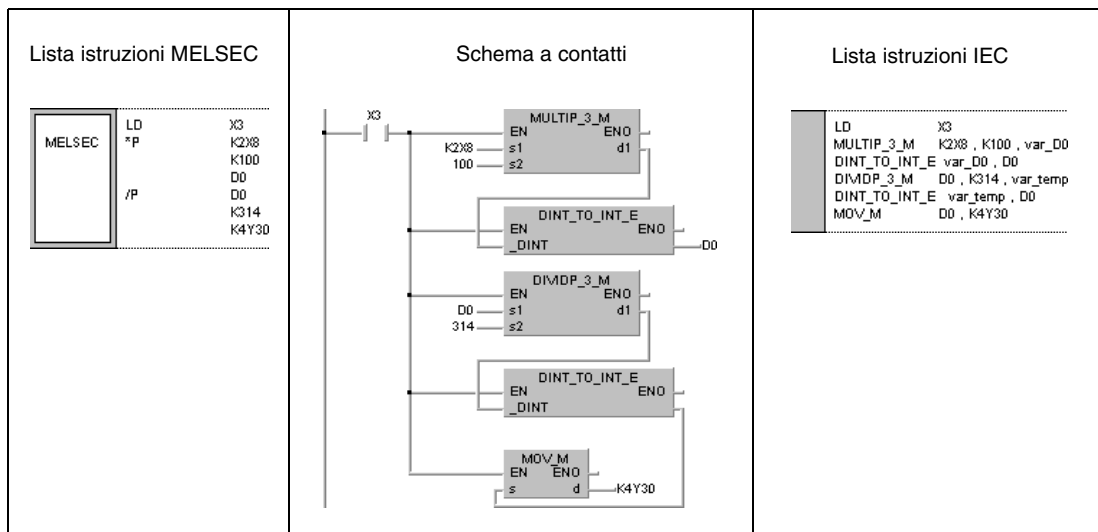
x

Il programma che segue moltiplica il dato BIN da X8 a XF ed il dato BIN da X10 a X1B. Il risultato viene inviato sulle uscite da Y30 a Y3F.



Programma di esempio 3 /P

Con il fronte di salita di X3, il programma che segue divide il dato contenuto da X8 a XF per 3,14. Il risultato viene inviato sulle uscite da Y30 a Y3F.

**NOTA**

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.2.4 Dx, DxP, D/, D/P

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

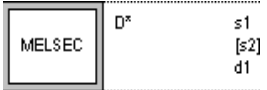
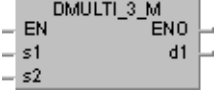
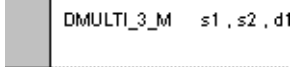
	Operandi															Definizione cifra	Numero di passi	Indice	Riporto	Errore	
	Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello						
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z						V
s1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●			
s2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●			
d1		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●			

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale di programmazione per il corrispondente numero di passi.

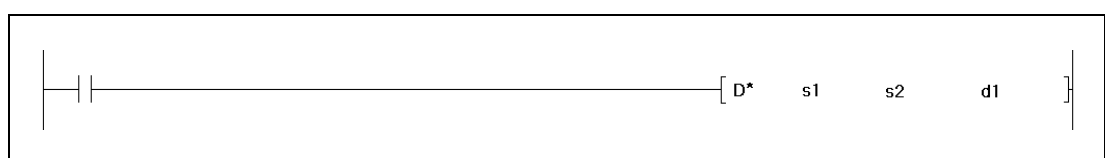
Operandi MELSEC Q

	Operandi								Segnale di errore	Numero di passi	
	Operando interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale Modulo funzione speciale U□NG□	Registro indice Registro indice Zn	Costante Costante K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	●	●	●	●	●	●	●	●	—	SM0	4
s2	●	●	●	●	●	●	●	●	—		
d1	●	●	●	—	—	—	—	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
--	--	---

GX Developer

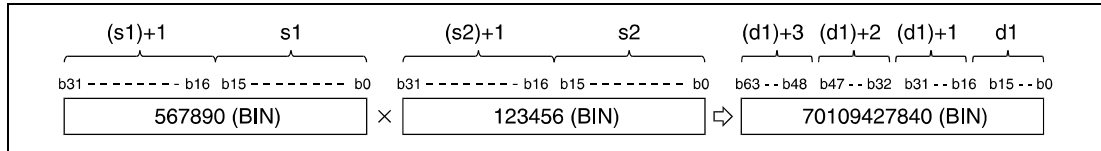


Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s1	Moltiplicatore o dividendo, oppure primo numero dell'operando che contiene moltiplicatore o dividendo	BIN 32-bit	ANY32
s2	Moltiplicatore o divisore, oppure primo numero dell'operando che contiene moltiplicatore o divisore	BIN 32-bit	ANY32
d1	Primo numero dell'operando in cui viene memorizzato il risultato dell'operazione di moltiplicazione o divisione	BIN 64-bit	Array [1..2] di ANY32

Funzioni **Moltiplicazione e divisione dati BIN a 32 bit****Dx** **Moltiplicazione BIN (32-bit)**

Il dato BIN a 32 bit in s1 viene moltiplicato con il dato BIN a 32 bit in s2. Il risultato è memorizzato in d1.



Se d1 indica un operando a bit, il risultato viene memorizzato a partire dai bit meno significativi.

Esempio:

K1: 4 bit meno significativi (da b0 a b3)

K4: 16 bit meno significativi (da b0 a b15)

K8: 32 bit (da b0 a b31)

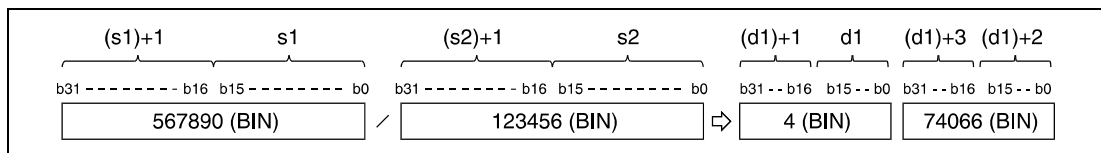
Se i 32 bit più significativi dell'operando a bit sono utilizzati per il risultato della moltiplicazione, memorizzare temporaneamente il dato in un operando a word, e trasferire successivamente il dato dell'operando a word nell'operando a bit indicato da (d1)+2 e (d1)+3.

I dati BIN a 32 bit indicati da s1 e s2 devono essere nel campo da -2147483648 a 2147483647.

Il bit più significativo (b31 o b63 per d1) di ciascun operando determina se il dato in s1, s2 o d1 è positivo (bit = 0) o negativo (bit = 1).

D/ **Divisione BIN (32-bit)**

Il dato BIN a 32 bit in s1 viene diviso per il dato BIN a 32 bit in s2. Il risultato è memorizzato in d1.



Se viene usato un operando a word, il risultato dell'operazione viene memorizzato come array di DINT (64 bit), suddiviso fra quoziente e resto. Il quoziente è contenuto negli elementi meno significativi dell'array (32 bit). Il resto è contenuto negli elementi più significativi dell'array (32 bit).

Se viene usato un operando a bit, vengono utilizzati solo i 32 bit relativi al quoziente.

I dati BIN a 32 bit indicati da s1 e s2 devono essere nel campo da -2147483648 a 2147483647.

Il bit più significativo (b31) di ciascun operando determina se il dato in s1, s2, d1 o (d1)+2 è positivo (bit = 0) o negativo (bit = 1).

Errori di esecuzione

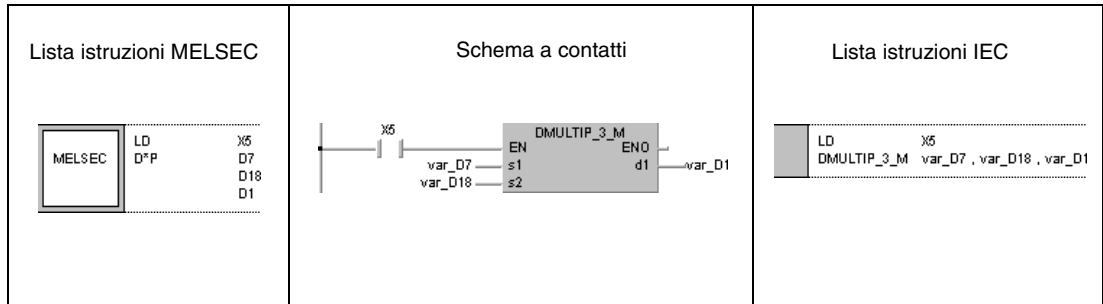
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- A1 o V sono stati indicati da s1 o s2, oppure A0, A1, Z e V sono stati indicati da d1 (serie A).
- Divisione per zero (serie Q e System Q = codice di errore 4100).

Programma di esempio 1

DxP

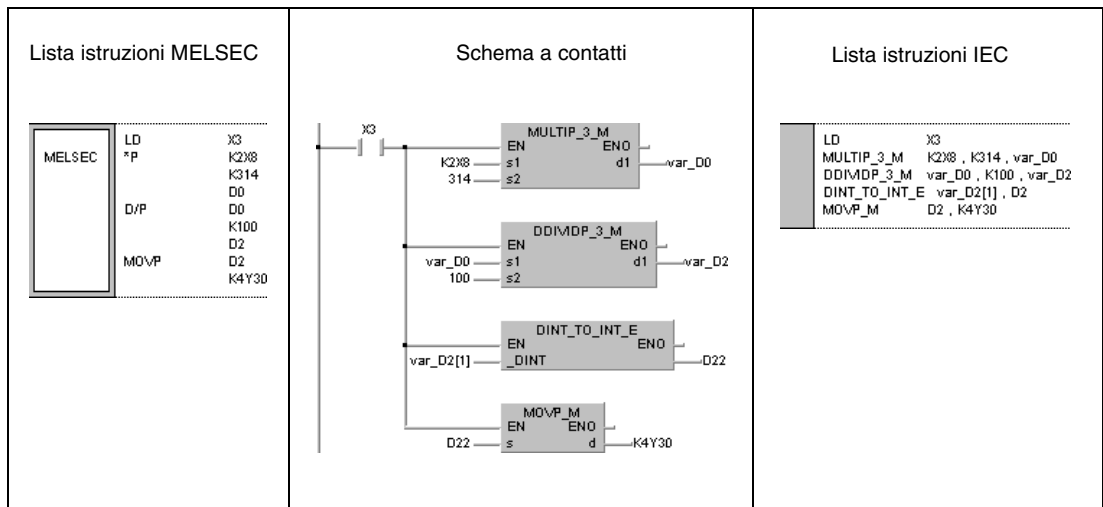
Con il fronte positivo di X5, il programma seguente moltiplica il dato BIN contenuto in D7 e D8 con il dato BIN in D18 e D19. Il risultato viene memorizzato da D1 a D4.



Programma di esempio 2

xP

Con il fronte di salita di X3, il programma che segue moltiplica il dato contenuto da X8 a XF per 3,14. Il risultato viene inviato sulle uscite da Y30 a Y3F.



NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.2.5 B+, B+P, B-, B-P

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

	Operandi utilizzabili															Definizione cifre	Numero di passi	Indice	Riporto	Errore	
	Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello						
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1							Z
s	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●				
d		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●					
s1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●				
s2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●				
d1		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●					

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale di programmazione per il corrispondente numero di passi.

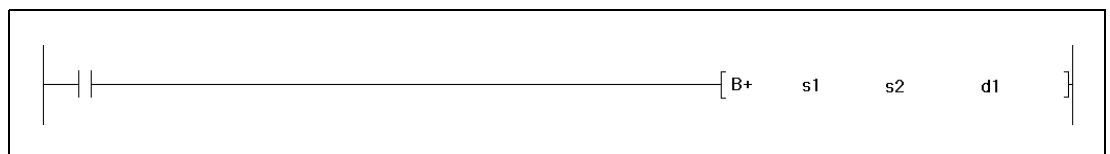
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	●	●	—	SM0	3
d	●	●	●	●	●	●	●	—	—		
s1	●	●	●	●	●	●	●	●	—	SM0	4
s2	●	●	●	●	●	●	●	●	—		
d1	●	●	●	●	●	●	●	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p>	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p>
--------------------------------	--------------------------	-----------------------------

GX Developer



Variabili

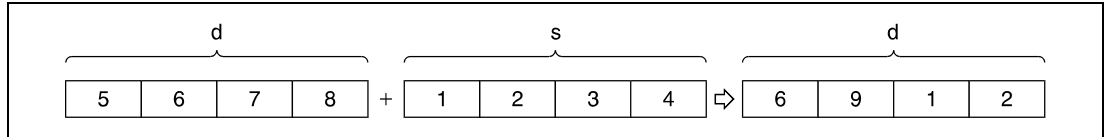
Dati impostati	Significato	Tipo dati
s	Addendo o sottraendo, oppure primo numero di operando che contiene addendo o sottraendo	BCD 4-cifre
d	Addendo o minuendo, oppure primo numero di operando contenente addendo o minuendo	
s1	Addendo o minuendo, oppure primo numero di operando contenente addendo o minuendo	
s2	Addendo o sottraendo, oppure primo numero di operando che contiene addendo o sottraendo	
d1	Primo numero dell'operando in viene memorizzato il risultato	

Funzioni Operazioni di somma e sottrazione dati BCD a 4 cifre

B+ Somma BCD (4-cifre)

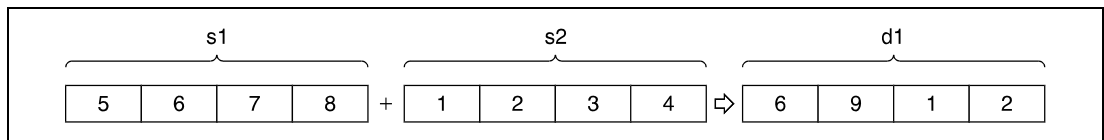
● Variante 1:

Il dato BCD a 4 cifre in d viene sommato al dato BCD a 4 cifre in s. Il risultato della somma viene memorizzato in d.



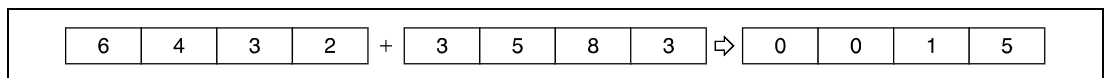
● Variante 2:

Il dato BCD a 4 cifre in s1 viene sommato al dato BCD a 4 cifre in s2. Il risultato è memorizzato in d1.



I dati BCD a 4 cifre indicati da s, d, s1, s2, e d1 sono nel campo da 0 a 9999. Le cifre non indicate sono considerate come 0 (ad es. 12 = 0012).

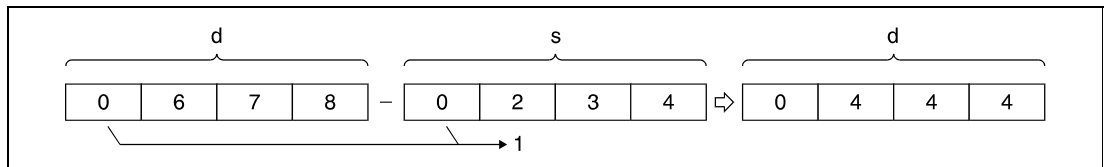
Se il risultato della somma supera 9999, i bit più pesanti sono ignorati (overflow). In questo caso la segnalazione di riporto non viene attivata.



B- Sottrazione BCD (4-cifre)

● Variante 1:

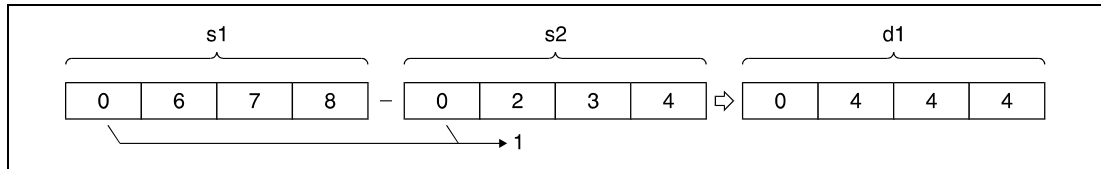
Il dato BCD a 4 cifre in s viene sottratto dal dato BCD a 4 cifre in d. Il risultato della sottrazione viene memorizzato in d.



¹ Le cifre non specificate vengono considerate come 0.

● Variante 2:

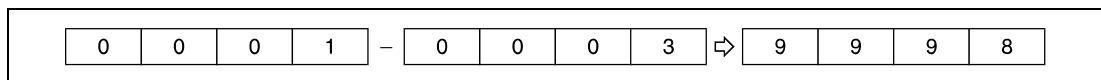
Il dato BCD a 4 cifre in s2 viene sottratto dal dato BCD a 4 cifre in s1. Il risultato è memorizzato in d1.



¹ Le cifre non specificate vengono considerate come 0.

I dati BCD a 4 cifre indicati da s, d, s1, s2 e d1 devono essere nel campo da 0 a 9999.

Se il risultato della sottrazione è negativo, il minuendo viene diminuito del valore contenuto nel sottraendo. In questo caso la segnalazione di riporto non viene attivata.



Nel seguito del programma, è necessario accertarsi che risultati positivi o negativi siano gestiti in modo adeguato.

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il dato BCD a 4 cifre indicato da s, d, s1, s2, o d1 supera il campo ammesso per l'operando (da 0 a 9999) (serie Q e System Q = codice di errore 4100).

B+, B+P, B-, B-P

Programma di esempio 1

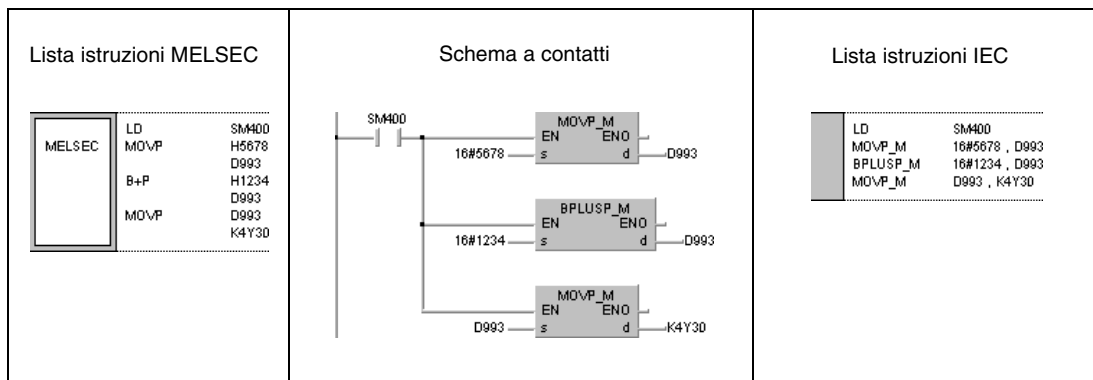
B+P (s, d)

Il programma seguente somma il dato BCD 5678 al dato BCD 1234. Il risultato viene memorizzato in D993 ed inviato sulle uscite da Y30 a Y3F.

La prima riga del programma memorizza il valore 5678 in D993 sul fronte di salita di SM400.

Il passo successivo somma il dato BCD 1234 al dato BCD in D993.

L'istruzione MOV nell'ultimo passo di programma invia il risultato in D993 sulle uscite da Y30 a Y3F.



Programma di esempio 2

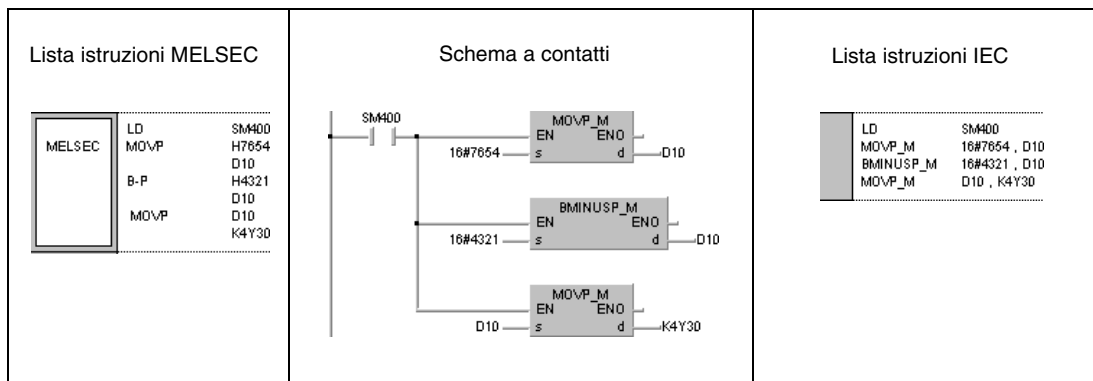
B-P (s, d)

Il programma seguente sottrae il dato BCD 4321 dal dato BCD 7654. Il risultato viene memorizzato in D10 ed inviato sulle uscite da Y30 a Y3F.

La prima riga del programma memorizza il valore 7654 in D10 sul fronte di salita di SM400.

Il passo successivo sottrae il dato BCD 4321 dal dato BCD in D10.

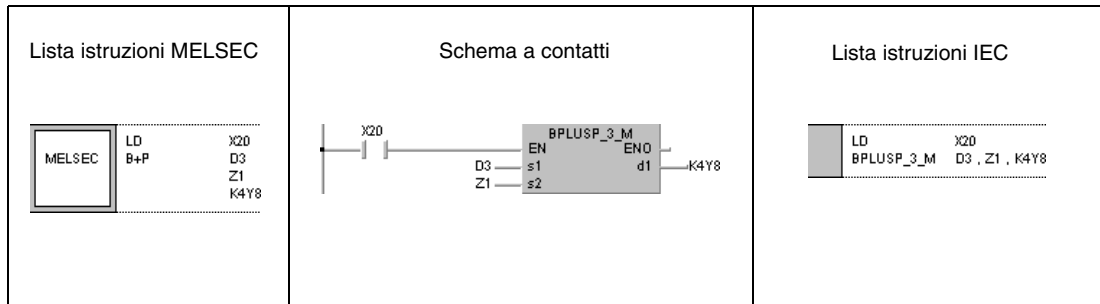
L'istruzione MOV nell'ultimo passo di programma invia il risultato in D10 sulle uscite da Y30 a Y3F.



Programma di esempio 3

B+P (s1, s2, d1)

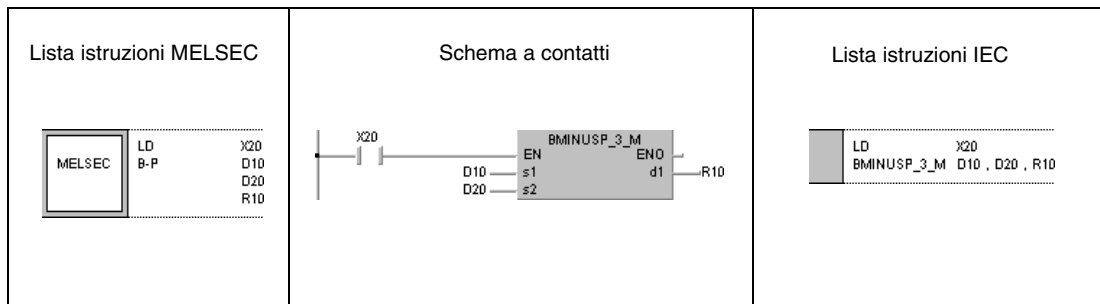
Il programma che segue somma il dato BCD in D3 con quello in Z1 sul fronte di salita di X20. Il risultato viene inviato sulle uscite da Y8 a Y17.



Programma di esempio 4

B-P (s1, s2, d1)

Il programma che segue somma il dato BCD in D20 con quello in D10 sul fronte di salita di X20. Il risultato è memorizzato in R10.



DB+, DB+P, DB-, DB-P

6.2.6 DB+, DB+P, DB-, DB-P

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

	Operandi utilizzabili																	Definizione cifre	Numero di passi	Indice	Riporto	Errore		
	Operandi a bit							Operandi a word (16-bit)							Costanti		Puntatore						Livello	
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K						H (16#)	P
s	●	●	●	●	●	●	●	●	●	●	●	●	●	●	● ²	●	●							
d		●	●	●	●	●	●	●	●	●	●	●	●	●	● ²									
s1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	● ²	●	●							
s2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	● ²	●	●							
d1		●	●	●	●	●	●	●	●	●	●	●	●	●	● ²									

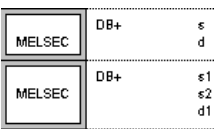
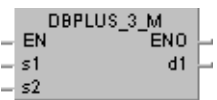
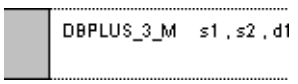
¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale di programmazione per il corrispondente numero di passi.

² Solo CPU AnA, AnAS e AnU.

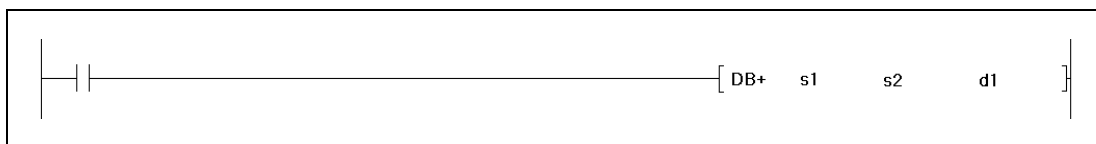
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	●	●	—	SM0	3
d	●	●	●	●	●	●	●	—	—		
s1	●	●	●	●	●	●	●	●	—	SM0	4
s2	●	●	●	●	●	●	●	●	—		
d1	●	●	●	●	●	●	●	—	—		

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
		

GX Developer



Variabili

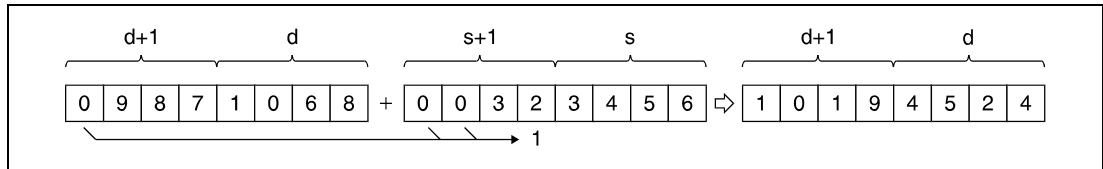
Dati impostati	Significato	Tipo dati
s	Addendo o sottraendo, oppure primo numero di operando che contiene addendo o sottraendo	BCD 8-cifre
d	Addendo o minuendo, oppure primo numero di operando contenente addendo o minuendo	
s1	Addendo o minuendo, oppure primo numero di operando contenente addendo o minuendo	
s2	Addendo o sottraendo, oppure primo numero di operando che contiene addendo o sottraendo	
d1	Primo numero dell'operando in viene memorizzato il risultato	

Funzioni Operazioni di somma e sottrazione dati BCD a 8 cifre

DB+ Somma BCD (8-cifre)

● Variante 1:

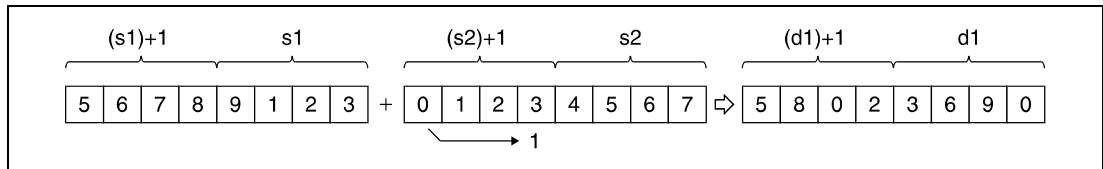
Il dato BCD a 8 cifre in d viene sommato al dato BCD a 8 cifre in s. Il risultato della somma viene memorizzato in d.



¹ Le cifre non specificate vengono considerate come 0.

● Variante 2:

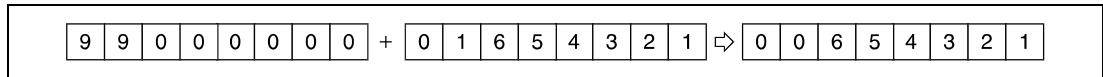
Il dato BCD a 8 cifre in s1 viene sommato al dato BCD a 8 cifre in s2. Il risultato è memorizzato in d1.



¹ Le cifre non specificate vengono considerate come 0.

I dati BCD a 8 cifre indicati da s, d, s1 e d1 sono nel campo da 0 a 99999999. Le cifre non indicate sono considerate come 0 (ad es. 12345 = 0012345).

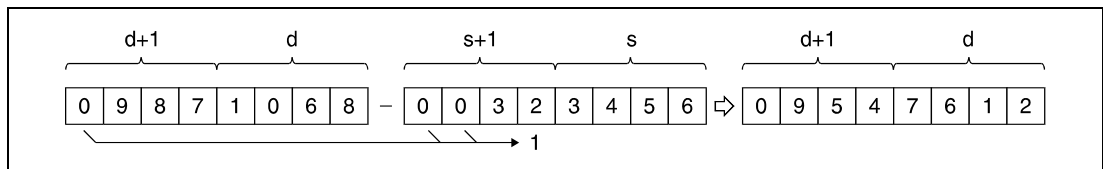
Se il risultato della somma supera 99999999, i bit più pesanti sono ignorati (overflow). In questo caso la segnalazione di riporto non viene attivata.



DB- Sottrazione BCD (8-cifre)

● Variante 1:

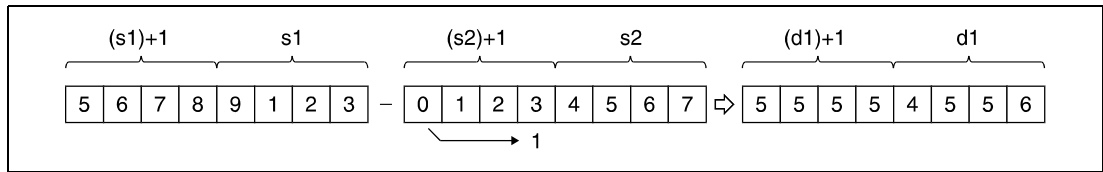
Il dato BCD a 8 cifre in s viene sottratto dal dato BCD a 8 cifre in d. Il risultato della sottrazione viene memorizzato in d.



¹ Le cifre non specificate vengono considerate come 0.

● Variante 2:

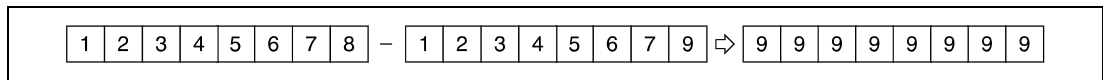
Il dato BCD a 8 cifre in s2 viene sottratto dal dato BCD a 8 cifre in s1. Il risultato è memorizzato in d1.



¹ Le cifre non specificate vengono considerate come 0.

I dati BCD a 8 cifre indicati da s, d, s1 e d1 sono nel campo da 0 a 99999999. Le cifre non indicate sono considerate come 0 (ad es. 12345 = 0012345).

Se il risultato della sottrazione è negativo, il minuendo viene diminuito del valore contenuto nel sottraendo. In questo caso la segnalazione di riporto non viene attivata.



Nel seguito del programma, è necessario accertarsi che risultati positivi o negativi siano gestiti in modo adeguato.

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il dato BCD a 8 cifre indicato da s, d, s1, s2, o d1 supera il campo ammesso per l'operando (da 0 a 99999999) (serie Q e System Q = codice di errore 4100).

Programma di esempio 1

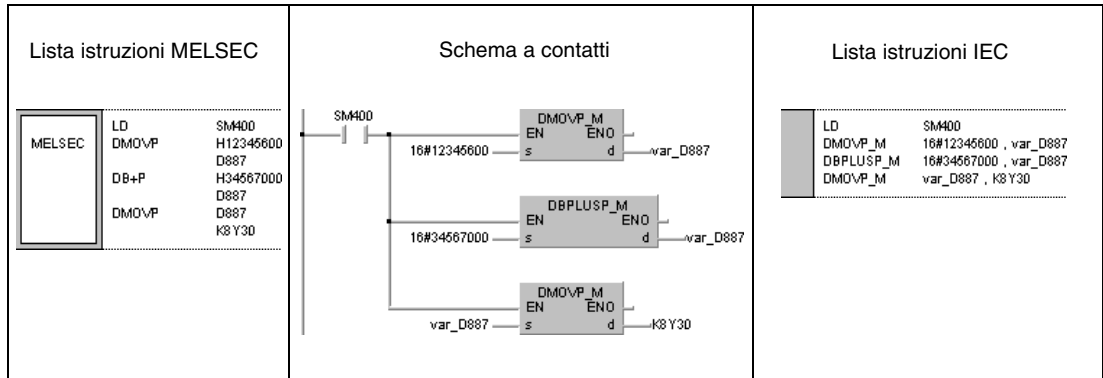
DB+P (s, d)

Il programma seguente somma il dato BCD 12345600 al dato BCD 34567000. Il risultato viene memorizzato in D887 e D888 ed inviato sulle uscite da Y30 a Y4F.

La prima riga del programma memorizza il valore 12345600 in D887 e D888 sul fronte di salita di SM400.

Il passo successivo somma il dato BCD 34567000 al dato BCD in D887 e D888.

L'istruzione DMOVP nell'ultimo passo di programma invia il risultato in D887 e D888 sulle uscite da Y30 a Y4F.



Programma di esempio 2

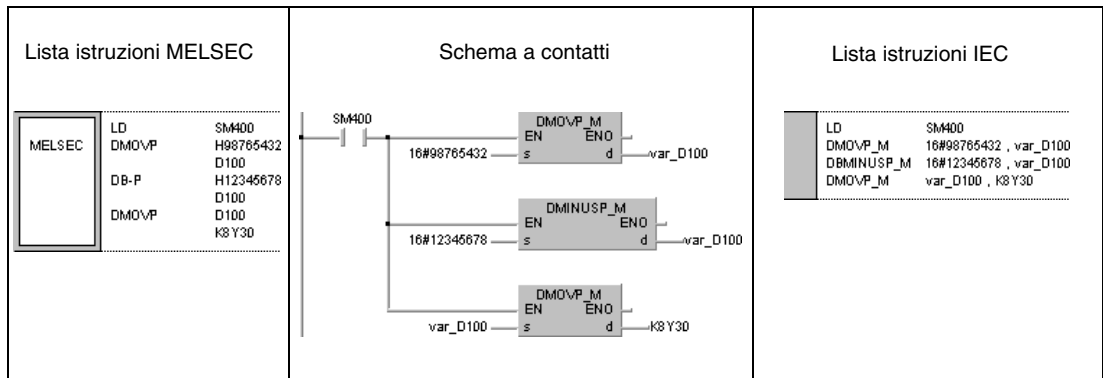
DB-P (s, d)

Il programma seguente sottrae il dato BCD 12345678 dal dato BCD 98765432. Il risultato viene memorizzato in D100 e D101 ed inviato sulle uscite da Y30 a Y4F.

La prima riga del programma memorizza il valore 98765432 in D100 e D101 sul fronte di salita di SM400.

Il passo successivo sottrae il dato BCD 12345678 dal dato BCD in D100 e D101.

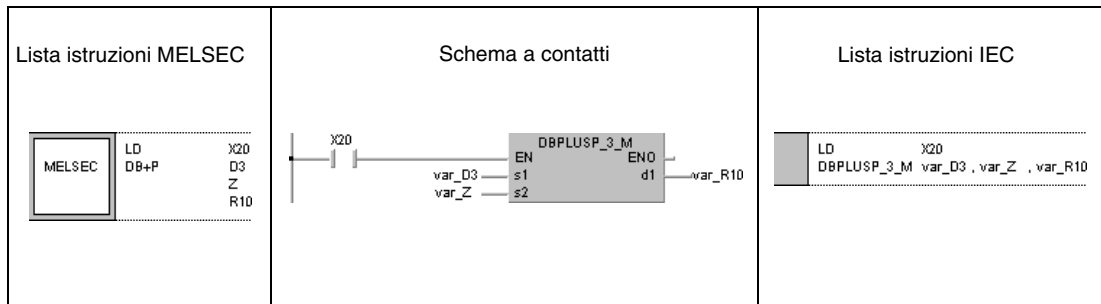
L'istruzione DMOVP nell'ultimo passo di programma invia il risultato in D100 e D101 sulle uscite da Y30 a Y4F.



DB+, DB+P, DB-, DB-P

Programma di esempio 3 DB+P (s1, s2, d1)

Il programma seguente somma il dato BCD in D3 e D4 con il dato BCD in Z e V sul fronte di salita di X20. Il risultato viene memorizzato in R10 e R11.



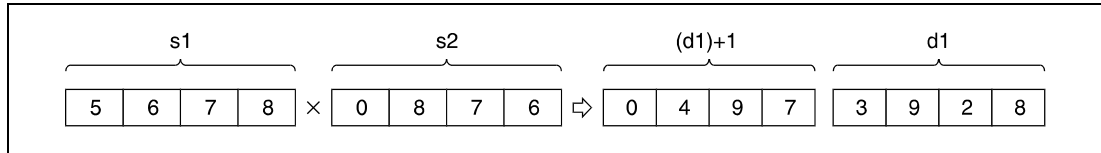
NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

Funzioni Operazioni di moltiplicazione e divisione dati BCD a 4 cifre

Bx Moltiplicazione BCD (4-cifre)

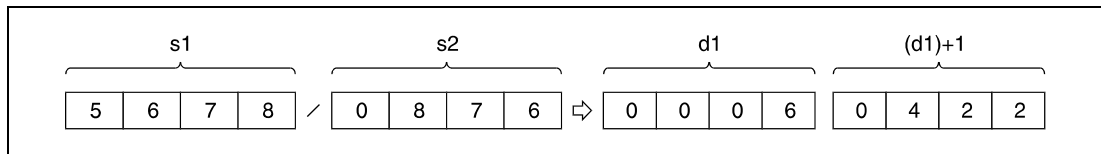
Il dato BCD a 4 cifre in s1 viene moltiplicato con il dato BCD a 4 cifre in s2. Il risultato è memorizzato in d1.



I dati BCD a 4 cifre indicati da s1 e s2 devono essere nel campo da 0 a 9999.

B/ Divisione BCD (4-cifre)

Il dato BCD a 4 cifre in s1 viene diviso per il dato BCD a 4 cifre in s2. Il risultato è memorizzato in d1.



Il risultato della divisione viene memorizzato in due elementi di un array di WORD da 16 bit. L'elemento inferiore dell'array memorizza il quoziente (4 cifre BCD), mentre l'elemento superiore memorizza il resto (4 cifre BCD).

Se d è un operando a bit, il resto della divisione non viene memorizzato.

Errori di esecuzione

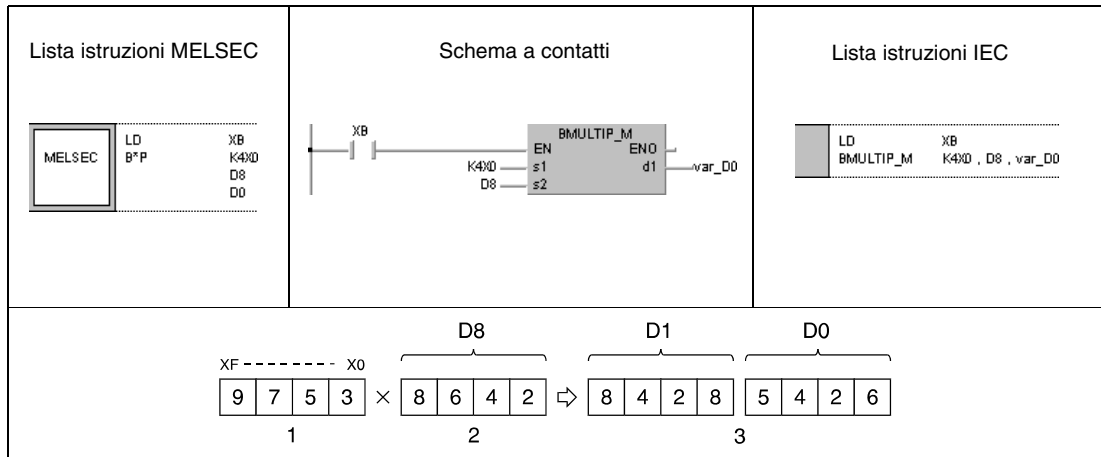
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il dato BCD a 4 cifre in s1, s2 o d supera il campo operandi ammesso (codice di errore: 4101).
- Divisione per zero (serie Q e System Q = codice di errore 4100).

Programma di esempio 1

BxP

Il programma che segue moltiplica il dato BCD da X0 a XF con il dato BCD in D8 sul fronte di salita di XB. Il risultato viene memorizzato in D0 e D1.

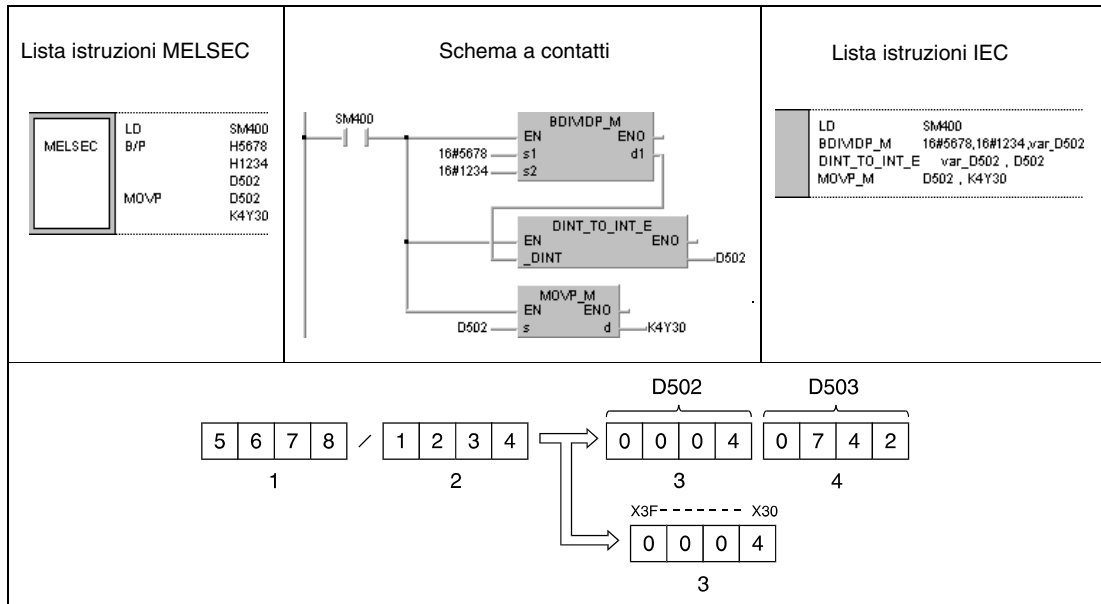


- ¹ Moltiplicando
- ² Moltiplicatore
- ³ Risultato della moltiplicazione

Programma di esempio 2

B/P

Il programma che segue divide il dato BCD 5678 per il dato BCD 1234. Il risultato viene memorizzato in D502 e il resto in D503. L'ultimo passo di programma invia il quoziente sulle uscite da Y30 a Y3F.



- ¹ Dividendo
- ² Divisore
- ³ Quoziente
- ⁴ Resto

NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.2.8 DBx, DBxP, DB/, DB/P

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

	Operandi utilizzabili															Definizione cifre	Numero di passi	Indice	Riporto	Errore					
	Operandi a bit							Operandi a word (16-bit)													Costanti		Punta-tore		Livello
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z						V	K	H (16#)	P	I
s1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●							
s2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●							
d1		●	●	●	●	●	●	●	●	●	●	●	●										●		

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale di programmazione per il corrispondente numero di passi.

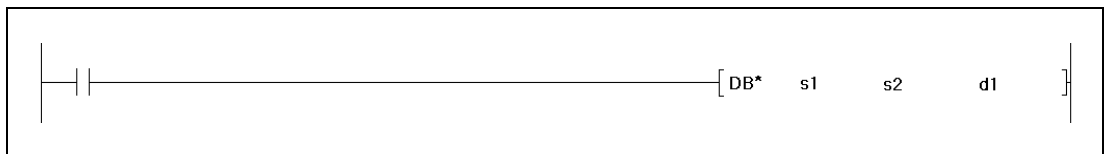
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	●	●	●	●	●	●	●	●	—	SM0	4
s2	●	●	●	●	●	●	●	●	—		
d1	●	●	●	—	—	—	—	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p>MELSEC DB* s1 s2 d1</p> </div>	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p>DBMULTI_M s1, s2, d1</p> </div>
---	--------------------------	---

GX Developer



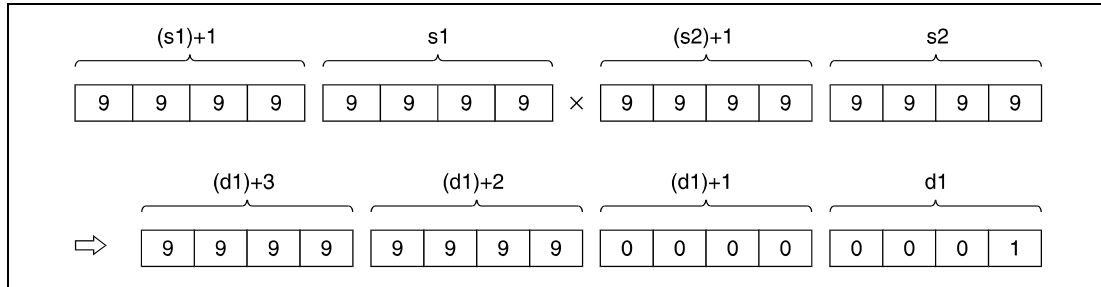
Variabili

Dato impostato	Significato	Tipo dati
s1	Moltiplicatore o dividendo, oppure primo numero dell'operando che contiene moltiplicatore o dividendo	BCD 8-cifre
s2	Moltiplicatore o divisore, oppure primo numero dell'operando che contiene moltiplicatore o divisore	BCD 8-cifre
d1	Primo numero dell'operando in cui viene memorizzato il risultato dell'operazione di moltiplicazione o divisione	BCD 16-cifre

Funzioni **Operazioni di moltiplicazione e divisione dati BCD a 8 cifre**

DBx Moltiplicazione BCD (8-cifre)

Il dato BCD a 8 cifre in s1 viene moltiplicato con il dato BCD a 8 cifre in s2. Il risultato è memorizzato in d1.



Se d1 indica un operando a bit, il risultato viene memorizzato a partire dai bit meno significativi.

Esempio:

K1: 4 bit meno significativi (da b0 a b3)

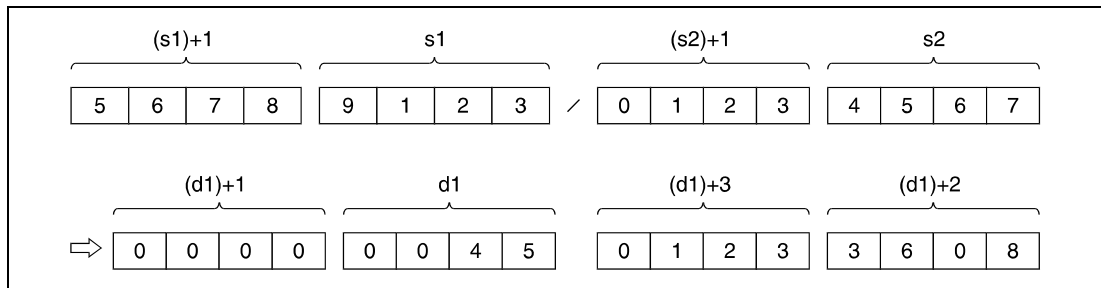
K4: 16 bit meno significativi (da b0 a b15)

K8: 32 bit (da b0 a b31)

I dati BCD a 8 cifre indicati da s1 e s2 sono nel campo da 0 a 99999999. Le cifre non indicate sono considerate come 0 (ad es. 12345 = 0012345).

DB/ Divisione BCD (8-cifre)

Il dato BCD a 8 cifre in s1 viene diviso per il dato BCD a 8 cifre in s2. Il risultato è memorizzato in d1.



Il risultato della divisione viene memorizzato in due elementi di un array di WORD da 32 bit. L'elemento inferiore dell'array memorizza il quoziente (8 cifre BCD), mentre l'elemento superiore memorizza il resto (8 cifre BCD).

Se d è un operando a bit, il resto della divisione non viene memorizzato.

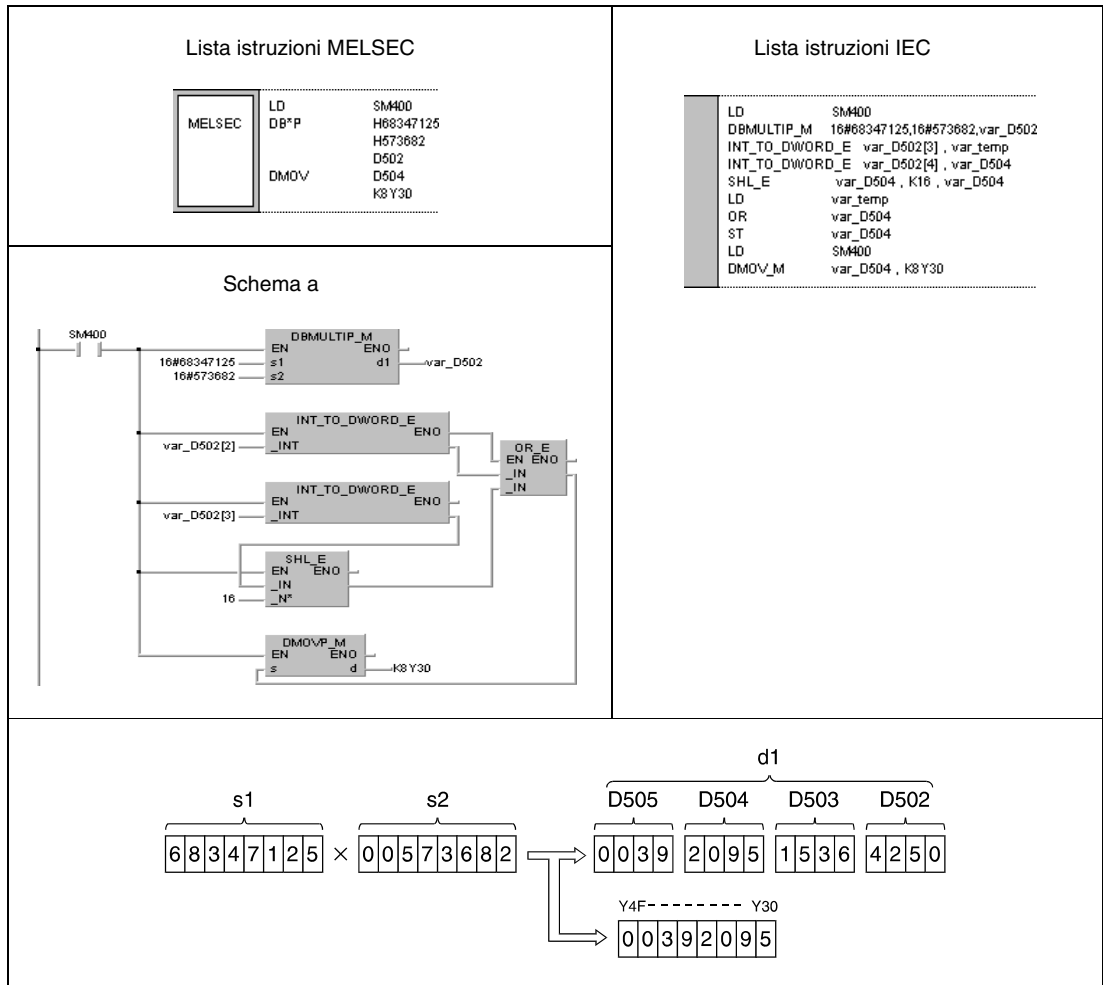
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il dato BCD a 8 cifre in s1, s2 o d supera il campo operandi ammesso (codice di errore: 4101).
- Divisione per zero (serie Q e System Q = codice di errore 4100).

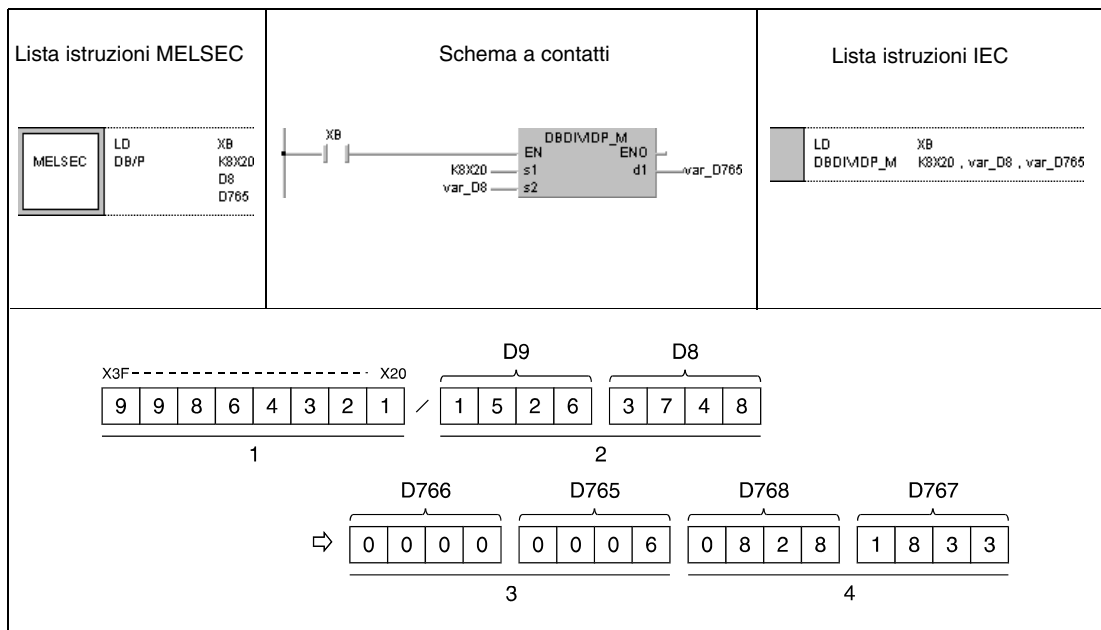
Programma di esempio 1 DBxP

Il programma che segue moltiplica il dato BCD 68347125 con il dato BCD 576682. Il risultato viene memorizzato da D502 a D505. Il passo di programma successivo invia le otto cifre più significative (D504, D505) sulle uscita da Y30 a Y4F.



Programma di esempio 2 DB/P

Il programma che segue divide il dato BCD da X20 a X3F con il dato BCD in D8 e D9 sul fronte di salita di XB. Il risultato viene memorizzato da D765 a D768.



- ¹ Dividendo
- ² Divisore
- ³ Quoziente
- ⁴ Resto

NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.2.9 E+, E+P, E-, E-P

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

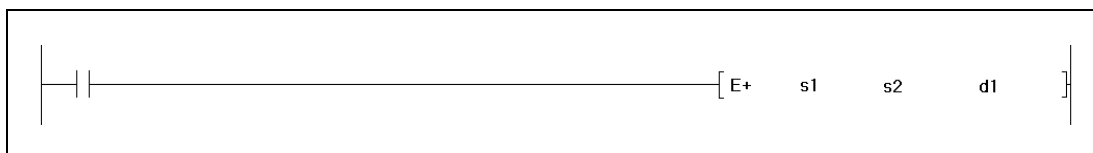
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti E			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	●	●	—	●	—	SM0	3
d	—	●	●	—	●	●	—	—	—		
s1	—	●	●	—	●	●	—	●	—	SM0	4
s2	—	●	●	—	●	●	—	●	—		
d1	—	●	●	—	●	●	—	—	—		

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Addendo o sottraendo, oppure primo numero di operando che contiene addendo o sottraendo	Numero reale
d	Addendo o minuendo, oppure primo numero di operando contenente addendo o minuendo	
s1	Addendo o minuendo, oppure primo numero di operando contenente addendo o minuendo	
s2	Addendo o sottraendo, oppure primo numero di operando che contiene addendo o sottraendo	
d1	Primo numero dell'operando in viene memorizzato il risultato	

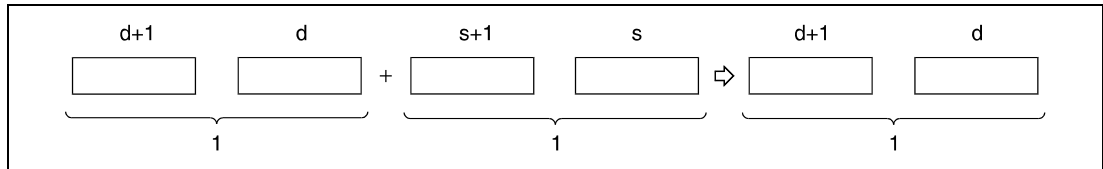
NOTA

Utilizzare comandi IEC con gli editor IEC

Funzioni Operazioni di somma e sottrazione dati in virgola mobile
E+ Somma dati in virgola mobile

● Variante 1:

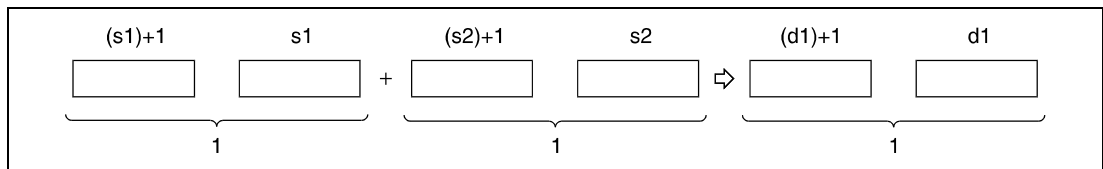
Il dato in virgola mobile in d viene sommato al dato in virgola mobile in s. Il risultato della somma viene memorizzato in d.



¹ Dati in virgola mobile, tipo dati per numeri reali

● Variante 2:

Il dato in virgola mobile in s1 viene sommato al dato in virgola mobile in s2. Il risultato è memorizzato in d1.



¹ Dati in virgola mobile, tipo dati per numeri reali

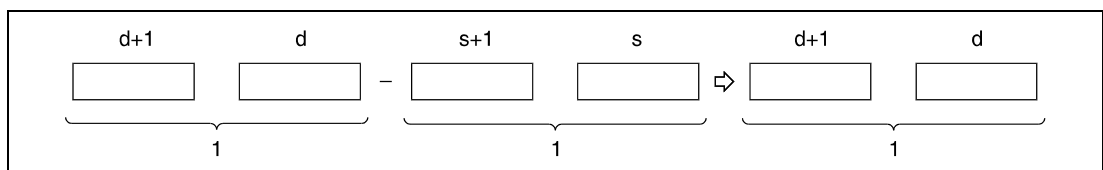
I dati in virgola mobile definiti da s, d, s1, s2, e d1 sono compresi nel campo:

$$0, \pm 2^{-127} \leq (s, d, s1, s2, d1) < \pm 2^{129}$$

E- Sottrazione dati in virgola mobile

● Variante 1:

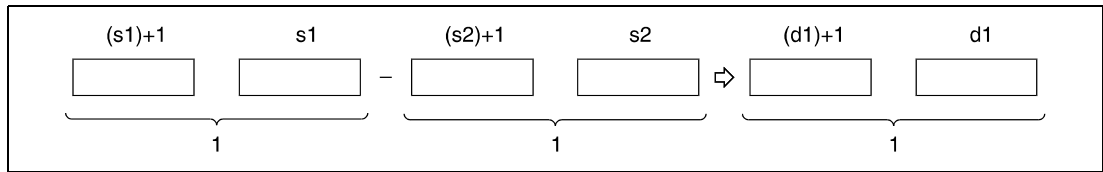
Il dato in virgola mobile in s viene sottratto dal dato in virgola mobile in s. Il risultato viene memorizzato in d.



¹ Dati in virgola mobile, tipo dati per numeri reali

● Variante 2:

Il dato in virgola mobile in s3 viene sottratto dal dato in virgola mobile in s1. Il risultato è memorizzato in d1.



¹ Dati in virgola mobile, tipo dati per numeri reali

I dati in virgola mobile definiti da s, d, s1, s2, e d1 sono compresi nel campo:

$$0, \pm 2^{-127} \leq (s, d, s1, s2, d1) < \pm 2^{129}$$

Errori di esecuzione

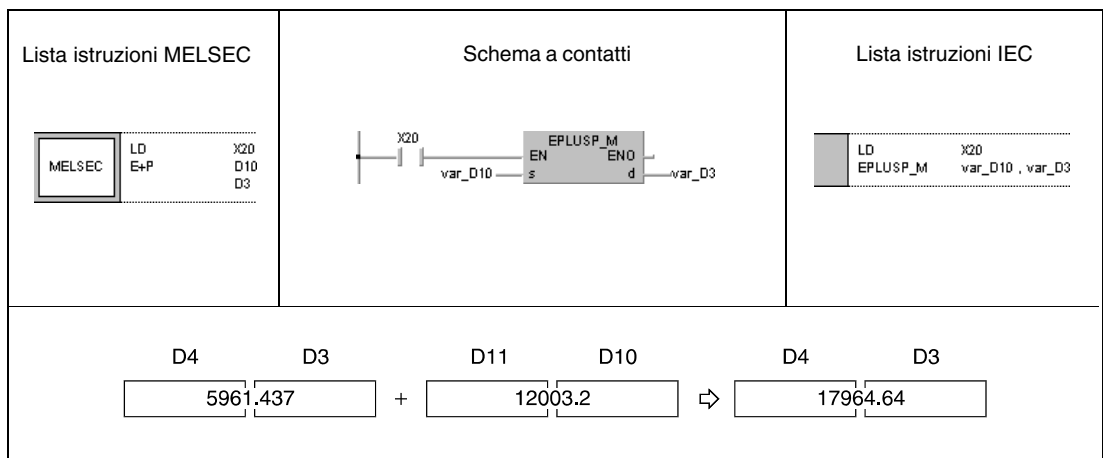
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il dato in virgola mobile in s, d, s1, s2, o d1 supera il campo di valori ammessi. (codice di errore 4100).

Programma di esempio 1

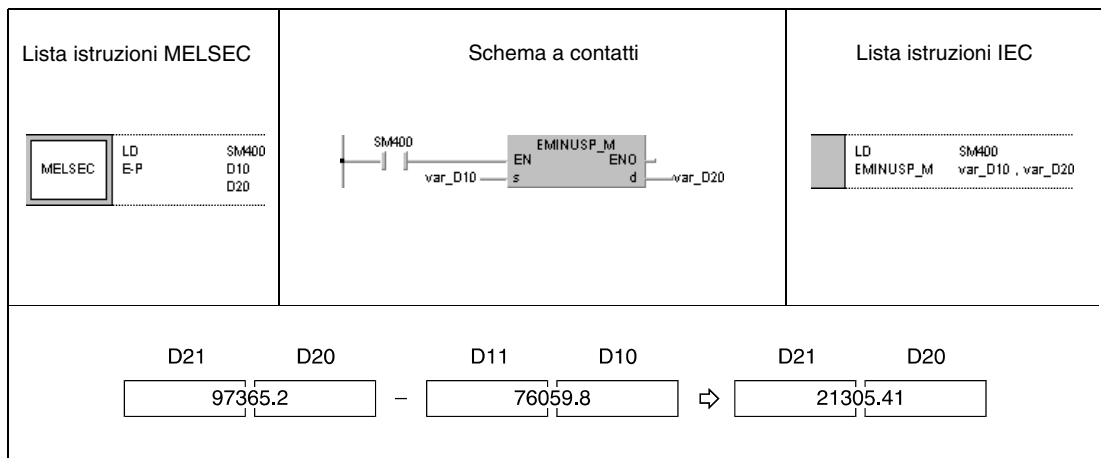
E+P (s, d)

Il programma che segue somma il dato in virgola mobile in D3 e D4 con il dato in virgola mobile in D10 e D11 sul fronte di salita di X20. Il risultato viene memorizzato in D3 e D4.



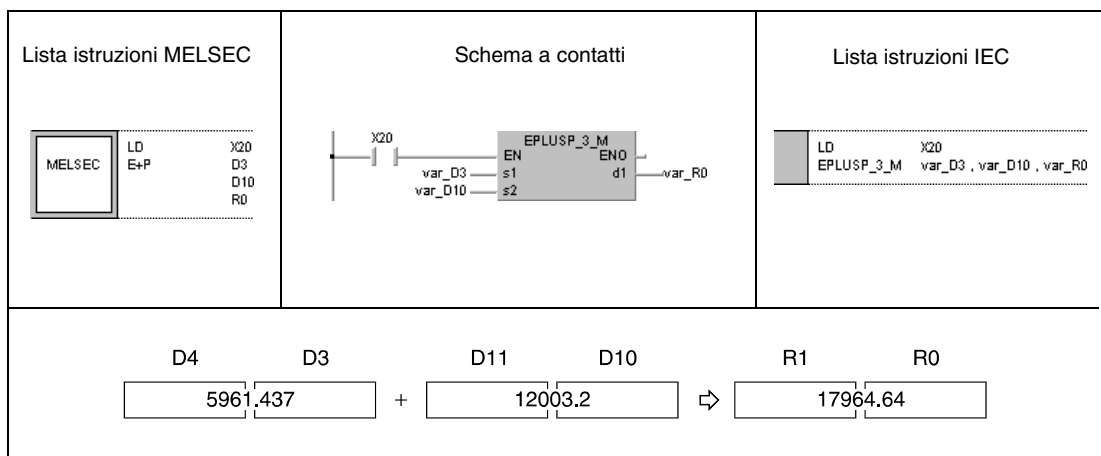
Programma di esempio 2 E-P (s, d)

Il programma che segue sottrae il dato in virgola mobile in D10 e D11 dal dato in virgola mobile in D10 e D11 sul fronte di salita di SM400. Il risultato viene memorizzato in D20 e D21.



Programma di esempio 3 E+P (s1, s2, d)

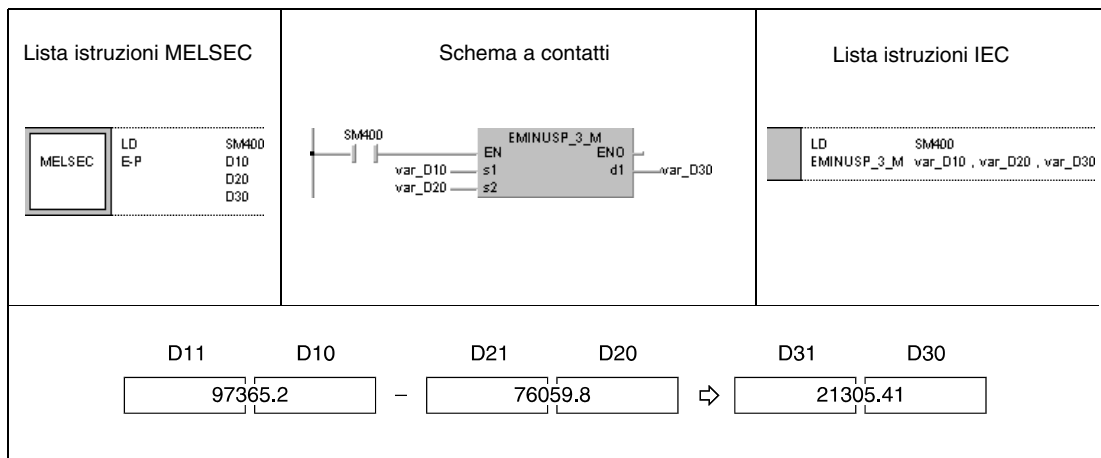
Il programma che segue somma il dato in virgola mobile in D3 e D4 con il dato in virgola mobile in D10 e D11 sul fronte di salita di X20. Il risultato viene memorizzato in R0 e R1.



E+, E+P, E-, E-P

Programma di esempio 4 E-P (s1, s2, d)

Il programma che segue sottrae il dato in virgola mobile in D20 e D21 dal dato in virgola mobile in D10 e D11 sul fronte di salita di SM400. Il risultato viene memorizzato in D30 e D31.



NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.2.10 Ex, ExP, E/, E/P

CPU

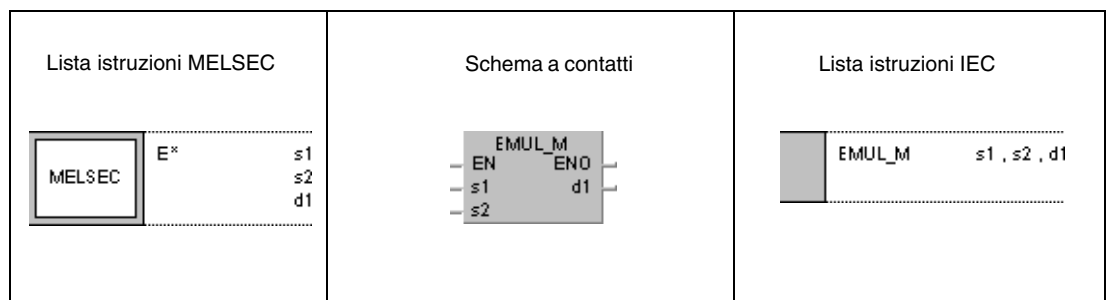
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

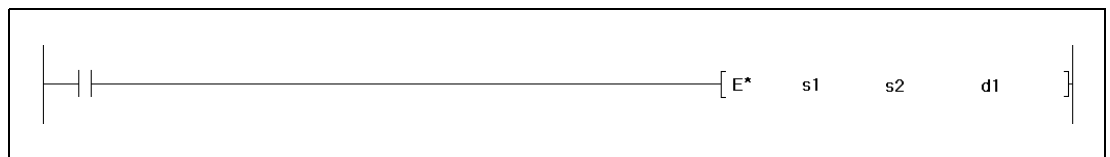
Operandi MELSEC Q

	Operandi utilizzabili									Segnale di errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti E	Altro U		
	Bit	Word		Bit	Word						
s1	—	●	●	—	●	●	—	●	—	SM0	4
s2	—	●	●	—	●	●	—	●	—		
d1	—	●	●	—	●	●	—	—	—		

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s1	Moltiplicatore o dividendo, oppure primo numero dell'operando che contiene moltiplicatore o dividendo	Numero reale
s2	Moltiplicatore o divisore, oppure primo numero dell'operando che contiene moltiplicatore o divisore	
d1	Primo numero dell'operando in cui viene memorizzato il risultato dell'operazione di moltiplicazione o divisione	

NOTA

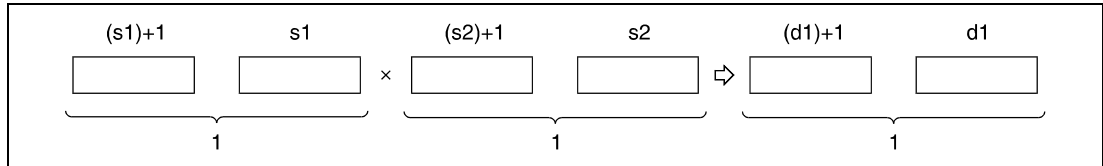
Utilizzare comandi IEC con gli editor IEC

Funzioni

Operazioni di moltiplicazione e divisione dati in virgola mobile

Ex Moltiplicazione dati in virgola mobile

Il dato in virgola mobile in s1 viene moltiplicato con il dato in virgola mobile in s2. Il risultato è memorizzato in d1.



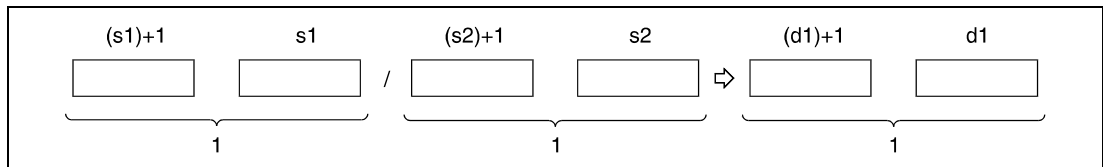
¹ Dati in virgola mobile, tipo dati per numeri reali

I dati in virgola mobile definiti da s1, s2, e d1 sono compresi nel campo:

$$0, \pm 2^{-127} \leq (s1, s2, d1) < \pm 2^{129}$$

E/ Divisione dati in virgola mobile

Il dato in virgola mobile in s1 viene diviso per il dato in virgola mobile in s2. Il risultato è memorizzato in d1.



¹ Dati in virgola mobile, tipo dati per numeri reali

I dati in virgola mobile definiti da s1, s2, e d1 sono compresi nel campo:

$$0, \pm 2^{-127} \leq (s1, s2, d1) < \pm 2^{129}$$

Errori di esecuzione

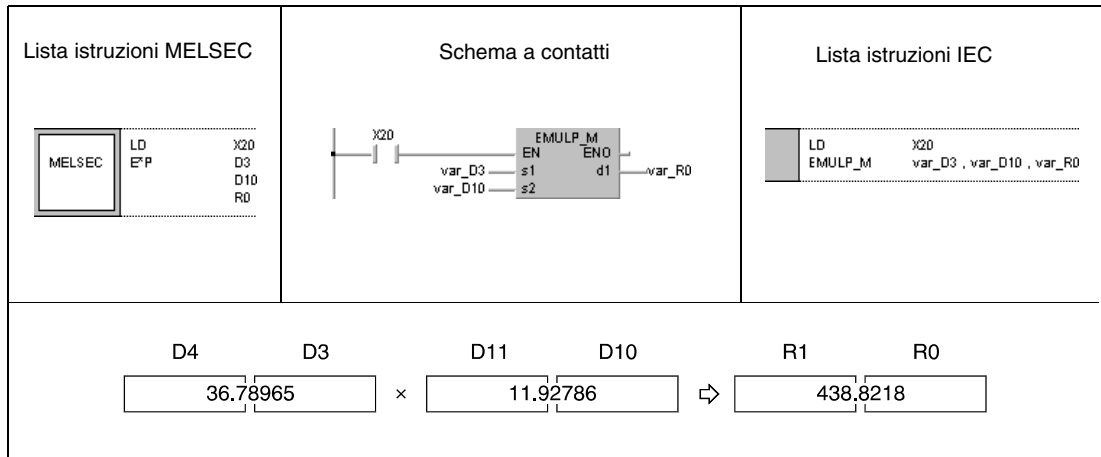
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il dato in virgola mobile in s1, s2 o d supera il campo operandi ammesso (codice di errore: 4100).
- Divisione per 0 (codice di errore 4100).

Programma di esempio 1

ExP

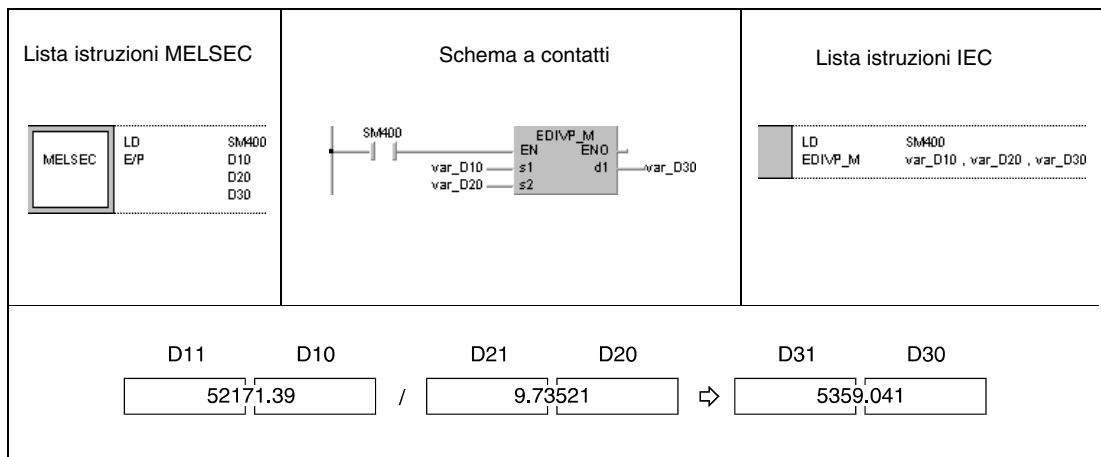
Il programma che segue moltiplica il dato in virgola mobile in D3 e D4 con il dato in virgola mobile in D10 e D11 sul fronte di salita di X20. Il risultato viene memorizzato in R0 e R1.



Programma di esempio 2

E/P

Il programma che segue divide il dato in virgola mobile in D10 e D11 per il dato in virgola mobile in D20 e D21 sul fronte di salita di SM400. Il risultato viene memorizzato in D30 e D31.



NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

BK+, BK+P, BK-, BK-P

6.2.11 BK+, BK+P, BK-, BK-P

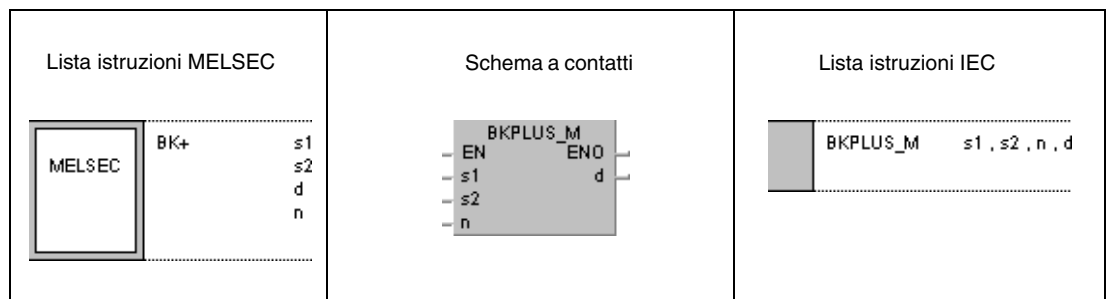
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

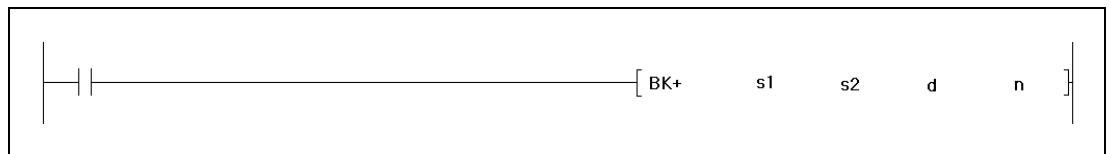
Operandi MELSEC Q

	Operandi utilizzabili									Segnale di errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	—	SM0	5
s2	—	●	●	—	—	—	—	●	—		
d	—	●	●	—	—	—	—	—	—		
n	●	●	●	●	●	●	●	●	—		

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s1	Addendo o minuendo, oppure primo numero di operando contenente addendo o minuendo	BIN 16-bit
s2	Addendo o sottraendo, oppure primo numero di operando che contiene addendo o sottraendo	
d	Primo numero di operando che contiene il risultato dell'operazione	
n	Numero dei blocchi dati	

NOTA

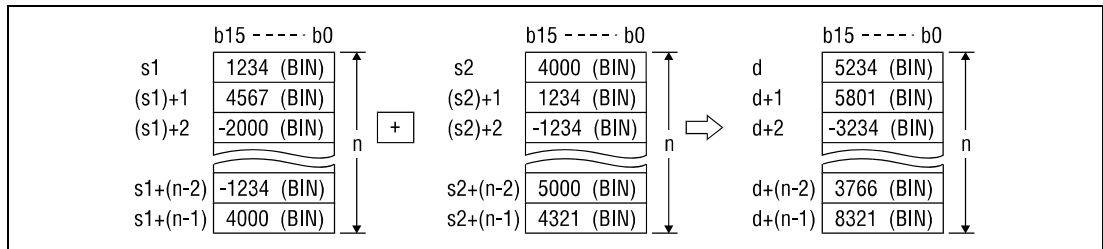
Utilizzare comandi IEC con gli editor IEC

Funzioni Operazioni di somma e sottrazione di blocchi BIN

BK+ Somma blocchi BIN

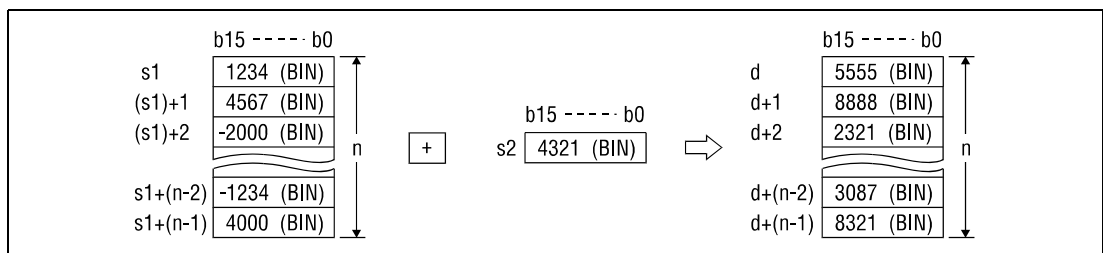
Le istruzioni per operazioni di somma su blocchi dati BIN sono composte dall'istruzione stessa, dai due operandi s1 e s2 da sommare, da un operando d in cui memorizzare il risultato, e dal numero di blocchi da confrontare.

Somma l'ennesimo blocco BIN a 16 bit in s1 con l'ennesimo blocco BIN a 16 bit in s2, iniziando con il primo numero operando. Il risultato della somma di ciascun blocco è memorizzato in d.



L'operazione di somma avviene in unità da 16 bit.

La costante indicata da s1 deve essere un numero BIN a 16 bit compreso fra -32768 e 32767.



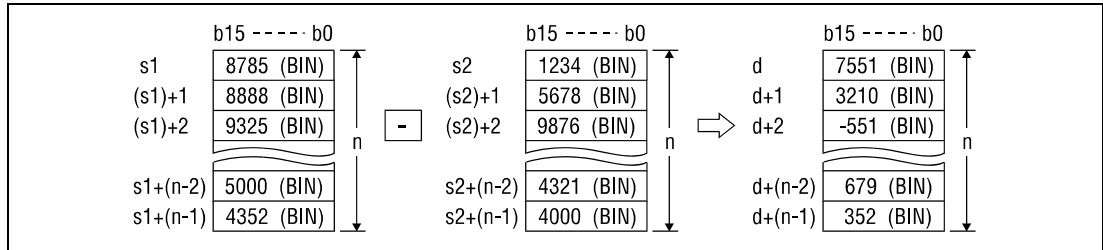
Il bit più significativo di ciascun operando determina se il dato in s1, s2 o d è positivo (bit = 0) o negativo (bit = 1).

Se vengono scavalcati il bit meno significativo o quello più significativo di un blocco il riporto non viene segnalato.

BK- Sottrazione blocchi BIN

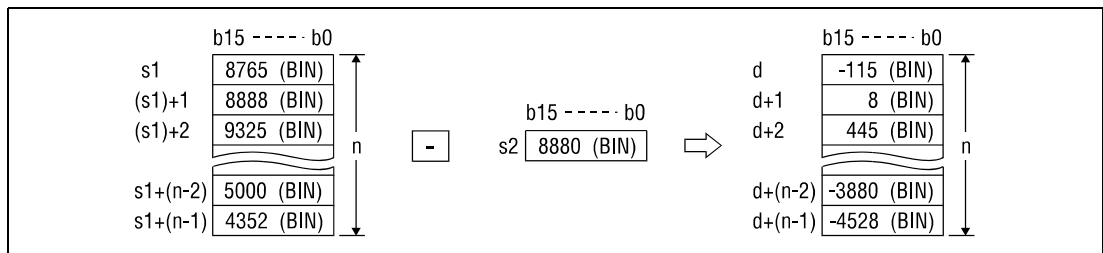
Le istruzioni per operazioni di sottrazione su blocchi dati BIN sono composte dall'istruzione stessa, dai due operandi s1 e s2 della sottrazione, da un operando d in cui memorizzare il risultato, e dal numero di blocchi da sottrarre.

Sottrae l'ennesimo blocco BIN a 16 bit in s2 dall'ennesimo blocco BIN a 16 bit in s1, iniziando con il primo numero operando. Il risultato della somma di ciascun blocco è memorizzato in d.



L'operazione di sottrazione avviene in unità da 16 bit.

La costante indicata da s2 deve essere un numero BIN a 16 bit compreso fra -32768 e 32767.



Il bit più significativo di ciascun operando determina se il dato in s1, s2 o d è positivo (bit = 0) o negativo (bit = 1).

Se vengono scavalcati il bit meno significativo o quello più significativo di un blocco il riporto non viene segnalato.

Errori di esecuzione

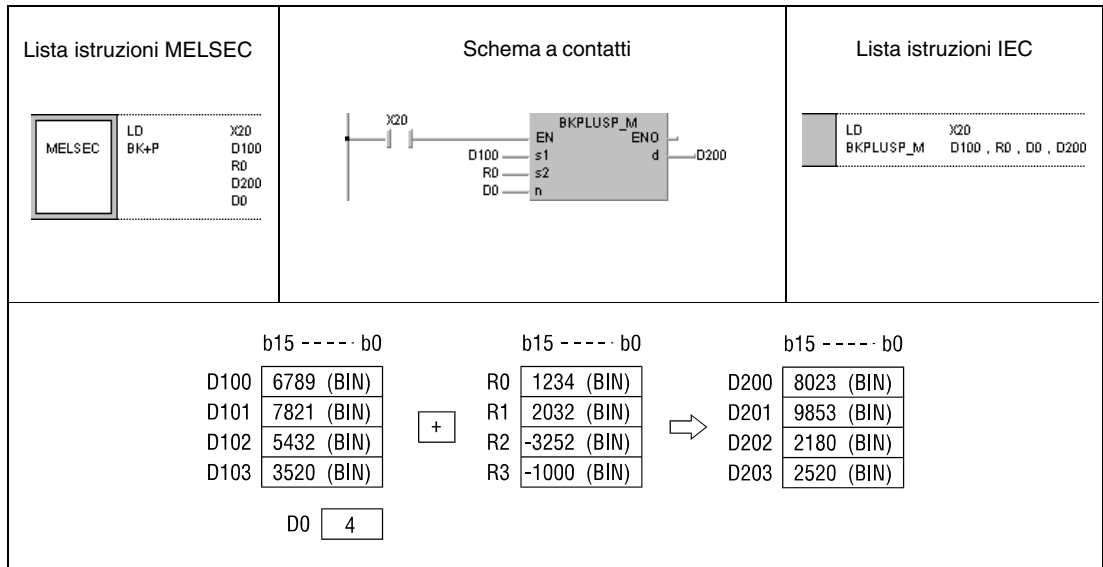
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il numero di blocchi dati in s1, s2 o d supera il campo operando ammesso.
- L'operando s1 si sovrappone agli operandi s2 o d.

Programma di esempio 1

BK+P

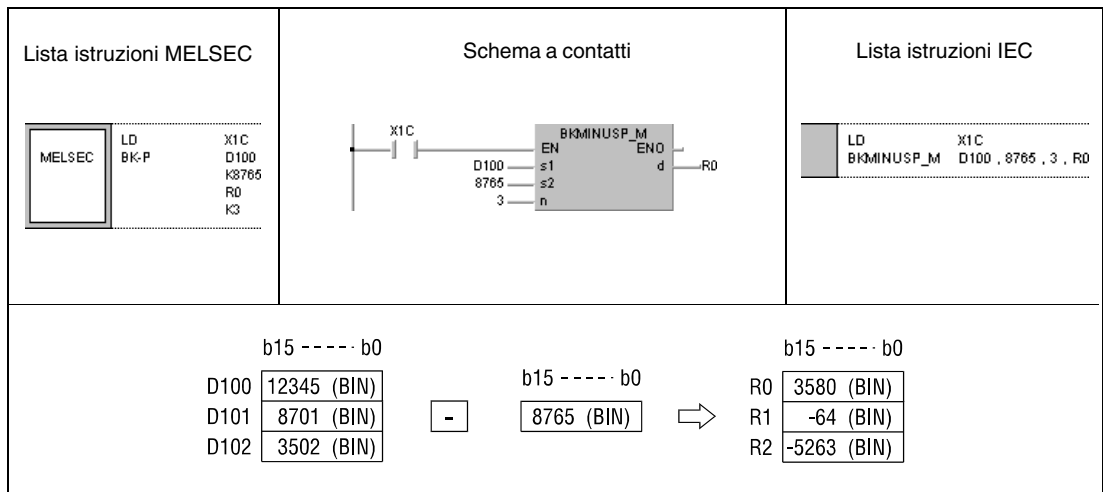
Sul fronte di salita di X20, il programma che segue somma il blocco dati BIN che inizia da D100 con il blocco dati BIN che inizia da R0. Il risultato dell'operazione viene memorizzato a partire da D200. Il numero di blocchi da sommare (4) è contenuto in D0



Programma di esempio 2

BK-P

Il programma seguente sottrae la costante 8765 dal blocco dati BIN che inizia da D100 sul fronte di salita di X1C. Il risultato dell'operazione viene memorizzato a partire da R0. Il numero di blocchi dati da sottrarre (3) è indicato dalla costante K3.



\$+, \$+P

6.2.12 \$+, \$+P

CPU

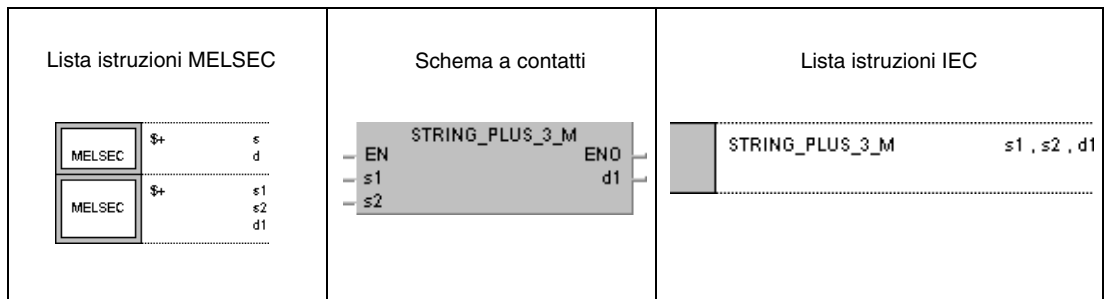
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

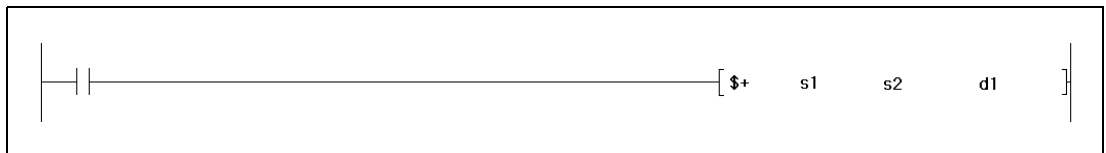
Operandi MELSEC Q

	Operandi utilizzabili									Segnale di errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti \$	Altro U		
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	●	—	SM0	3
d	—	●	●	—	—	—	—	—	—		
s1	—	●	●	—	—	—	—	●	—	SM0	4
s2	—	●	●	—	—	—	—	●	—		
d1	—	●	●	—	—	—	—	—	—		

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Dato da collegare o primo numero dell'operando che contiene questo dato	Stringa caratteri
d	Primo numero di operando che contiene il risultato dell'operazione	
s1	Dato da collegare o primo numero dell'operando che contiene questo dato	
s2	Dato da collegare o primo numero dell'operando che contiene questo dato	
d1	Primo numero di operando che contiene il risultato dell'operazione	

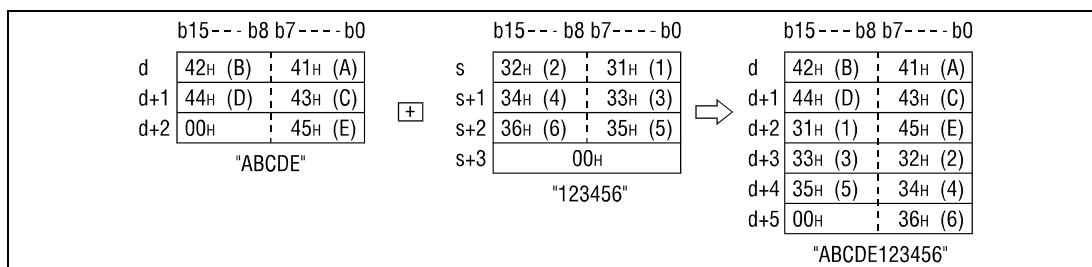
Funzioni Operazioni di collegamento di stringhe di caratteri

\$+ Collegamento stringhe di caratteri

● Variante 1:

I dati della stringa di caratteri s vengono appesi ai dati dei caratteri contenuti in d. La stringa risultante viene memorizzata in d.

La stringa di caratteri risultante inizia con il carattere contenuto nel byte meno significativo di d e termina con il codice "00H" in s.

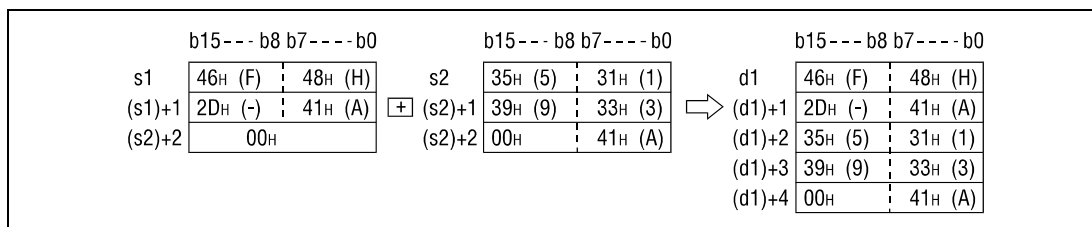


Il codice "00H" indica la fine di una stringa di caratteri. Quando vengono collegate due stringhe, questo codice viene ignorato nella prima stringa, mentre lo "00H" della seconda stringa segna la fine della stringa risultante.

● Variante 2:

I dati della stringa di caratteri in s2 vengono appesi ai caratteri in s1. La stringa di caratteri risultante viene memorizzata in d1.

La stringa di caratteri risultante inizia con il carattere contenuto nel byte meno significativo di s1 e termina con il codice "00H" in s2.



Il codice "00H" indica la fine di una stringa di caratteri. Quando vengono collegate due stringhe, questo codice viene ignorato nella prima stringa, mentre lo "00H" della seconda stringa segna la fine della stringa risultante.

Errori di esecuzione

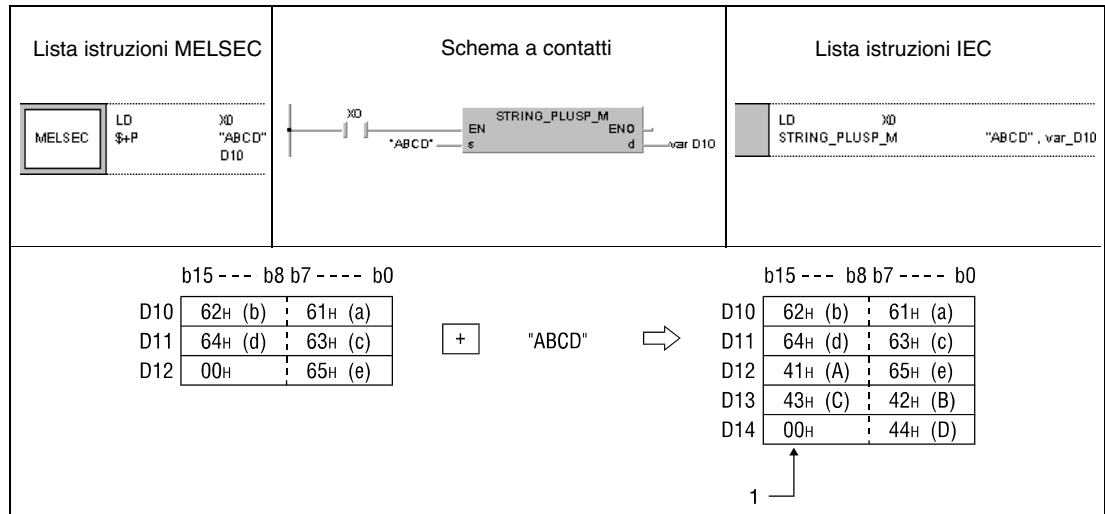
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- La stringa di caratteri risultante non può essere memorizzata (codice di errore 4100).
- I numeri degli operandi indicati da s, d, s1, s2 e d1 sono sovrapposti (codice di errore 4101).

Programma di esempio 1

S+P

Sul fronte di salita di X0, il programma che segue collega i dati della stringa contenuta da D10 a D12 alla stringa di caratteri "ABCD". La stringa di caratteri risultante viene memorizzata da D10 a D14.

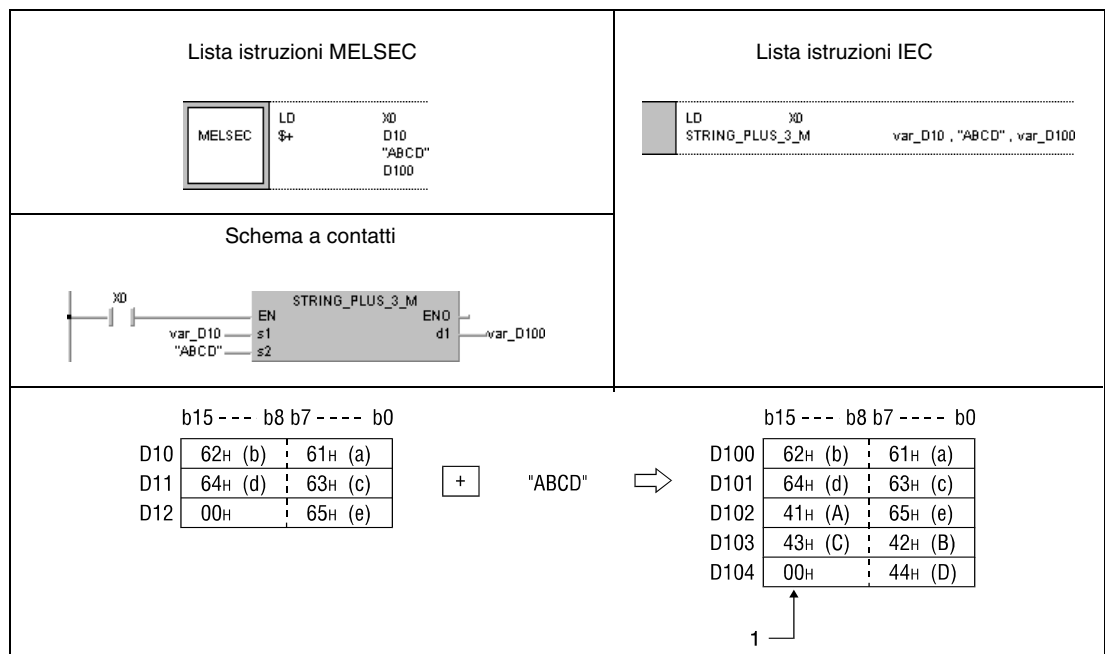


¹ "00H" indica la fine della stringa e viene inserito automaticamente.

Programma di esempio 2

S+

Fintanto che X0 è a 1, il programma che segue collega i dati della stringa contenuta da D10 a D12 alla stringa di caratteri "ABCD". La stringa di caratteri risultante viene memorizzata da D101 a D104.



¹ "00H" indica la fine della stringa e viene inserito automaticamente.

NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.2.13 INC, INCP, DEC, DECP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

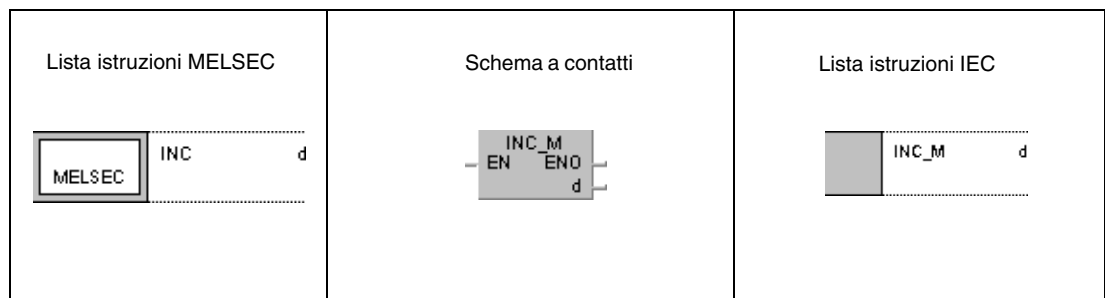
	Operandi utilizzabili															Definizione cifre K1 ↓ K4	Numero di passi 3 ↓ 1	Indice ●	Riporto M9012	Errore M9010 M9011
	Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello					
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z					
d	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●					

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale di programmazione per il corrispondente numero di passi.

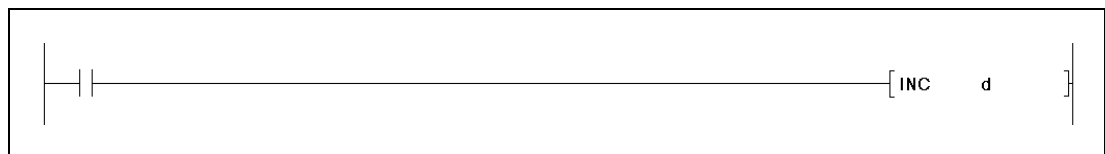
Operandi MELSEC Q

	Operandi utilizzabili									Segnale di errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
d	●	●	●	●	●	●	●	—	—	—	2

GX IEC Developer



GX Developer



Variabili

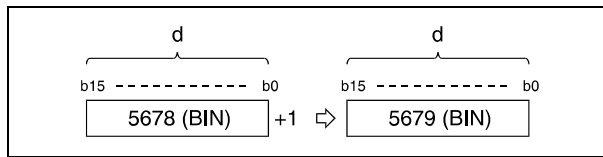
Dati impostati	Significato	Tipo dati
d	Primo numero di operando interessato dall'operazione INC (somma 1) o DEC (sottrae 1).	BIN 16-bit

INC, INCP, DEC, DECP

Funzioni Operazioni di incremento e decremento dati BIN a 16 bit

INC Incremento BIN 16-bit

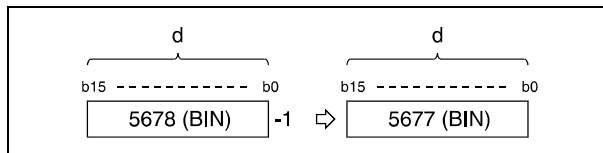
Somma 1 all'operando indicato da d (16 bit).



Se il contenuto di d è 32767, il risultato dopo l'incremento è -32768.

DEC Decremento BIN 16-bit

Sottrae 1 dall'operando indicato da d (16 bit).



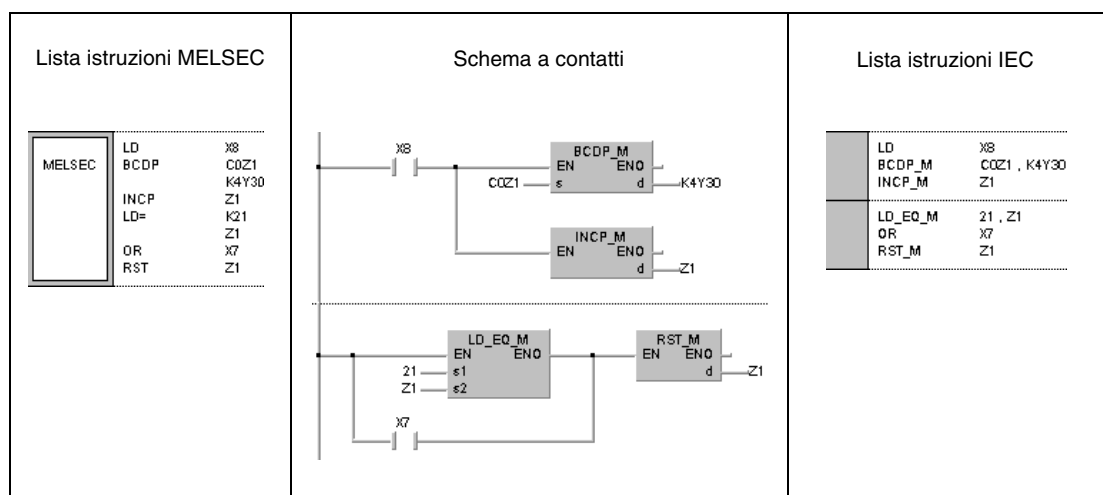
Se il contenuto di d è 0, il risultato dopo il decremento è -1.

Se il contenuto di d è -32768, il risultato dopo il decremento è 32767.

Programma di esempio 1

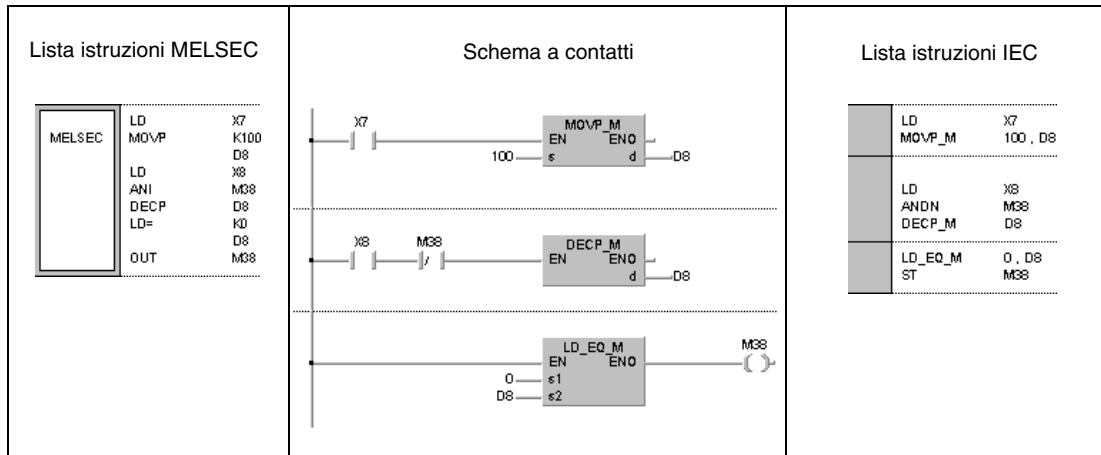
INCP

Sul fronte di salita di X9, il programma che segue invia il valore attuale del contatore (valore nominale = 9999) da C0 a C20 (C0 indicizzato da Z1) sulle uscite da Y30 a Y3F come dato BCD. Z1 viene azzerato (RST Z1), se Z1 è uguale a 21 (LD = K21 Z1) o se è attivo l'ingresso di reset X7.



Programma di esempio 2 DECP

L'esempio che segue mostra il programma di un contatore indietro. Il programma memorizza il valore 100 in D8 sul fronte di salita di X7. Fintanto che M38 è uguale a zero, il dato in D8 viene decrementato di 1 sul fronte di salita di X8. Quando D8 = 0, M38 viene settato a 1.



DINC, DINCP, DDEC, DDECP

6.2.14 DINC, DINCP, DDEC, DDECP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

	Operandi utilizzabili													Definizione cifre K1 ↓ K8	Numero di passi 3 ● ²	Indice ●	Riporto	Errore						
	Operandi a bit						Operandi a word (16-bit)						Costanti				Punta- tore		Livello		M9012	M9010 M9011		
	X	Y	M	L	S	B	F	T	C	D	W	R	A0				A1	Z	V	K	H (16#)	P	I	N
d	●	●	●	●	●	●	●	●	●	●	●	●	●	●	● ¹									●

¹ Tranne CPU AnN

² Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale di programmazione per il corrispondente numero di passi.

Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
d	●	●	●	●	●	●	●	—	—	—	2 ¹⁾

¹ Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.

Se viene usata una CPU QnA oppure una CPU a processore singolo del System Q: 2

Se viene usata una CPU multiprocessore del System Q con

operandi interni a word (tranne file registri ZR): 3

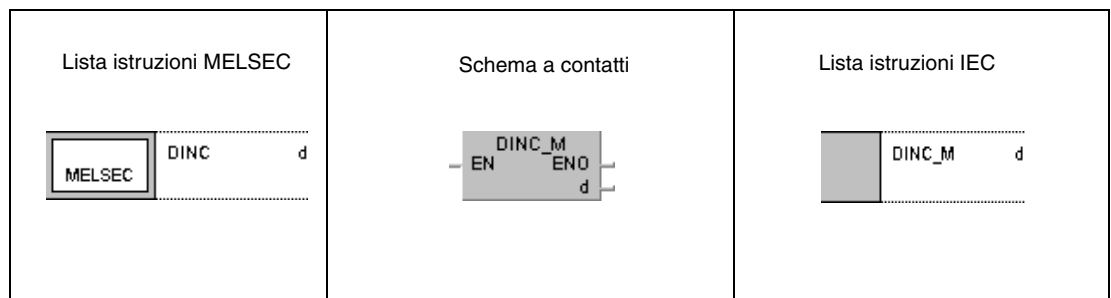
costanti: 3

Operandi a bit con numero operando multiplo di 16, in cui la cifra indicata è K8,

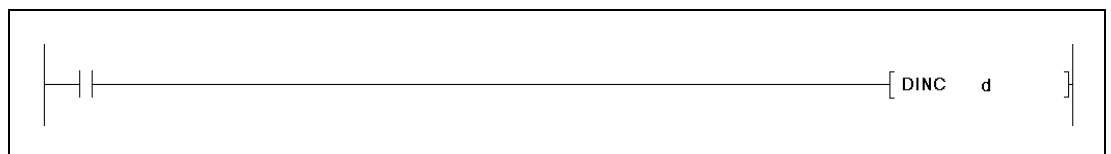
e che non usa indicizzazione: 3

Se viene usata una CPU multiprocessore del System Q con operandi diversi da quelli menzionati:2

GX IEC Developer



GX Developer



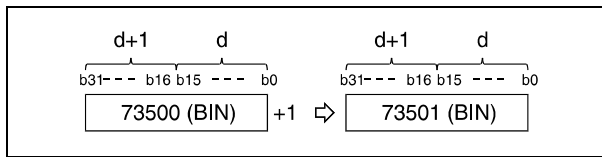
Variabili

Dati impostati	Significato	Tipo dati
d	Primo numero di operando interessato dall'operazione INC (somma 1) o DEC (sottrae 1).	BIN 32-bit

Funzioni Operazioni di incremento e decremento dati BIN a 32 bit

DINC Incremento BIN 32-bit

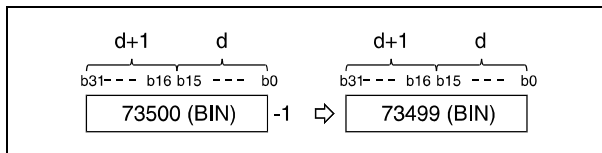
Somma 1 all'operando indicato da d (32 bit).



Se il contenuto di d è 2147483647, il risultato dopo l'incremento è -2147483648.

DDEC Decremento BIN 32-bit

Sottrae 1 dall'operando indicato da d (16 bit).



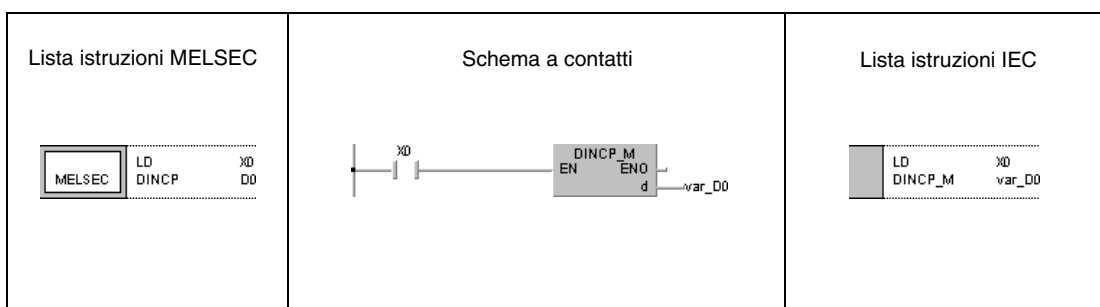
Se il contenuto di d è 0, il risultato dopo il decremento è -1.

Se il contenuto di d è -2147483647, il risultato dopo il decremento è 2147483647.

Programma di esempio 1

DINCP

Sul fronte di salita di X0, il programma che segue somma 1 al dato in D0.

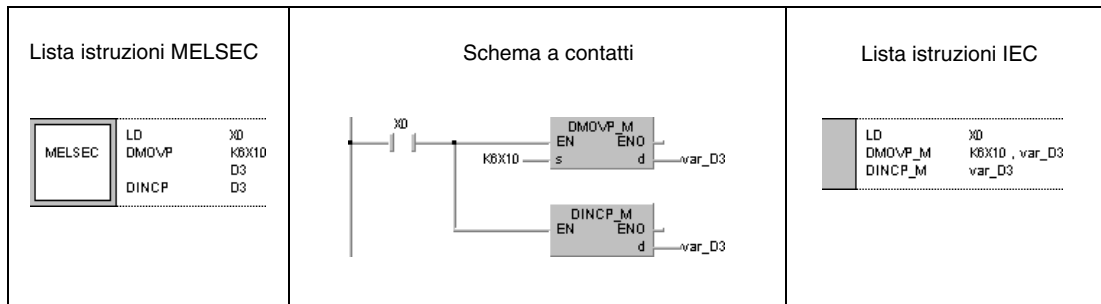


DINC, DINCP, DDEC, DDECP

Programma di esempio 2 DINC

di esempio 2

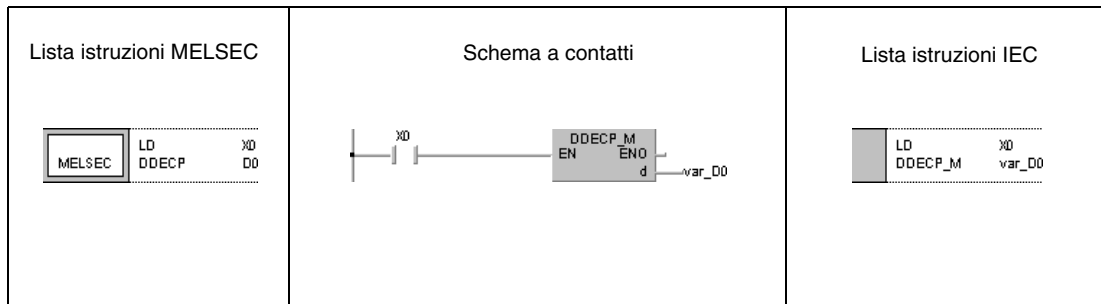
Sul fronte di salita di X0, il programma che segue somma 1 al dato da X10 a X27. Il risultato viene memorizzato in D3 e D4.



Programma di esempio 3 DDECP

di esempio 3

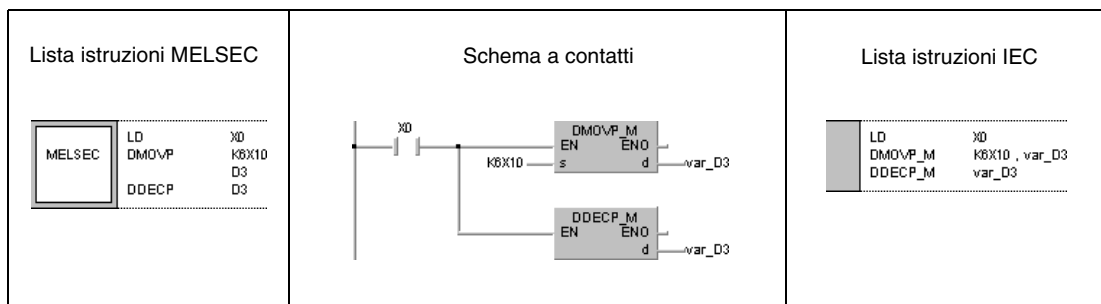
Sul fronte di salita di X0, il programma che segue sottrae 1 dal dato in D0.



Programma di esempio 4 DDECP

di esempio 4

Sul fronte di salita di X0, il programma che segue sottrae 1 dal dato contenuto da X10 a X27. Il risultato viene memorizzato in D3 e D4.



NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.3 Istruzioni di conversione dati

Le istruzioni che seguono convertono dati fra tipi differenti:

Conversione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
BIN (16/32-bit) ↓ BCD (4-/8-cifre)	BCD	BCD_M
	BCDP	BCDP_M
	DBCD	DBCD_M
	DBCDP	DBCDP_M
BCD (4-/8-cifre) ↓ BIN (16/32-bit)	BIN	BIN_M
	BINP	BINP_M
	DBIN	DBIN_M
	DBINP	DBINP_M
BIN (16/32-bit) ↓ Dati virgola mobile	FLT	FLT_M
	FLTP	FLTP_M
	DFLT	DFLT_M
	DFLTP	DFLTP_M
Dati virgola mobile ↓ BIN (16/32-bit)	INT	INT_MD
		INT_E_MD
	INTP	INT_P_MD
		INT_P_E_MD
	DINT	DINT_MD
		DINT_E_MD
DINTP	DINT_P_MD	
	DINT_P_E_MD	
BIN 16-bit ↓ BIN 32-bit	DBL	DBL_M
	DBLP	DBLP_M
BIN 32-bit ↓ BIN 16-bit	WORD	WORD_M
	WORDP	WORDP_M
BIN (16/32-bit) ↓ Dati codice GRAY	GRY	GRY_M
	GRYP	GRYP_M
	DGRY	DGRY_M
	DGRYP	DGRYP_M
Dati codice GRAY ↓ BIN (16/32-bit)	GBIN	GBIN_M
	GBINP	GBINP_M
	DGBIN	DGBIN_M
	DGBINP	DGBINP_M
Inversione di segno BIN (16/32-bit) (complemento a 2)	NEG	NEG_M
	NEGP	NEGP_M
	DNEG	DNEG_M
	DNEGP	DNEGP_M
Inversione di segno Dati virgola mobile	ENEG	ENEG_M
	ENEGP	ENEGP_M
Blocco BIN (16-bit) ↓ Blocco BCD (4 cifre)	BKBCD	BKBCD_M
	BKBCDP	BKBCDP_M
Blocco BCD (4 cifre) ↓ Blocco BIN (16-bit)	BKBIN	BKBIN_M
	BKBINP	BKBINP_M

6.3.1 BCD, BCDP, DBCD, DBCDP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

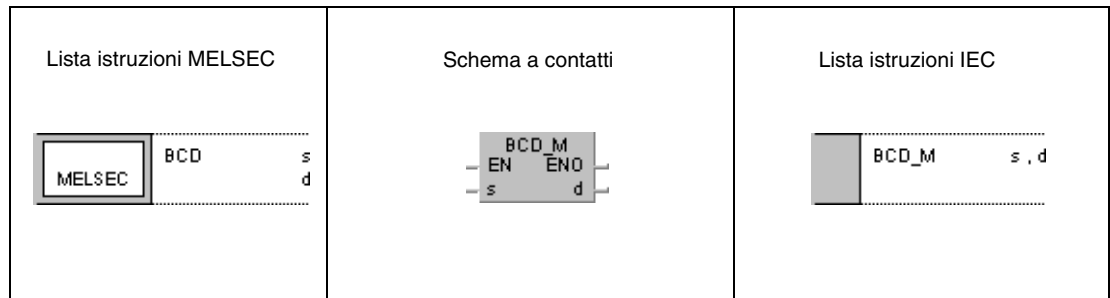
Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Riporto	Errore											
Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello				M9012	M9010 M9011											
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H (16#)	P	I	N										
BCD																														
s	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●										K1 ↓ K4	5	●		●	
d		●	●	●	●	●	●	●	●	●	●	●	●	●	●												1			
DBCD																														
s	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●										K1 ↓ K8	9	●		●	
d		●	●	●	●	●	●	●	●	●	●	●	●	●	●											1				

¹ Consultare la sezione "Programmazione con CPU AnA, AnAS e AnU" di questo manuale di programmazione per il corrispondente numero di passi.

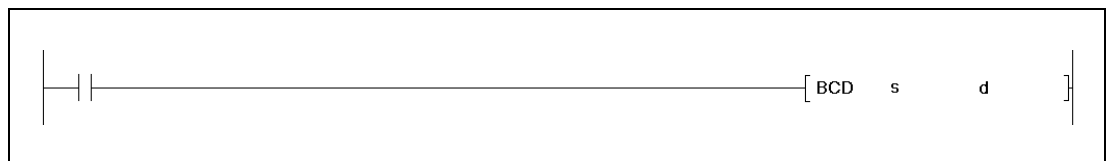
Operandi MELSEC Q

Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word							
s	●	●	●	●	●	●	●	—	—	SM0	3
d	●	●	●	●	●	●	—	—			

GX IEC Developer



GX Developer



Variabili

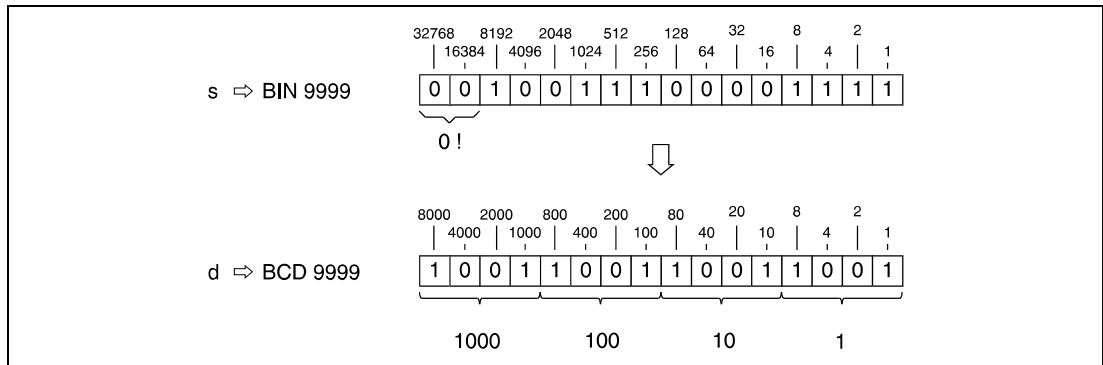
Dati impostati	Significato	Tipo dati
s	Dato BIN o primo numero dell'operando contenente il dato BIN	BIN 16/32-bit
d	Primo numero dell'operando che memorizza il dato BCD.	BCD 4-/8-cifre

Funzioni Conversione da dati BIN a dati BCD

BCD Conversione da dati BIN a 16 bit a dati BCD a 4 cifre

Il dato BIN in s (da 0 a 9999) viene convertito in dato BCD. Il risultato è memorizzato in d.

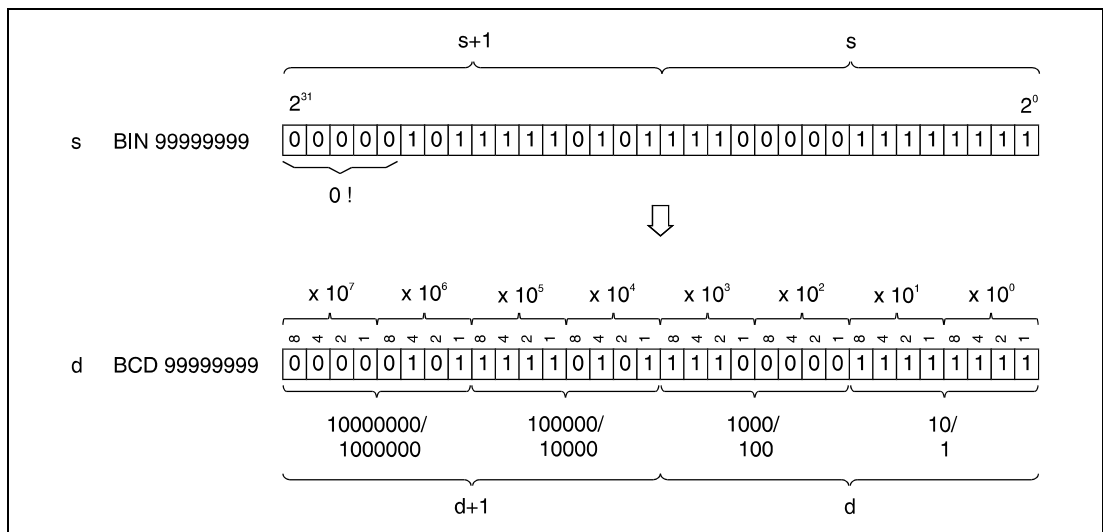
I due bit più significativi del dato BIN in s devono essere azzerati quando il dato viene convertito in dato BCD a 4-cifre.



DBCD Conversione da dati BIN a 32 bit a dati BCD a 8 cifre

Il dato BIN in s (da 0 a 99999999) viene convertito in dato BCD. Il risultato è memorizzato in d.

I cinque bit più significativi del dato BIN in s devono essere azzerati quando il dato viene convertito in dato BCD a 8-cifre.



Errori di esecuzione

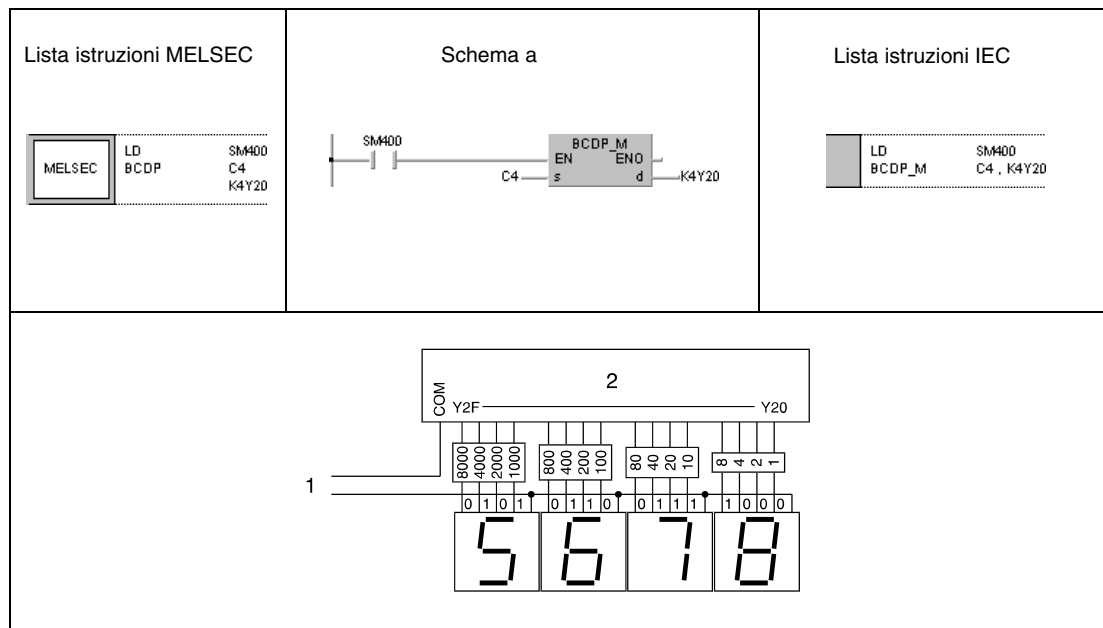
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il dato BIN a 16 bit supera il campo valori ammesso da 0 a 9999. (serie Q e System Q = codice di errore 4100).
- Il dato BIN a 32 bit in s+1 supera il campo valori ammesso da 0 a 99999999. (serie Q e System Q = codice di errore 4100).

Esempio di programma

BCDP

Il programma seguente invia il valore attuale di C4 (5678) sulle uscite da Y20 a Y2F. Il modulo di uscita visualizza il valore sul display.



¹ Alimentatore uscite

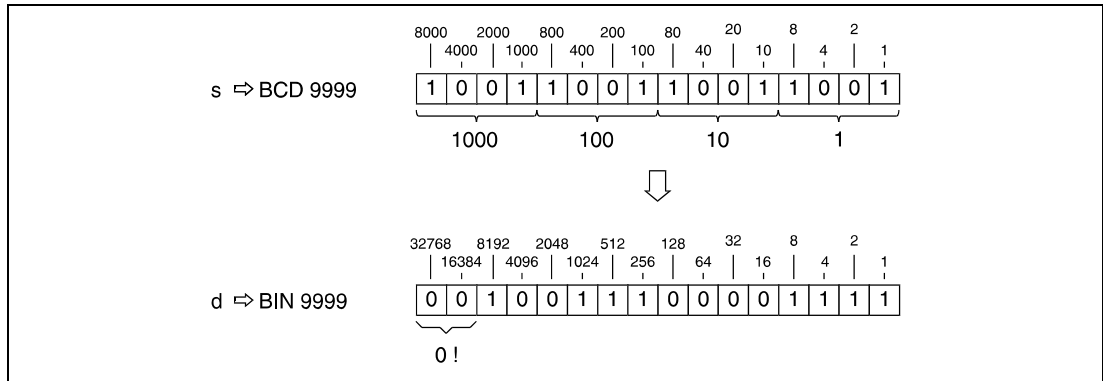
² Modulo di uscita

Funzioni Conversione da dati BCD a dati BIN

BIN Conversione da dati BCD a 4 cifre a dati BIN a 16 bit

Il dato BCD in s (da 0 a 9999) viene convertito in dato BIN. Il risultato è memorizzato in d.

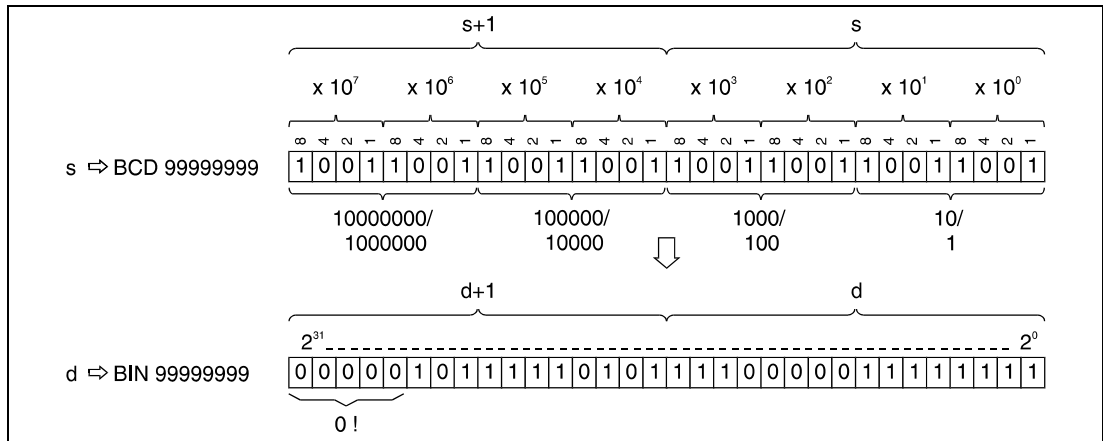
I due bit più significativi del dato BIN in d devono essere azzerati quando il dato viene convertito in dato BCD a 4-cifre.



DBIN Conversione da dati BCD a 8 cifre a dati BIN a 32 bit

Il dato BCD in s (da 0 a 99999999) viene convertito in dato BIN. Il risultato è memorizzato in d.

I cinque bit più significativi del dato BIN in d devono essere azzerati quando il dato viene convertito in dato BCD a 8-cifre.



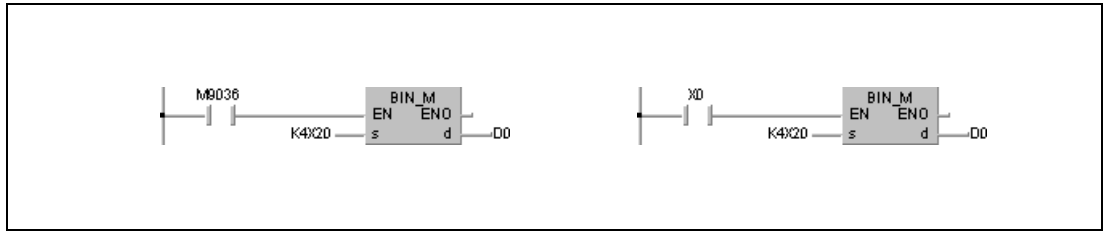
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Le singole cifre in s non sono comprese fra 0 e 9.
- Sulle CPU della serie Q o del System Q questo errore può essere disabilitato impostando SM722 su ON. Tuttavia l'istruzione non viene eseguita, indipendentemente dallo stato di SM722, se il valore specificato in s è fuori del campo ammesso.

NOTA

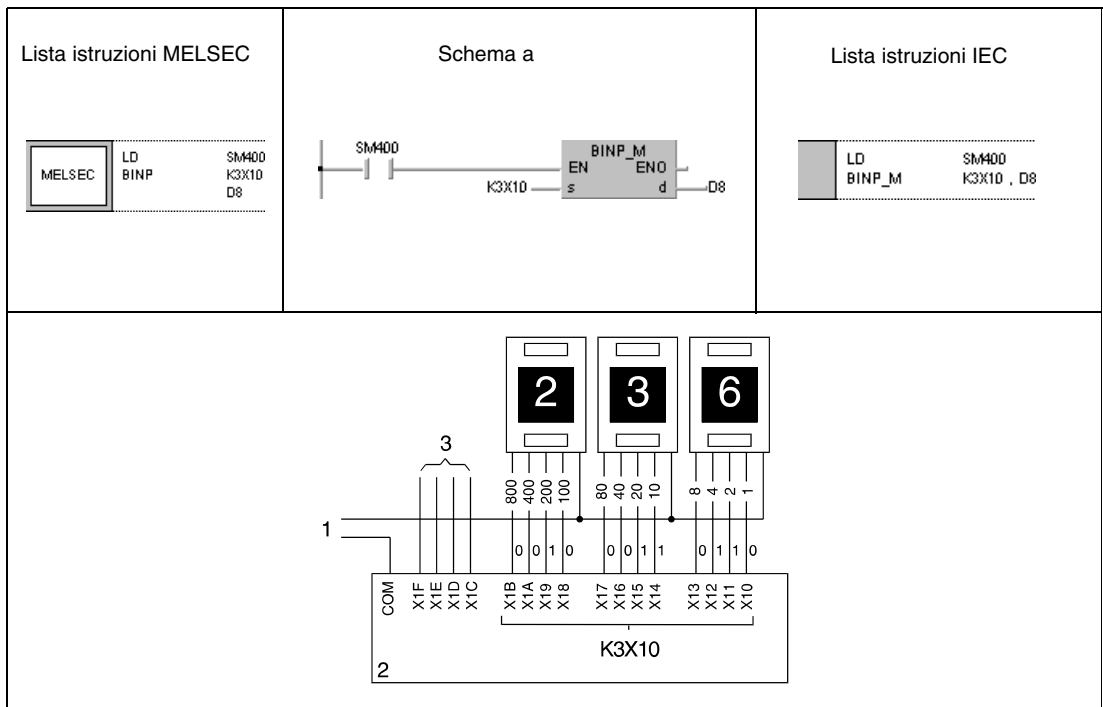
A causa dei ritardi di commutazione delle unità di visualizzazione BCD, si possono verificare errori nell'esecuzione del programma se si usano i relé speciali M9036 o M9037 come condizione di ingresso. In questo caso il dato BCD deve essere prima impostato con un normale dispositivo di ingresso e poi convertito (solo serie A).



Programma di esempio 1

BINP

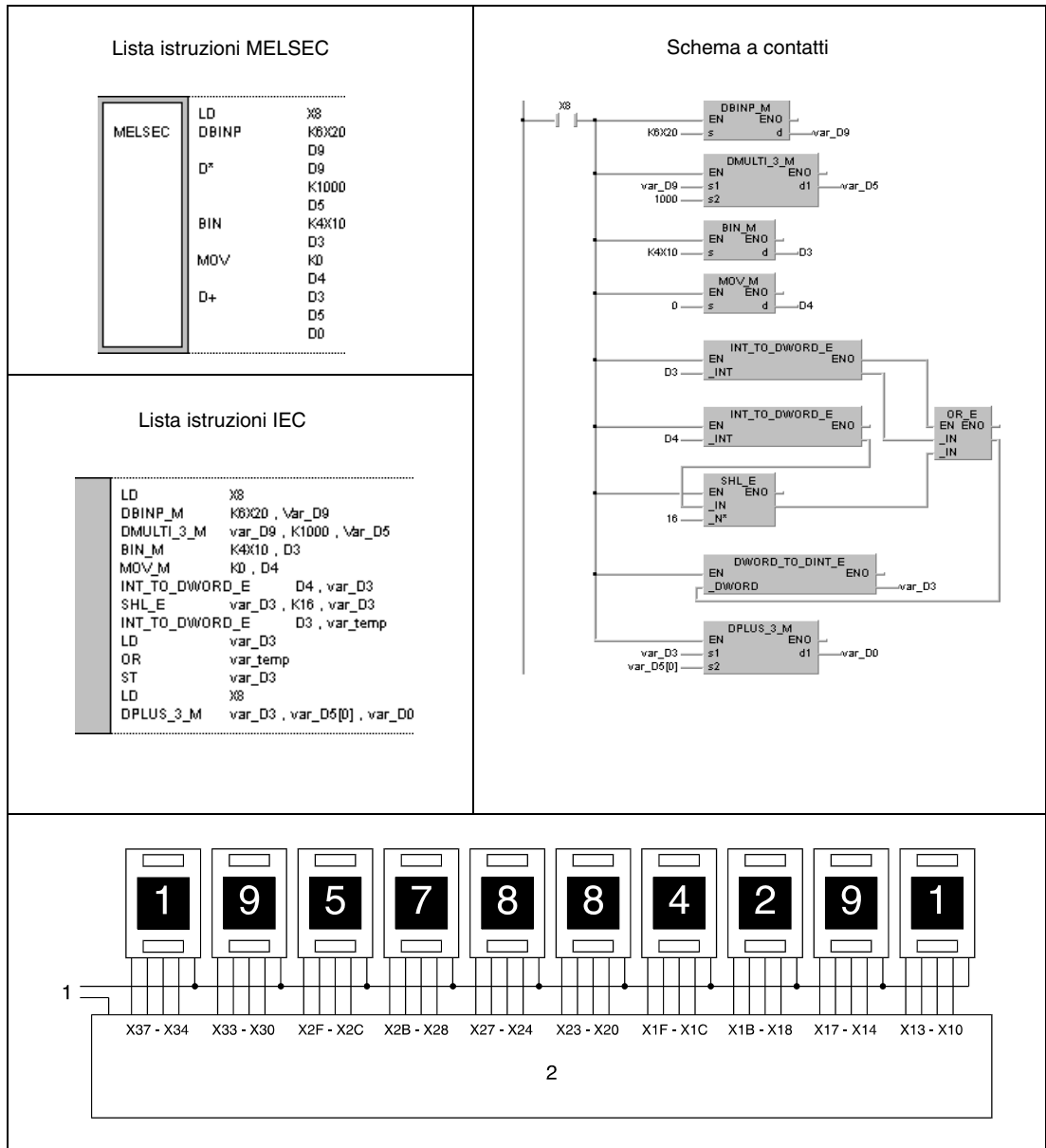
Il programma seguente converte il dato BCD da X10 a X1B in dato BIN sul fronte di salita di SM400. Il risultato è memorizzato in D8.



- 1 Alimentazione ingressi
- 2 Modulo ingressi
- 3 Ingressi disponibili

Programma di esempio 2 DBINP

Il programma seguente converte il dato BCD da X10 a X37 in dato BIN, con il fronte di salita di X8. Il risultato viene memorizzato in D0 e D1.



- ¹ Alimentazione ingressi
- ² Modulo ingressi

NOTA

Il dato BCD contenuto da X10 a X37, eccedente il campo ammesso di 2147483647, non può essere elaborato da operandi a 32 bit! In questo caso il valore contenuto in D0 e D1 diventa negativo. Per ulteriori dettagli, vedere capitolo "Elaborazione dati numerici" nel manuale di programmazione.

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.3.3 FLT, FLTP, DFLT, DFLTP

CPU

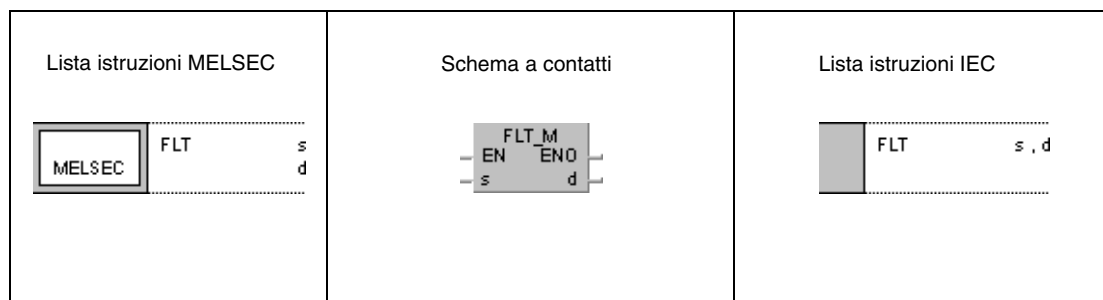
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

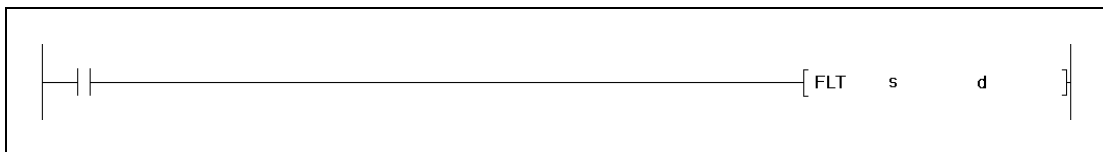
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	●	—	—	3	
d	—	●	●	—	●	—	—	—	—		

GX IEC Developer



GX Developer



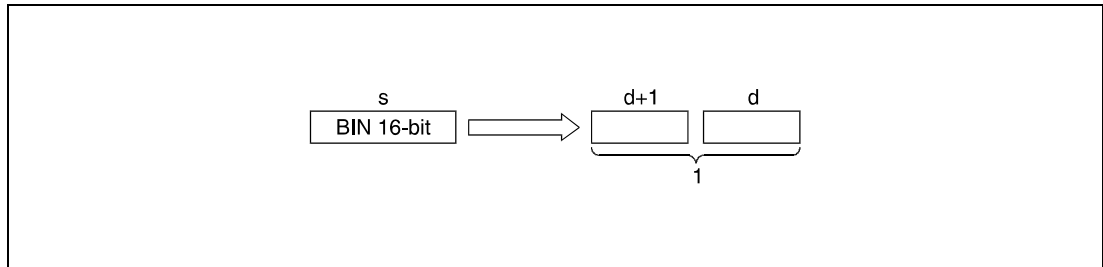
Variabili

Dati impostati	Significato	Tipo dati
s	Dato BIN o primo numero dell'operando contenente il dato BIN	BIN 16/32-bit
d	Primo numero dell'operando che memorizza il dato in virgola mobile.	Numero reale

Funzioni Conversione da dati BIN a dati in virgola mobile

FLT Conversione da dati BIN 16 bit a dati in virgola mobile

Il dato BIN a 16 bit in s viene convertito in dato in virgola mobile. Il risultato è memorizzato in d.

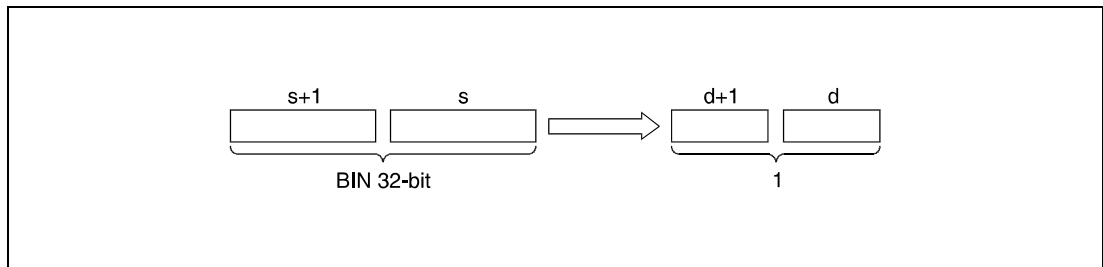


¹ Dati in virgola mobile, tipo dati per numeri reali

Il dato BIN a 16 bit indicato da s deve essere nel campo da -32768 a 32767.

DFLT Conversione da dati BIN 32 bit a dati in virgola mobile

Il dato BIN a 32 bit in s viene convertito in dato in virgola mobile. Il risultato è memorizzato in d.

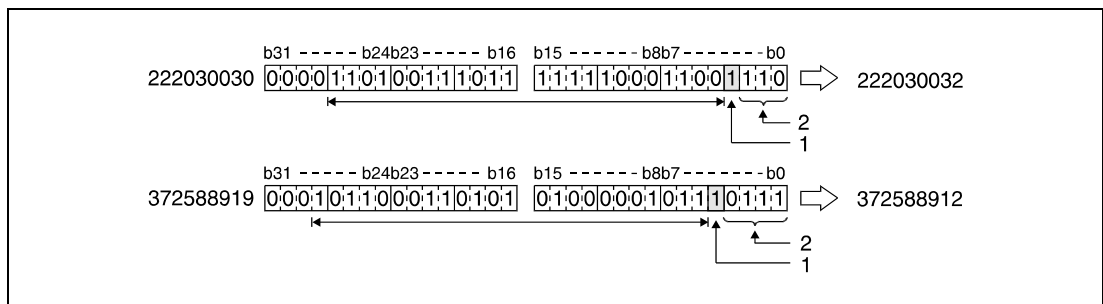


¹ Dati in virgola mobile, tipo dati per numeri reali

I dati BIN a 32 bit indicati da s e s+1 devono essere nel campo da -2147483648 a 2147483647.

A causa della elaborazione con semplici procedure a 32 bit dei dati in virgola mobile (tipo dati dei numeri reali), il numero di bit significativi è 24 per una rappresentazione binaria, o circa 7 cifre per una rappresentazione decimale.

Il risultato della conversione viene arrotondato al venticinquesimo bit. Tutti i bit più pesanti non sono considerati. Per questa ragione, se l'intero risultante supera il campo da -16777216 a 16777215 (valore BIN a 24 bit), la conversione può risultare errata.



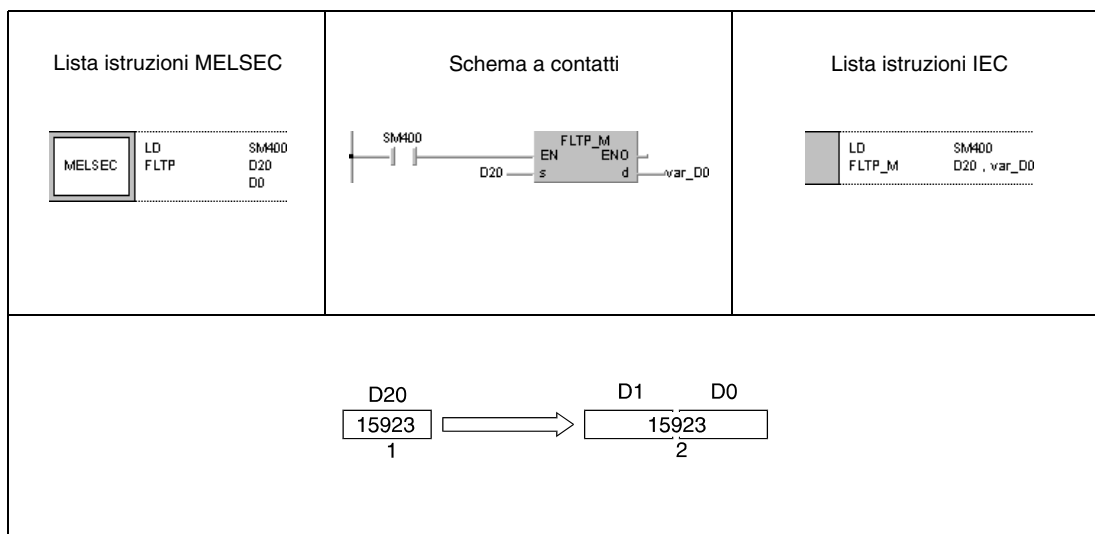
¹ Arrotondato

² Eliminato

Programma di esempio 1

FLTP

Sul fronte di salita di SM400, il programma che segue converte il dato BIN a 16 bit in D20 in un dato in virgola mobile. Il risultato viene memorizzato in D0 e D1.

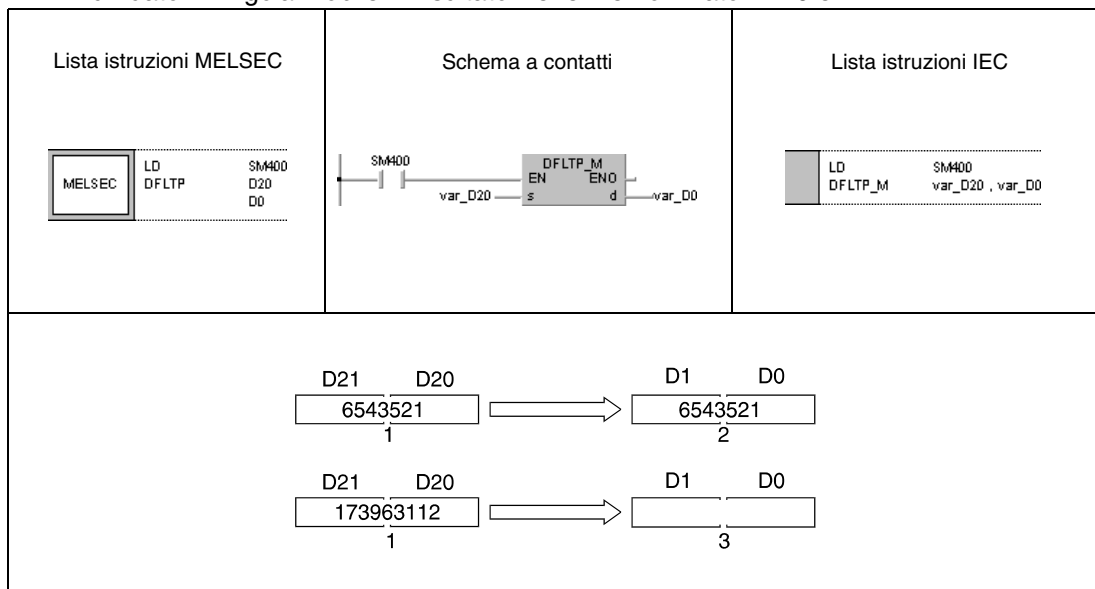


- ¹ Confronto dati
- ² Dati in virgola mobile, tipo dati per numeri reali

Programma di esempio 2

DFLTP

Sul fronte di salita di SM400, il programma che segue converte il dato BIN a 32 bit in D20 e D21 in un dato in virgola mobile. Il risultato viene memorizzato in D0 e D1.



- ¹ Confronto dati
- ² Dati in virgola mobile, tipo dati per numeri reali
- ³ Errore di conversione, perché esistono 7 cifre significative

NOTA

Questi programmi non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provocano messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.3.4 INT, INTP, DINT, DINTP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

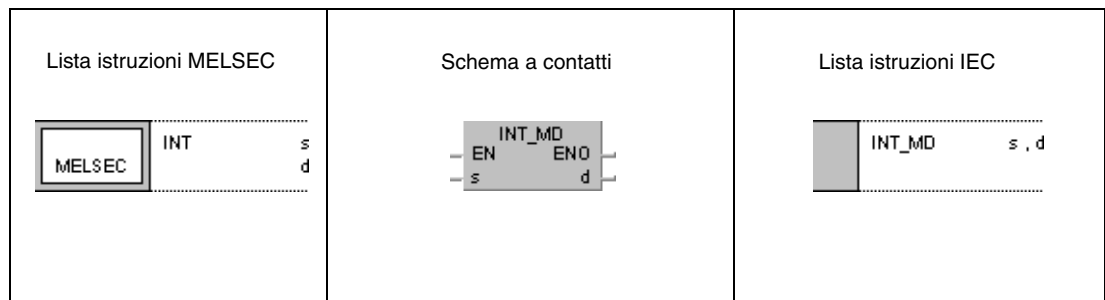
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

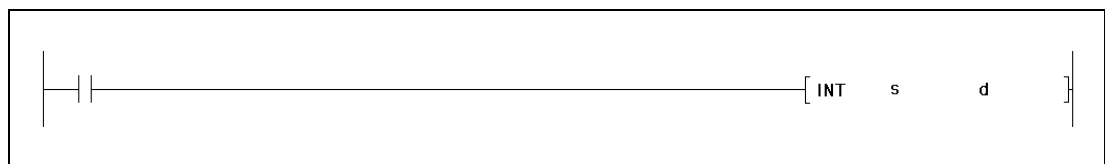
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti E			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	●	●	—	●	—	SM0	3
d	●	●	●	●	●	●	●	—	—		

GX IEC Developer



GX Developer

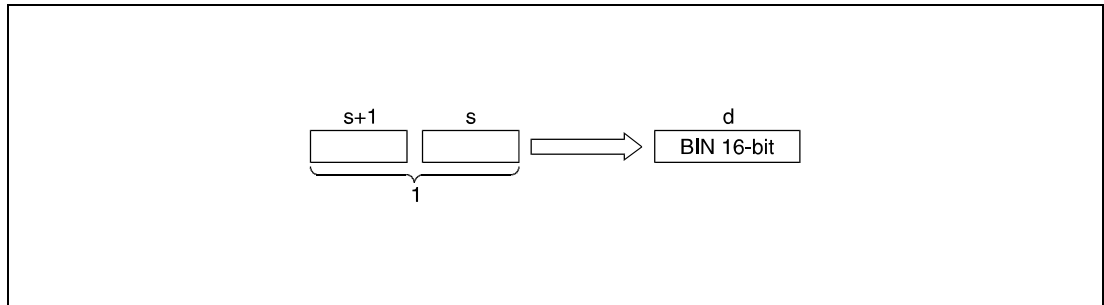


Variabili

Dati impostati	Significato	Tipo dati
s	Dato virgola mobile o primo numero dell'operando contenente il dato in virgola mobile	Numero reale
d	Primo numero dell'operando che memorizza il dato BIN.	BIN 16/32-bit

Funzioni **Conversione da dati in virgola mobile a dati BIN****INT** **Conversione da dati in virgola mobile a dati BIN 16 bit**

Il dato in virgola mobile in s viene convertito in dato BIN a 16 bit. Il risultato è memorizzato in d.



¹ Dati in virgola mobile, tipo dati per numeri reali

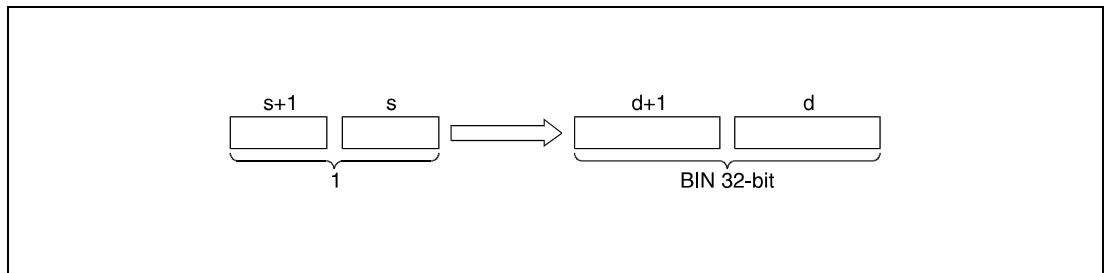
Il dato in virgola mobile in s e s+1 deve essere nel campo da -32768 a 32767.

Il valore intero convertito viene memorizzato come dato BIN 16 bit.

Il valore intero convertito viene arrotondato alla prima cifra dopo la virgola.

DINT **Conversione da dati in virgola mobile a dati BIN 32 bit**

Il dato in virgola mobile in s viene convertito in dato BIN a 32 bit. Il risultato è memorizzato in d.



¹ Dati in virgola mobile, tipo dati per numeri reali

Il dato in virgola mobile in s e s+1 deve essere nel campo da -2147483648 a 2147483647.

Il valore intero convertito viene memorizzato come dato BIN 32 bit.

Il valore intero convertito viene arrotondato alla prima cifra dopo la virgola.

Errori di esecuzione

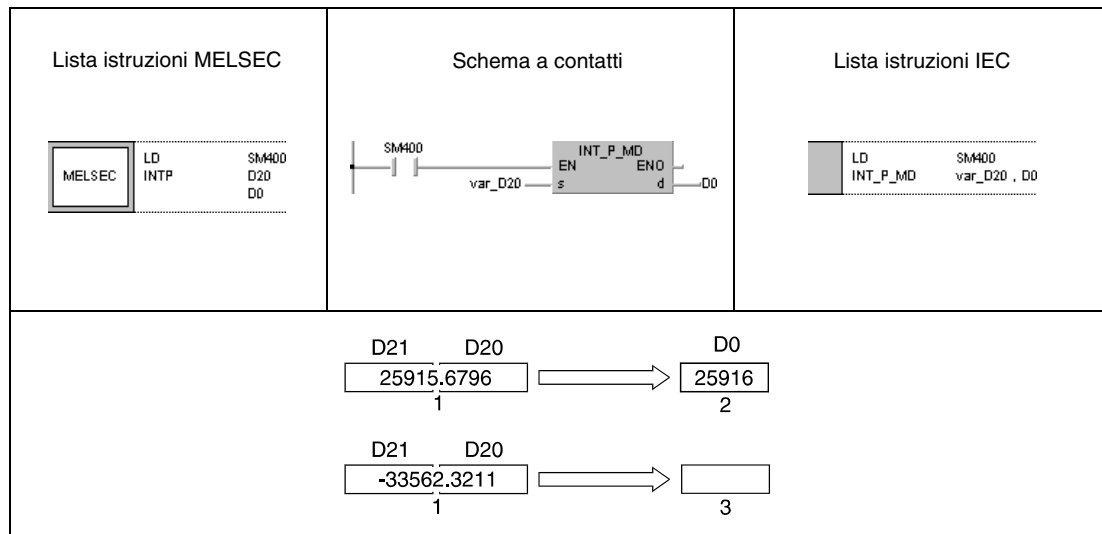
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il dato in virgola mobile indicato da s per l'esecuzione di una istruzione INT supera il campo ammesso da -32768 a 32767.
- Il dato in virgola mobile indicato da s per l'esecuzione di una istruzione DINT supera il campo ammesso da -2147483648 a 2147483647.

Programma di esempio 1

INTP

Sul fronte di salita di SM400, il programma che segue converte il dato in virgola mobile in D20 e D21 in un dato BIN a 16 bit. Il risultato è memorizzato in D0.



¹ Dati in virgola mobile, tipo dati per numeri reali

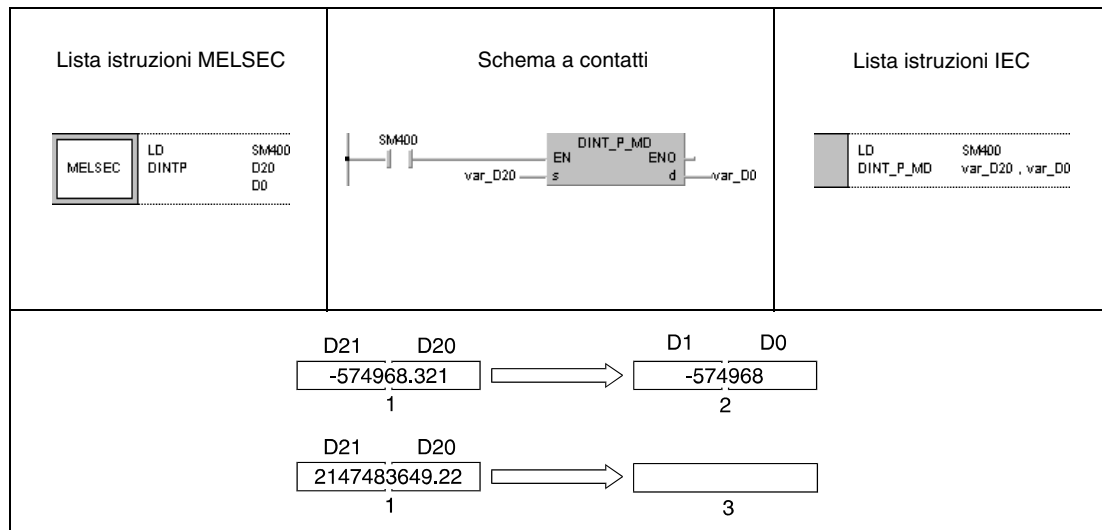
² Confronto dati

³ Nessun risultato. Il valore supera il campo ammesso per una istruzione INT. Viene restituito un codice di errore.

Programma di esempio 2

DINTP

Sul fronte di salita di SM400, il programma che segue converte il dato in virgola mobile in D20 e D21 in un dato BIN a 32 bit. Il risultato è memorizzato in D0.



¹ Dati in virgola mobile, tipo dati per numeri reali

² Confronto dati

³ Nessun risultato. Il valore supera il campo ammesso per una istruzione DINT. Viene restituito un codice di errore.

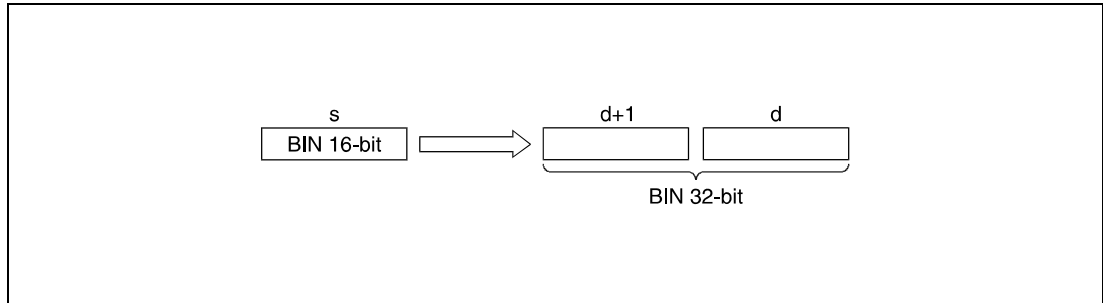
NOTA

Questi programmi non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provocano messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

Funzioni **Conversione da dati BIN a 16 bit a dati BIN a 32 bit**

DBL **Conversione da dati BIN a 16 bit a dati BIN a 32 bit**

Il dato BIN a 16 bit in s viene convertito nel dato BIN a 32 bit con segno. Il risultato è memorizzato in d.



Esempio di programma

DBLP

Il programma seguente converte il dato BIN a 16 bit in D100 in un dato BIN a 32 bit sul fronte di salita di X20. Il risultato viene memorizzato in R0 e R1.

<p style="text-align: center;">Lista istruzioni MELSEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">MELSEC</td> <td style="padding: 2px;">LD</td> <td style="padding: 2px;">X20</td> </tr> <tr> <td style="padding: 2px;"></td> <td style="padding: 2px;">DBLP</td> <td style="padding: 2px;">D100 R0</td> </tr> </table>	MELSEC	LD	X20		DBLP	D100 R0	<p style="text-align: center;">Schema a contatti</p>	<p style="text-align: center;">Lista istruzioni IEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">LD</td> <td style="padding: 2px;">X20</td> </tr> <tr> <td style="padding: 2px;">DBLP_M</td> <td style="padding: 2px;">D100 , var_R0</td> </tr> </table>	LD	X20	DBLP_M	D100 , var_R0
MELSEC	LD	X20										
	DBLP	D100 R0										
LD	X20											
DBLP_M	D100 , var_R0											

NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

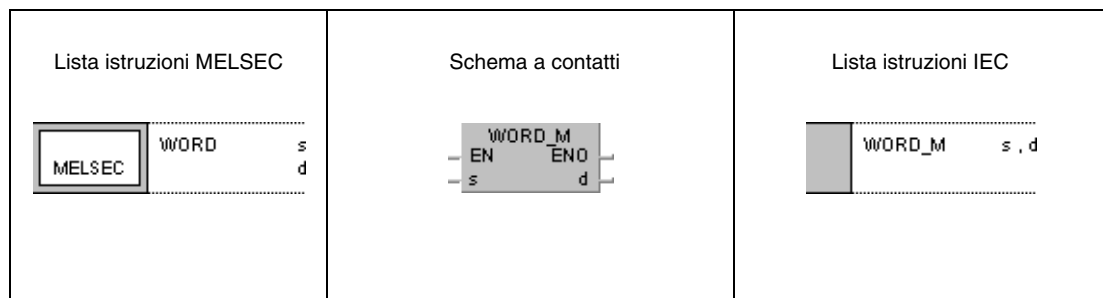
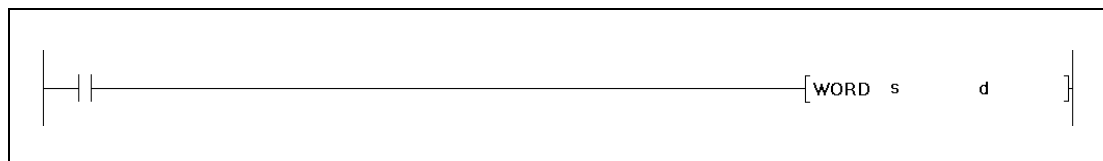
6.3.6 WORD, WORDP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi
MELSEC Q

	Operandi utilizzabili									Segnale di errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	●	●	—	SM0	3
d	●	●	●	●	●	●	—	—	—		

GX IEC
DeveloperGX
Developer

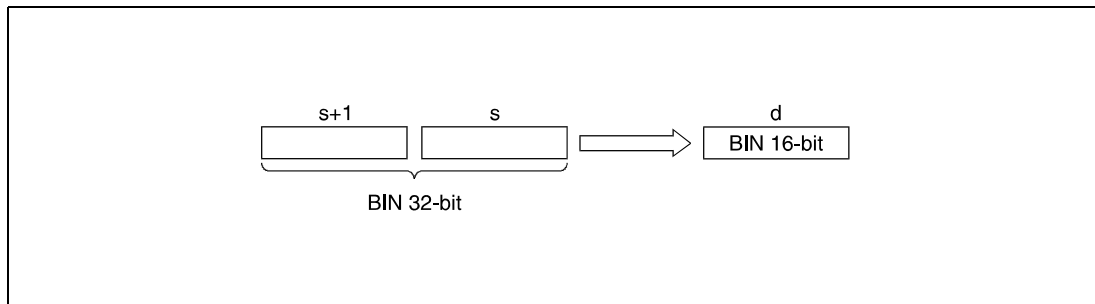
Variabili

Dati impostati	Significato	Tipo dati
s	Primo numero dell'operando che contiene i dati da convertire	BIN 32-bit
d	Primo numero dell'operando che memorizza il dato convertito.	BIN 16-bit

Funzioni Conversione da dati BIN a 32 bit a dati BIN a 16 bit

WORD Conversione da dati BIN a 32 bit a dati BIN a 16 bit

Il dato BIN a 32 bit in s viene convertito nel dato BIN a 16 bit con segno. Il risultato è memorizzato in d.



Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il dato BIN indicato da s e s+1 supera il campo valori ammesso da -32768 a 32767 (codice di errore: 4100).

Esempio di programma

WORDP

Il programma seguente converte il dato BIN a 32 bit in D100 e D101 in un dato BIN a 16 bit sul fronte di salita di X20. Il risultato è memorizzato in R0.

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC

NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

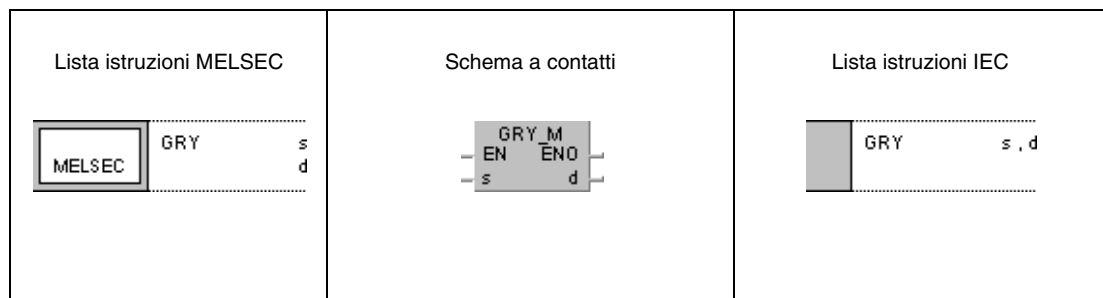
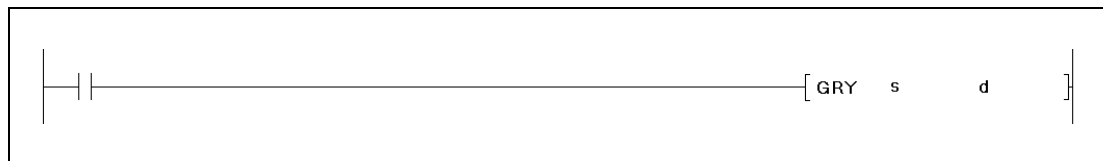
6.3.7 GRY, GRYP, DGRY, DGRYP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi
MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	●	—	SM0	3	
d	●	●	●	●	●	●	—	—			

GX IEC
DeveloperGX
Developer

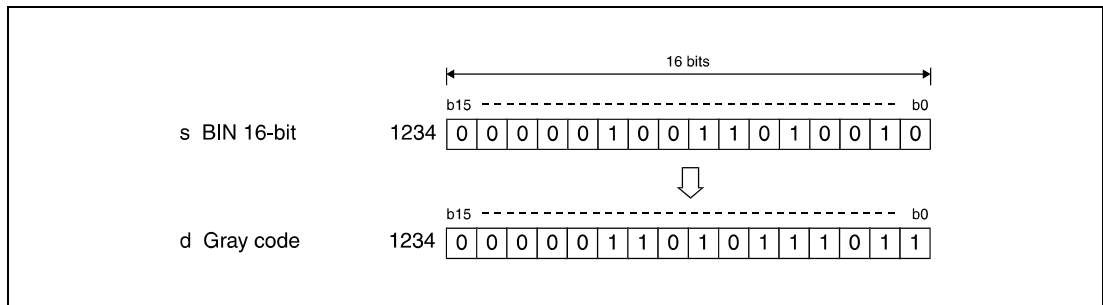
Variabili

Dati impostati	Significato	Tipo dati
s	Dato BIN o primo numero dell'operando contenente il dato BIN	BIN 16/32-bit
d	Primo numero dell'operando che memorizza il dato convertito in codice Gray.	Dati codice GRAY 16-/32-bit

Funzioni Conversione da dati BIN a dati in codice Gray

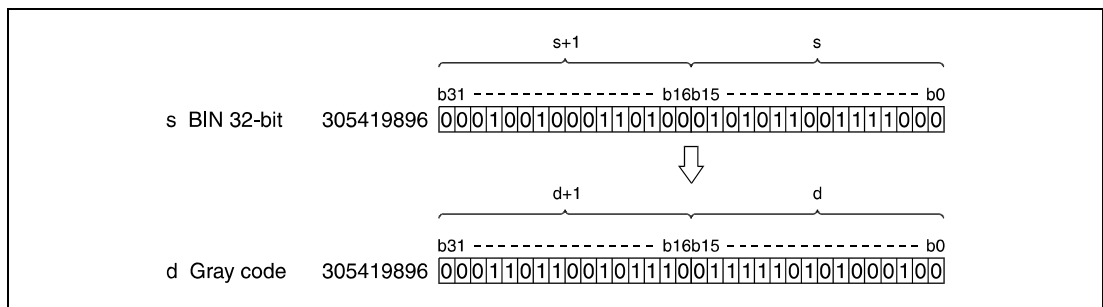
GRY Conversione da dati BIN a 16 bit a dati in codice Gray

Il dato BIN a 16 bit in s viene convertito in dato in codice Gray. Il risultato è memorizzato in d.



DGRY Conversione da dati BIN a 32 bit a dati in codice Gray

Il dato BIN a 32 bit in s viene convertito in dato in codice Gray. Il risultato è memorizzato in d.



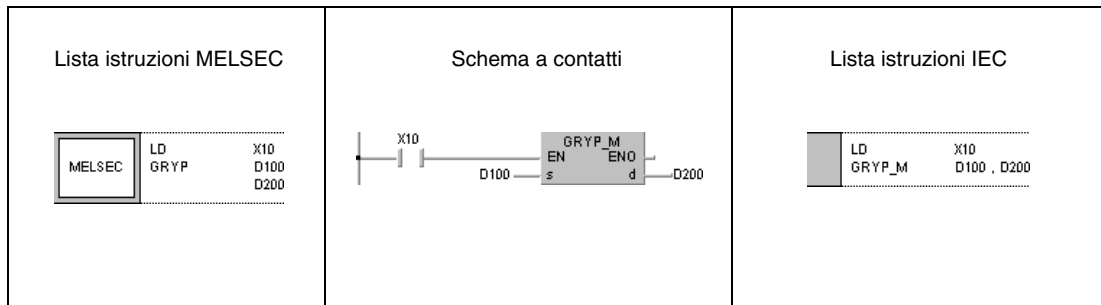
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

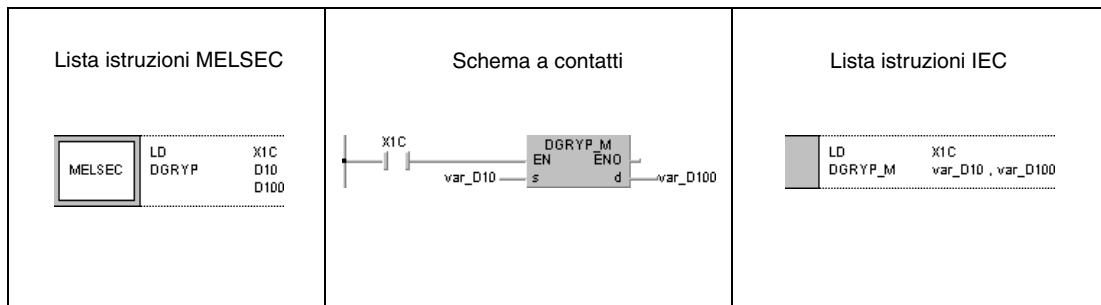
- Il dato in s è negativo.

Programma di esempio 1**GRYP**

Il programma seguente converte il dato BIN a 16 bit in D100 in un dato in codice Gray sul fronte di salita di X10. Il risultato è memorizzato in D200.

**Programma di esempio 2****DGRYP**

Il programma seguente converte il dato BIN a 32 bit in D10 e D11 in un dato in codice Gray sul fronte di salita di X1C. Il risultato viene memorizzato in D100 e D101.

**NOTA**

Il programma di esempio 2 non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.3.8 GBIN, GBINP, DGBIN, DGBINP

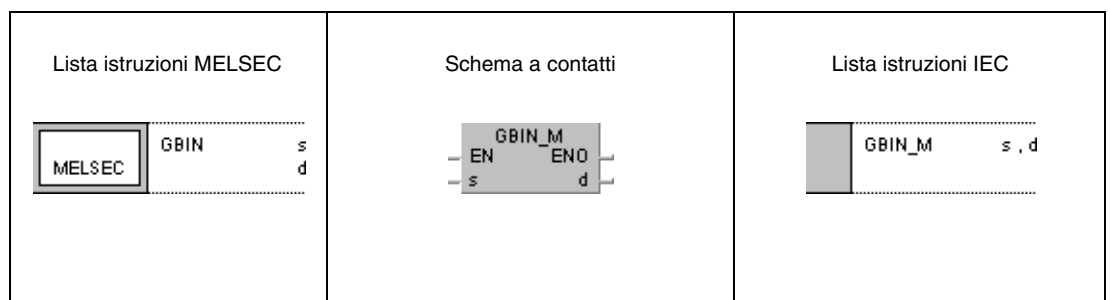
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

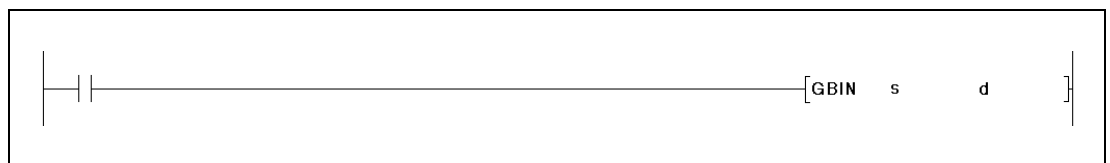
Operandi MELSEC Q

	Operandi utilizzabili									Segnale di errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	●	●	—	SM0	3
d	●	●	●	●	●	●	—	—	—		

GX IEC Developer



GX Developer



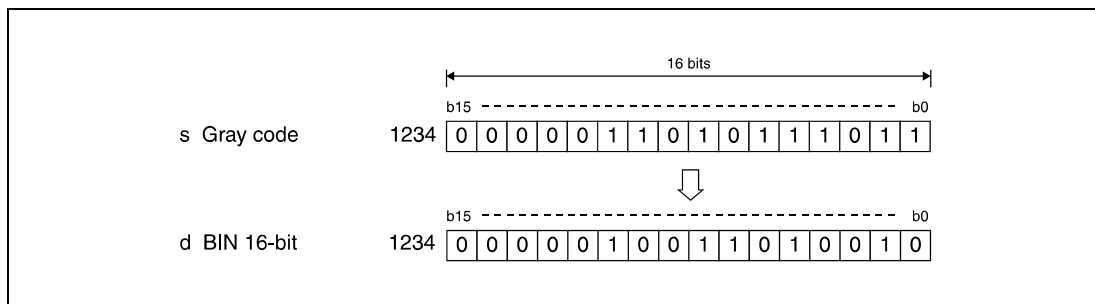
Variabili

Dati impostati	Significato	Tipo dati
s	Dato in codice Gray o primo numero dell'operando contenente il codice Gray.	Dati codice GRAY 16-/32-bit
d	Primo numero dell'operando che memorizza il dato BIN convertito.	BIN 16/32-bit

Funzioni Conversione da dati in codice Gray a dati BIN

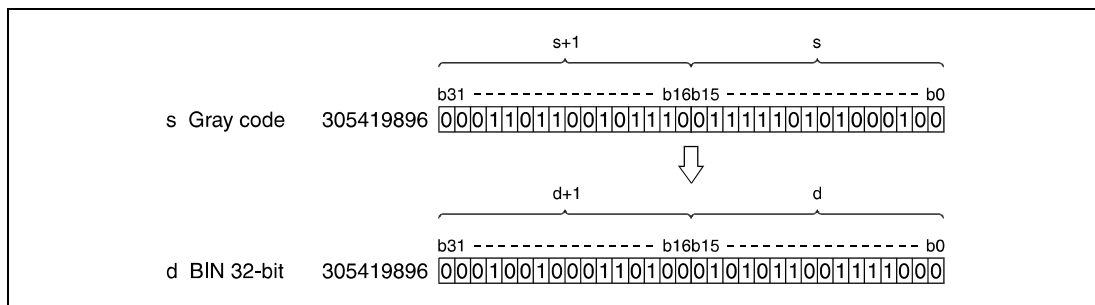
GBIN Conversione da dati in codice Gray a dati BIN a 16 bit

Il dato in codice Gray in s viene convertito in dato BIN a 16 bit. Il risultato è memorizzato in d.



DGBIN Conversione da dati in codice Gray a dati BIN a 32 bit

Il dato in codice Gray in s viene convertito in dato BIN a 32 bit. Il risultato è memorizzato in d.



Errori di esecuzione

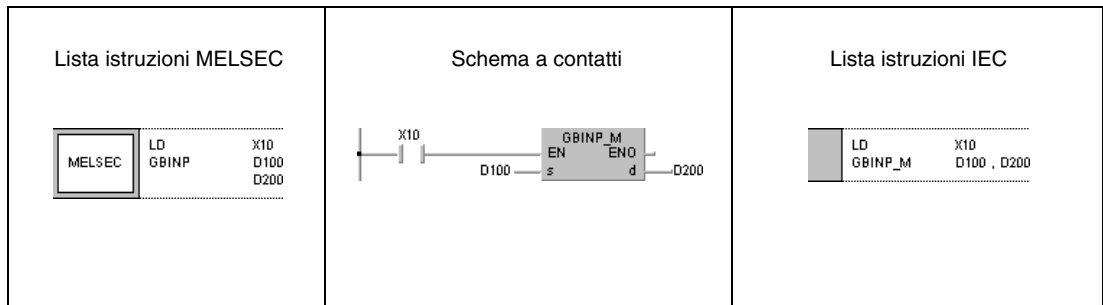
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il dato in s supera il campo valori ammesso da 0 a 32767 durante una istruzione GBIN.
- Il dato in s supera il campo valori ammesso da 0 a 2147483647 durante una istruzione DGBIN.

Programma di esempio 1

GBINP

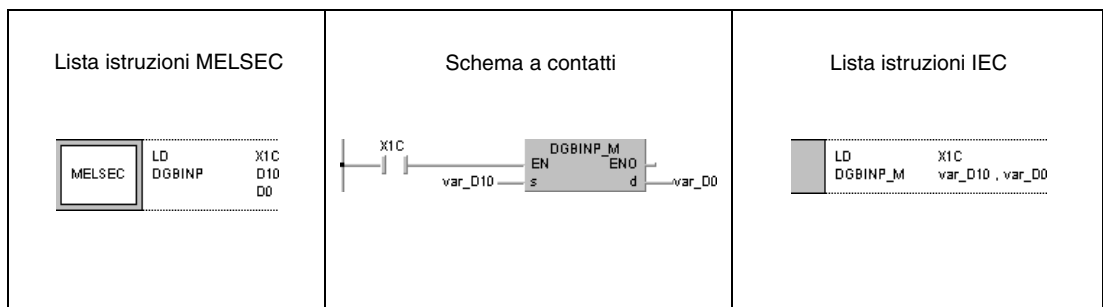
Il programma seguente converte il dato in codice Gray in D100 in un dato BIN a 16-bit sul fronte di salita di X10. Il risultato è memorizzato in D200.



Programma di esempio 2

DGBINP

Il programma seguente converte il dato in codice Gray in D10 e D11 in un dato BIN a 32 bit sul fronte di salita di X1C. Il risultato viene memorizzato in D0 e D1.



NOTA

Il programma di esempio 2 non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.3.9 NEG, NEGP, DNEG, DNEGP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

	Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Riporto	Errore							
	Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello												
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1							Z	V	K	H (16#)	P	I	N
d	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	● ²				K1 ↓ K4	3 ↓ 1	●		M9012	M9010 M9011	●

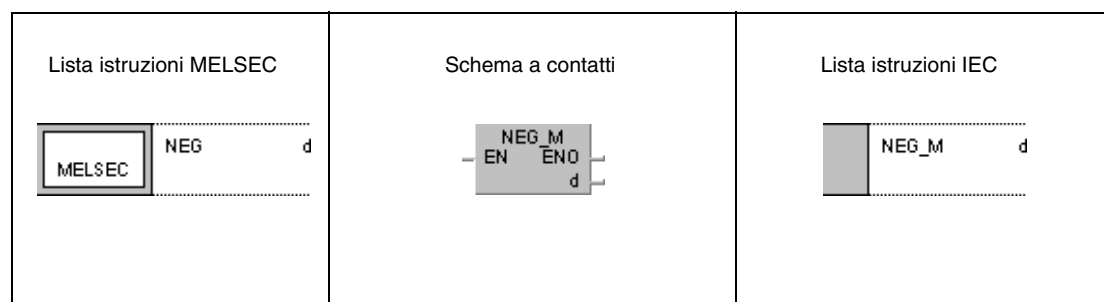
¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

² DNEG e DNEGP non sono validi per CPU AnN e AnS

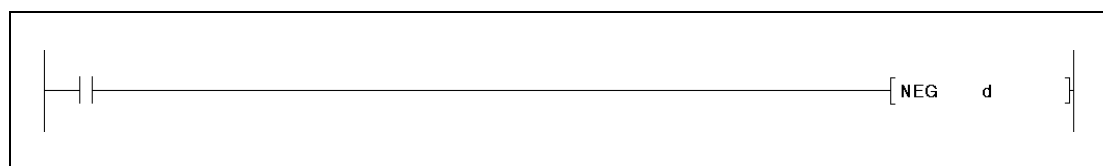
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
d	●	●	●	●	●	●	●	—	—	—	2

GX IEC Developer



GX Developer



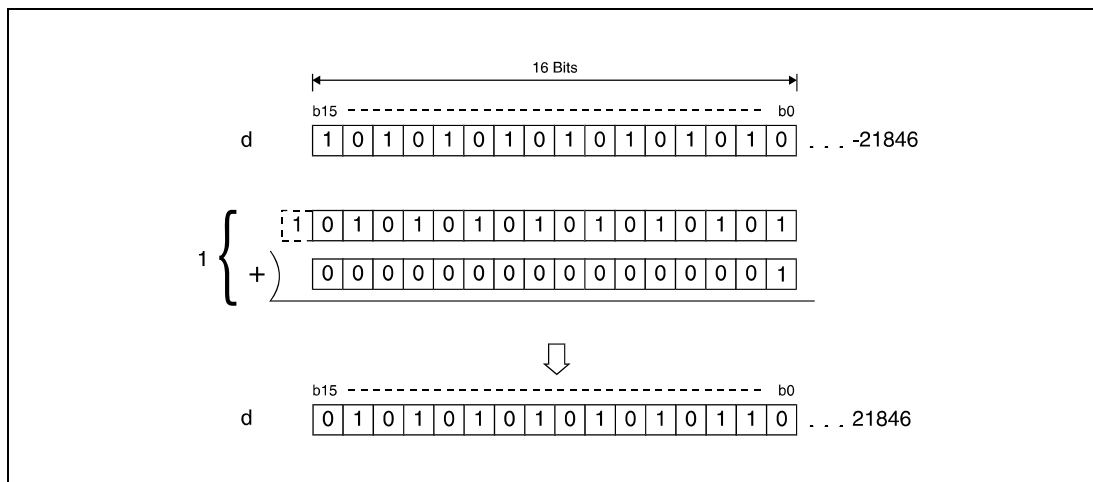
Variabili

Dati impostati	Significato	Tipo dati
d	Primo numero dell'operando che contiene i dati di cui si deve invertire il segno.	BIN 16/32-bit

Funzioni Inversione segno per dati BIN (complemento a 2)

NEG Negazione di dati BIN 16-bit

L'istruzione NEG (complemento a 2 / operazione NOT) inverte il segno dei dati BIN a 16 bit. Il dato BIN a 16 bit in d viene prima invertito, e poi viene sommato il valore 1. Il risultato è memorizzato in d.

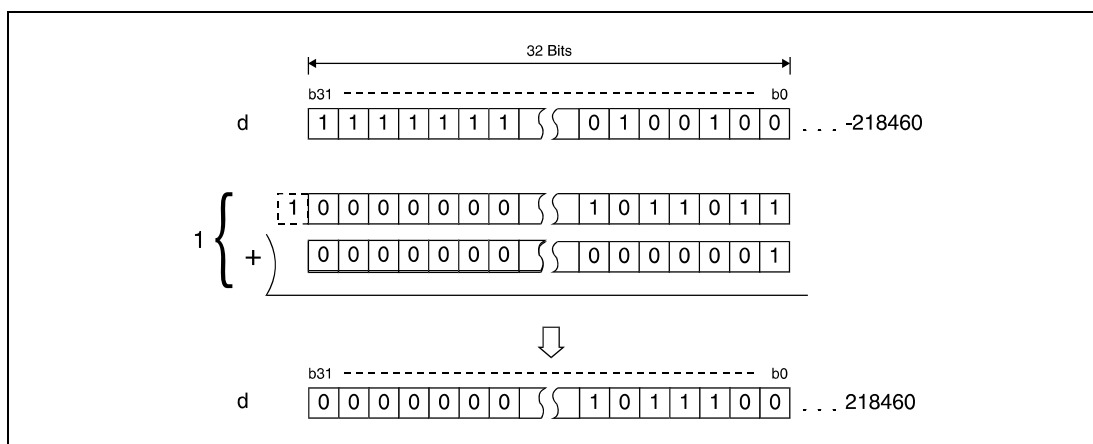


¹ Inversione con somma successiva

La funzione di questa istruzione è di cambiare un segno da negativo a positivo o viceversa.

DNEG Negazione di dati BIN 32-bit (solo serie Q e System Q)

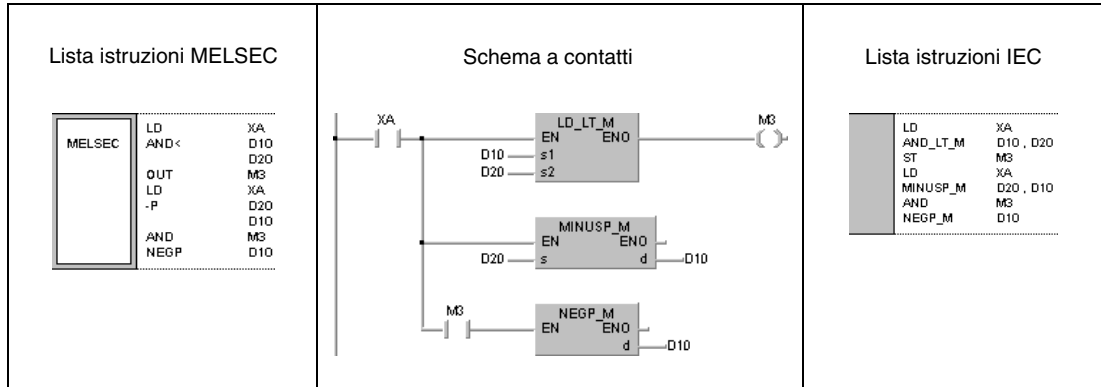
L'istruzione DNEG (complemento a 2 / operazione NOT) inverte il segno dei dati BIN a 32 bit. Il dato BIN a 32 bit in d viene prima invertito, e poi viene sommato il valore 1. Il risultato è memorizzato in d.



¹ Inversione con somma successiva

Esempio di programma NEGP

Il programma che segue sottrae il dato in D10 da quello in D20 sul fronte di salita di XA. M3 viene impostato a 1 se D10 è inferiore a D20. Se M3 = 1, il risultato in D10 è il valore assoluto (complemento a 2) e diviene positivo.



6.3.10 ENEG, ENEGP

CPU

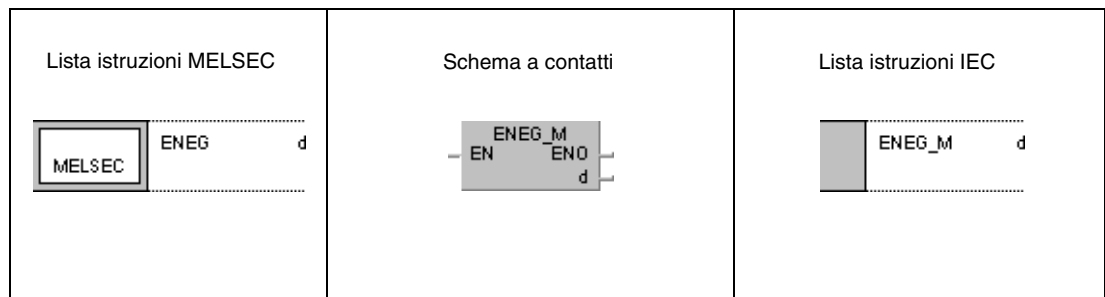
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

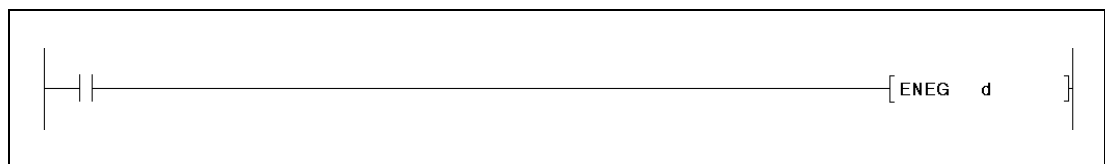
Operandi MELSEC Q

	Operandi utilizzabili									Segnale di errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
d	—	●	●	—	●	●	—	—	—	—	2

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
d	Primo numero dell'operando che contiene il dato in virgola mobile di cui si deve invertire il segno.	Numero reale

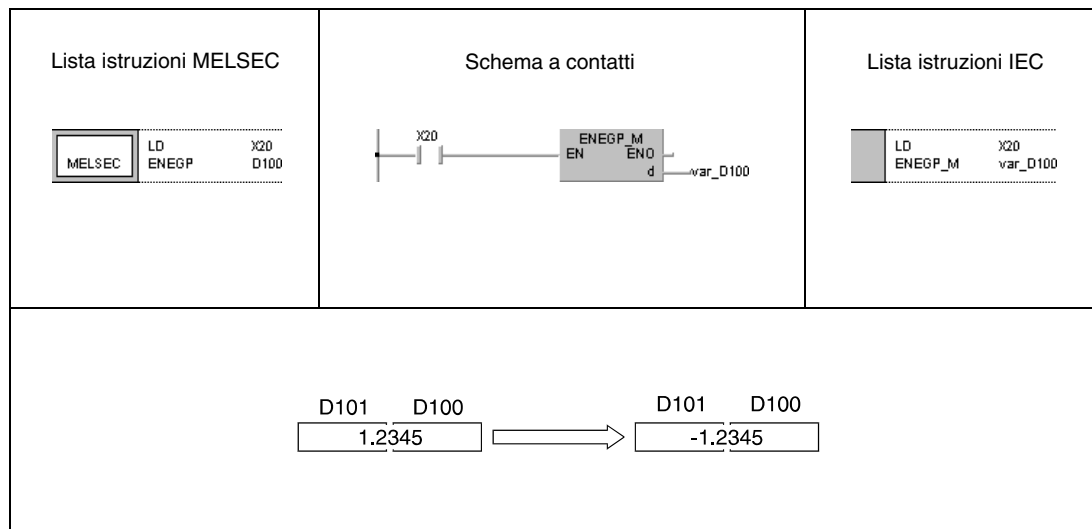
Funzioni **Inversione di segno per dati in virgola mobile****ENEG** **Negazione di dati in virgola mobile**

Queste istruzioni cambiano il segno del dato in virgola mobile in d. Il risultato rimane in d.

La funzione di queste istruzioni è di cambiare un segno da negativo a positivo o viceversa.

Esempio di programma ENEGP

Il programma seguente nega il dato virgola mobile in D100 e D101 sul fronte positivo di X20. Il risultato viene memorizzato in D100 e D101.

**NOTA**

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.3.11 BKBCD, BKBCDP


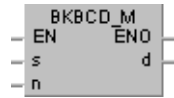
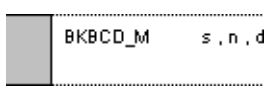
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

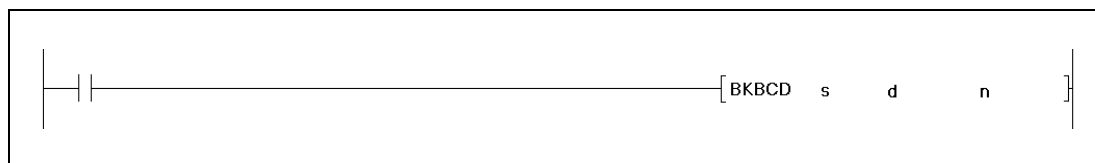
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	—	SMO	4	
d	—	●	●	—	—	—	—	—			
n	●	●	●	●	●	●	●	●			

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
--	--	---

GX Developer



Variabili

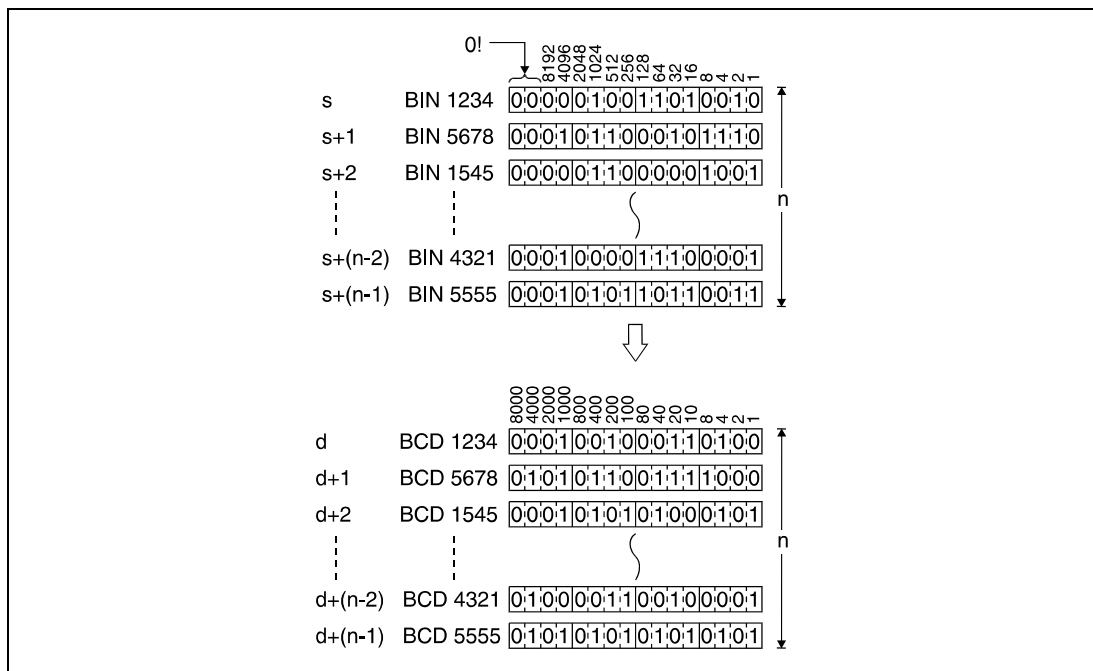
Dati impostati	Significato	Tipo dati
s	Primo numero dell'operando che contiene il dato BIN da convertire	BIN 16-bit
d	Primo numero dell'operando che memorizza il dato BCD convertito.	BCD 4-cifre
n	Numero di blocchi dati da convertire.	BIN 16-bit

Funzioni **Conversione da blocchi dati BIN a blocchi dati BCD****BKBCD Conversione da blocchi dati BIN a 16 bit a blocchi dati BCD a 4 cifre**

Questa istruzione converte ciascun ennesimo blocco dati BIN a 16 bit in s nell'ennesimo blocco dati BCD a 4 cifre I dati convertiti sono memorizzati in d.

Il blocco dati BIN a 16 bit indicato da s deve essere nel campo da 0 a 9999.

I due bit più significativi del dato BIN a 16 bit in s devono essere azzerati.

**Errori di esecuzione**

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il numero di blocchi dati indicato da n supera i numeri dispositivo indicati da s e d (codice di errore: 4101).
- Il dato del blocco dati BIN supera il campo valori ammesso da 0 a 9999 (codice di errore: 4100).
- I numeri degli operandi indicati da s e d sono sovrapposti (codice di errore 4101).

Per dettagli sulla indicizzazione, consultare il capitolo 3.6.

Esempio di programma

BKBCDP

Il programma seguente converte il dato del blocco BIN a 16 bit in D100 in un blocco dati BCD a 4 cifre sul fronte di salita di X20. I dati convertiti sono memorizzati in D200. Il numero di blocchi convertiti (3) è contenuto in D0

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC																																																																																																																																																																					
<div style="border: 1px solid black; padding: 5px; display: inline-block;"> <table border="0" style="border-collapse: collapse;"> <tr> <td style="border: 1px solid black; padding: 2px;">MELSEC</td> <td style="padding: 2px;">LD</td> <td style="padding: 2px;">X20</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;"></td> <td style="padding: 2px;">BKBCDP</td> <td style="padding: 2px;">D100 D200 D0</td> </tr> </table> </div>	MELSEC	LD	X20		BKBCDP	D100 D200 D0		<table border="0" style="border-collapse: collapse;"> <tr> <td style="border: 1px solid black; padding: 2px;">LD</td> <td style="padding: 2px;">X20</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">BKBCDP_M</td> <td style="padding: 2px;">D100 , D0 , D200</td> </tr> </table>	LD	X20	BKBCDP_M	D100 , D0 , D200																																																																																																																																																											
MELSEC	LD	X20																																																																																																																																																																					
	BKBCDP	D100 D200 D0																																																																																																																																																																					
LD	X20																																																																																																																																																																						
BKBCDP_M	D100 , D0 , D200																																																																																																																																																																						
<table style="margin: auto;"> <tr> <td style="padding-right: 10px;">D100</td> <td style="padding-right: 10px;">BIN 5432</td> <td style="text-align: right;"> <table style="border-collapse: collapse;"> <tr><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td></tr> <tr><td style="text-align: right;">8192</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">4096</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">2048</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">1024</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">512</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">256</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">128</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">64</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">32</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">16</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">8</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">4</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">2</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">1</td></tr> </table> </td> </tr> <tr> <td style="padding-right: 10px;">D101</td> <td style="padding-right: 10px;">BIN 4444</td> <td style="text-align: right;"> <table style="border-collapse: collapse;"> <tr><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td></tr> </table> </td> </tr> <tr> <td style="padding-right: 10px;">D102</td> <td style="padding-right: 10px;">BIN 3210</td> <td style="text-align: right;"> <table style="border-collapse: collapse;"> <tr><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td></tr> </table> </td> </tr> <tr> <td colspan="2"></td> <td style="text-align: center; padding: 10px 0;"> </td> </tr> <tr> <td style="padding-right: 10px;">D200</td> <td style="padding-right: 10px;">BCD 5432</td> <td style="text-align: right;"> <table style="border-collapse: collapse;"> <tr><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td></tr> <tr><td style="text-align: right;">8000</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">4000</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">2000</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">1000</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">500</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">250</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">125</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">62</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">31</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">15</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">7</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">3</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">1</td></tr> </table> </td> </tr> <tr> <td style="padding-right: 10px;">D201</td> <td style="padding-right: 10px;">BCD 4444</td> <td style="text-align: right;"> <table style="border-collapse: collapse;"> <tr><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td></tr> </table> </td> </tr> <tr> <td style="padding-right: 10px;">D202</td> <td style="padding-right: 10px;">BCD 3210</td> <td style="text-align: right;"> <table style="border-collapse: collapse;"> <tr><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td></tr> </table> </td> </tr> </table>			D100	BIN 5432	<table style="border-collapse: collapse;"> <tr><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td></tr> <tr><td style="text-align: right;">8192</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">4096</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">2048</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">1024</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">512</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">256</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">128</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">64</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">32</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">16</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">8</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">4</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">2</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">1</td></tr> </table>	0	0	1	0	1	0	1	0	1	1	1	0	0	0	8192		4096		2048		1024		512		256		128		64		32		16		8		4		2		1	D101	BIN 4444	<table style="border-collapse: collapse;"> <tr><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td></tr> </table>	0	0	1	0	0	0	1	0	1	1	1	1	0	0	0	D102	BIN 3210	<table style="border-collapse: collapse;"> <tr><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td></tr> </table>	0	0	0	1	1	0	0	0	1	0	1	0	1	0	0				D200	BCD 5432	<table style="border-collapse: collapse;"> <tr><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td></tr> <tr><td style="text-align: right;">8000</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">4000</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">2000</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">1000</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">500</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">250</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">125</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">62</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">31</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">15</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">7</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">3</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">1</td></tr> </table>	0	1	0	1	0	0	0	0	1	1	0	0	1	0	0	0	8000		4000		2000		1000		500		250		125		62		31		15		7		3		1	D201	BCD 4444	<table style="border-collapse: collapse;"> <tr><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td></tr> </table>	0	1	0	0	0	1	0	0	1	0	0	1	0	0	1	0	D202	BCD 3210	<table style="border-collapse: collapse;"> <tr><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td></tr> </table>	0	0	1	1	0	1	0	0	0	1	0	0	1	0	0	0
D100	BIN 5432	<table style="border-collapse: collapse;"> <tr><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td></tr> <tr><td style="text-align: right;">8192</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">4096</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">2048</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">1024</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">512</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">256</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">128</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">64</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">32</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">16</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">8</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">4</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">2</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">1</td></tr> </table>	0	0	1	0	1	0	1	0	1	1	1	0	0	0	8192		4096		2048		1024		512		256		128		64		32		16		8		4		2		1																																																																																																																												
0	0	1	0	1	0	1	0	1	1	1	0	0	0																																																																																																																																																										
8192		4096		2048		1024		512		256		128		64		32		16		8		4		2		1																																																																																																																																													
D101	BIN 4444	<table style="border-collapse: collapse;"> <tr><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td></tr> </table>	0	0	1	0	0	0	1	0	1	1	1	1	0	0	0																																																																																																																																																						
0	0	1	0	0	0	1	0	1	1	1	1	0	0	0																																																																																																																																																									
D102	BIN 3210	<table style="border-collapse: collapse;"> <tr><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td></tr> </table>	0	0	0	1	1	0	0	0	1	0	1	0	1	0	0																																																																																																																																																						
0	0	0	1	1	0	0	0	1	0	1	0	1	0	0																																																																																																																																																									
D200	BCD 5432	<table style="border-collapse: collapse;"> <tr><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td></tr> <tr><td style="text-align: right;">8000</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">4000</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">2000</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">1000</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">500</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">250</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">125</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">62</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">31</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">15</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">7</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">3</td><td style="border: 1px solid black; padding: 0 2px;"></td><td style="text-align: right;">1</td></tr> </table>	0	1	0	1	0	0	0	0	1	1	0	0	1	0	0	0	8000		4000		2000		1000		500		250		125		62		31		15		7		3		1																																																																																																																												
0	1	0	1	0	0	0	0	1	1	0	0	1	0	0	0																																																																																																																																																								
8000		4000		2000		1000		500		250		125		62		31		15		7		3		1																																																																																																																																															
D201	BCD 4444	<table style="border-collapse: collapse;"> <tr><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td></tr> </table>	0	1	0	0	0	1	0	0	1	0	0	1	0	0	1	0																																																																																																																																																					
0	1	0	0	0	1	0	0	1	0	0	1	0	0	1	0																																																																																																																																																								
D202	BCD 3210	<table style="border-collapse: collapse;"> <tr><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">1</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td><td style="border: 1px solid black; padding: 0 2px;">0</td></tr> </table>	0	0	1	1	0	1	0	0	0	1	0	0	1	0	0	0																																																																																																																																																					
0	0	1	1	0	1	0	0	0	1	0	0	1	0	0	0																																																																																																																																																								
<table style="margin: auto;"> <tr> <td style="padding-right: 10px;">D0</td> <td style="border: 1px solid black; padding: 2px 10px;">3</td> </tr> </table>			D0	3																																																																																																																																																																			
D0	3																																																																																																																																																																						

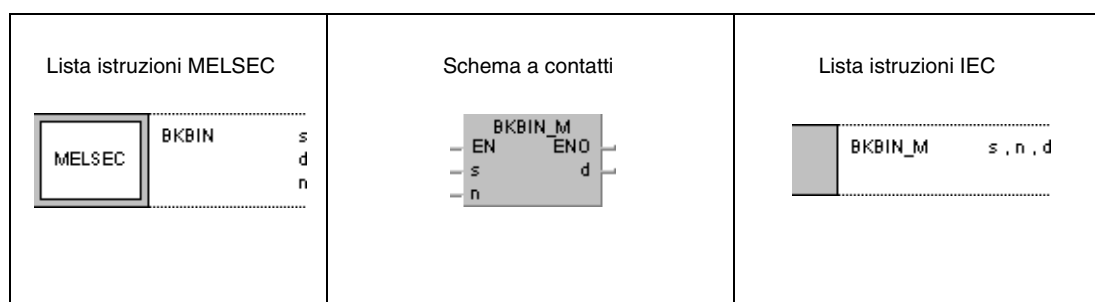
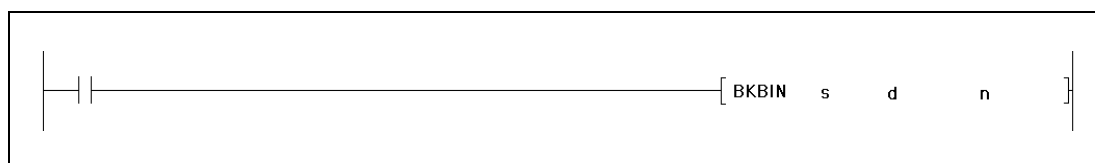
6.3.12 BKBIN, BKBINP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi
MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	—	SMO	4	
d	—	●	●	—	—	—	—	—			
n	●	●	●	●	●	●	●	—			

GX IEC
DeveloperGX
Developer

Variabili

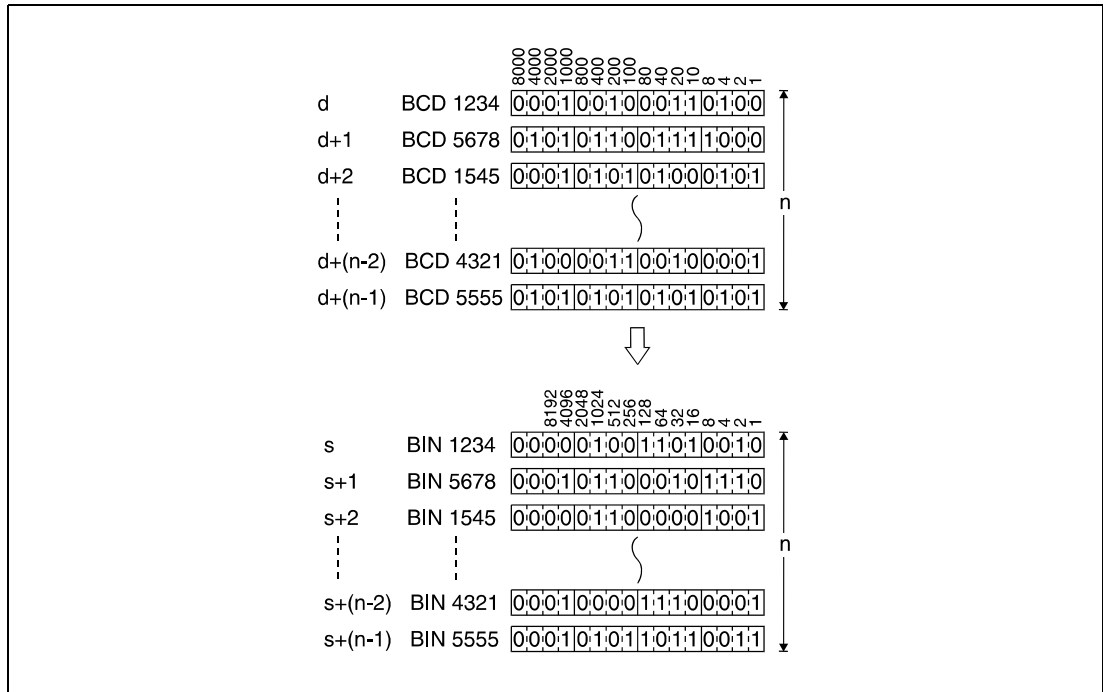
Dati impostati	Significato	Tipo dati
s	Primo numero dell'operando che contiene il dato BCD da convertire	BCD 4-cifre
d	Primo numero dell'operando che memorizza il dato BIN convertito.	BIN 16-bit
n	Numero di blocchi dati da convertire.	BIN 16-bit

Funzioni Conversione da blocchi dati BCD a blocchi dati BIN

BKBIN Conversione da blocchi dati BCD a 4 cifre a blocchi dati BIN a 16 bit

Questa istruzione converte ciascun ennesimo blocco dati BCD a 4 cifre in s nell'ennesimo blocco dati BIN a 16 bit. I dati convertiti sono memorizzati in d.

Il blocco dati BIN a 16 bit indicato da s deve essere nel campo da 0 a 9999.



Errori di esecuzione

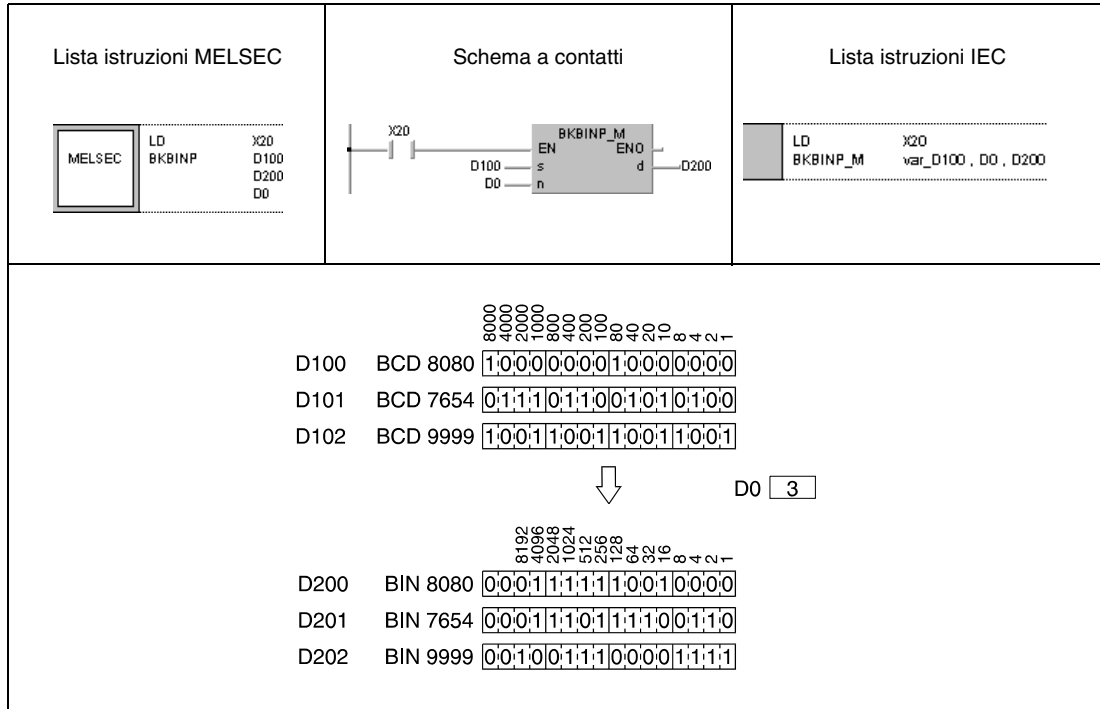
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il numero di blocchi dati indicato da n supera i numeri dispositivo indicati da s e d.
- Il blocco dati BCD in s supera il campo valori ammesso da 0 a 9999.
- I numeri degli operandi indicati da s e d sono sovrapposti.

Per dettagli sulla indicizzazione, consultare il capitolo 3.6.

Esempio di programma BKBINP

Il programma seguente converte il dato del blocco BCD a 4 cifre in D100 in un blocco dati BIN a 16 bit sul fronte di salita di X20. I dati convertiti sono memorizzati in D200. Il numero di blocchi convertiti (3) è contenuto in D0

**NOTA**

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.4 Istruzioni per trasferimento dati

Queste istruzioni trasferiscono, invertono o scambiano dati. Sono disponibili in totale 24 istruzioni:

NOTA

I dati trasferiti rimangono memorizzati fino a quando non vengono sostituiti. I dati rimangono quindi memorizzati anche se la condizione di ingresso del trasferimento non è più vera.

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Trasferimento dati BIN (16-/32-bit)	MOV	MOV_M
	MOVP	MOVP_M
	DMOV	DMOV_M
	DMOVP	DMOVP_M
Trasferimento di Dati virgola mobile	EMOV	EMOV_M
	EMOVP	EMOVP_M
Trasferimento di Dati a stringa di caratteri	\$MOV	STRING_MOV_M
	\$MOVP	STRING_MOVP_M
Trasferimento dati BIN invertiti (16-/32-bit)	CML	CML_M
	CMLP	CMLP_M
	DCML	DCML_M
	DCMLP	DCMLP_M
Trasferimento blocchi dati	BMOV	BMOV_M
	BMOVP	BMOVP_M
Trasferimento blocchi dati identici	FMOV	FMOV_M
	FMOVP	FMOVP_M
Scambio dati BIN (16-/32-bit)	XCH	XCH_M
	XCHP	XCHP_M
	DXCH	DXCH_M
	DXCHP	DXCHP_M
Scambio dati BIN (blocchi 16-bit)	BXCH	BXCH_M
	BXCHP	BXCHP_M
Scambio byte (byte leggero e pesante)	SWAP	SWAP_MD
	SWAPP	SWAP_P_MD

NOTA

Utilizzare comandi IEC con gli editor IEC

6.4.1 MOV, MOVP, DMOV, DMOVP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi
MELSEC A

Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Rapporto																
Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello				M9012	M9010 M9011															
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1							Z	V	K	H (16#)	P	I	N								
MOV																																		
s	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●					K1 ↓ K4	5	1	●								●	
d		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●																		
DMOV																																		
s	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●					K1 ↓ K8	7	1	●								●	
d		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●																		

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

Operandi
MELSEC Q

Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word							
s	●	●	●	●	●	●	●	—	—	—	3 ¹⁾
d	●	●	●	●	●	●	—	—	—	—	

¹ Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.



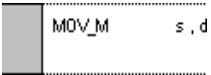
Se viene usata una CPU QnA oppure una CPU a processore singolo del System Q: 3

Se viene usata una CPU multiprocessore del System Q con operandi interni a word (tranne file registri ZR) o costanti: 2

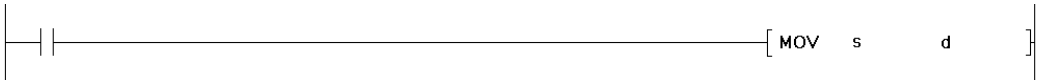
Operandi a bit con numero operando multiplo di 16, in cui la cifra indicata è K4, e che non usa indicizzazione: 2

Se viene usata una CPU multiprocessore del System Q con operandi diversi da quelli menzionati: 3

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
		

GX
Developer


--

Variabili

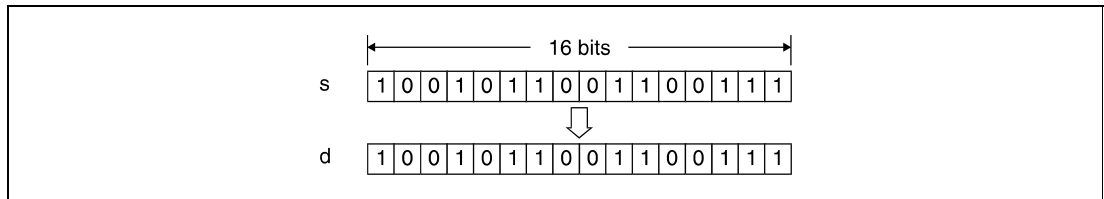
Dati impostati	Significato	Tipo dati
s	Dato sorgente o primo numero dell'operando contenente il dato da trasferire.	BIN 16/32-bit
d	Primo numero dell'operando di destinazione per la memorizzazione dei dati trasferiti.	

Funzioni

Trasferimento dati BIN

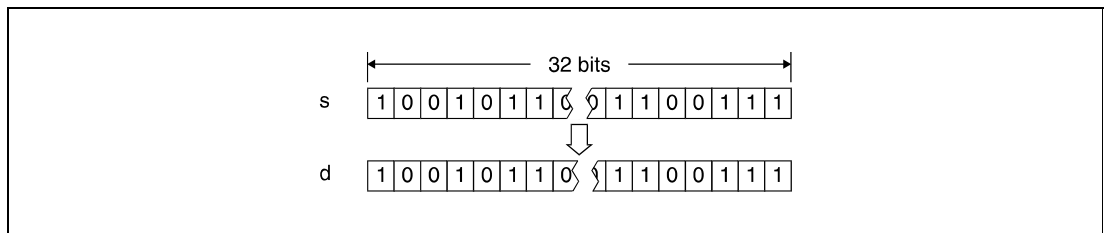
MOV Trasferimento dati BIN a 16-bit

L'istruzione MOV trasferisce il dato BIN a 16 bit in s nell'operando indicato da d.



DMOV Trasferimento dati BIN a 32-bit

L'istruzione DMOV trasferisce il dato BIN a 32 bit in s nell'operando indicato da d.



Programma di esempio 1

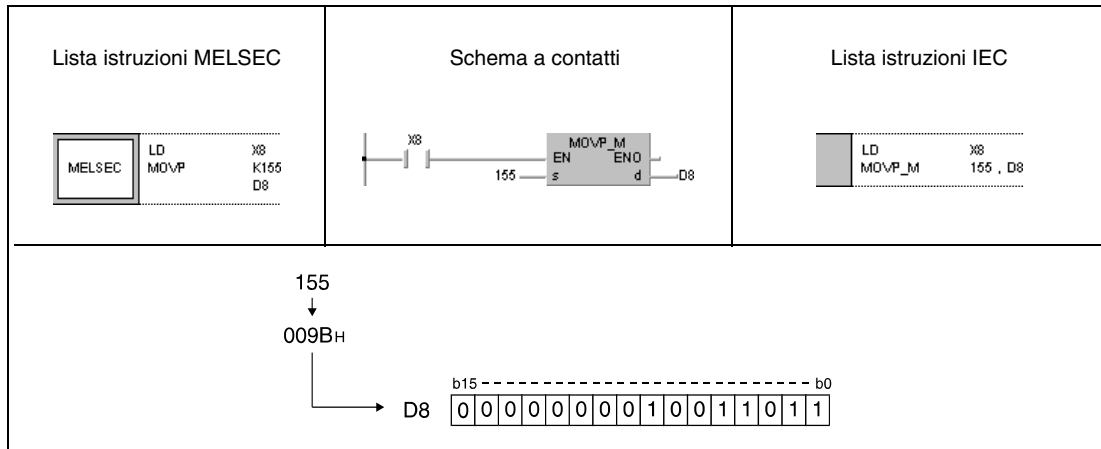
MOVP

Questo programma trasferisce i dati da X0 a XB in D8 sul fronte di salita di SM400.

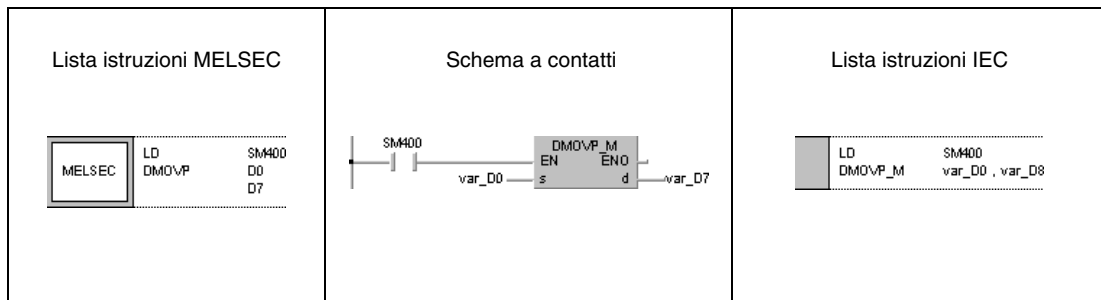
<p>Lista istruzioni MELSEC</p> <pre> MELSEC LD SM400 MOVP K3#D D8 </pre>	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p> <pre> LD SM400 MOVP_M K3#D, D8 </pre>
--	--------------------------	--

Programma di esempio 2 **MOVP**

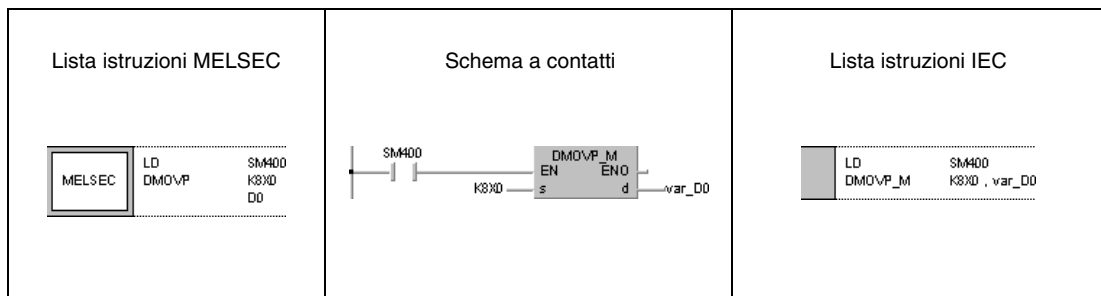
Il programma che segue trasferisce la costante 155 come valore BIN in D8 sul fronte di salita di X8.

**Programma di esempio 3** **DMOVP**

Il programma che segue trasferisce il dato da D0 e D1 a D7 e D8 sul fronte di salita di SM400.

**Programma di esempio 4** **DMOVP**

Questo programma trasferisce i dati da X0 a X1F in D0 e D1, sul fronte di salita di SM400.

**NOTA**

I programmi di esempio 3 e 4 non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provocano messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.4.2 EMOV, EMOVP

CPU


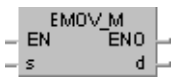
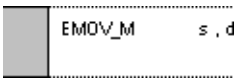
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

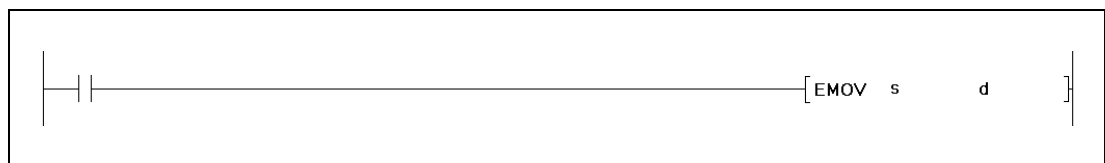
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti E			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	●	●	—	●	—	3	
d	—	●	●	—	●	●	—	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
--	--	---

GX Developer

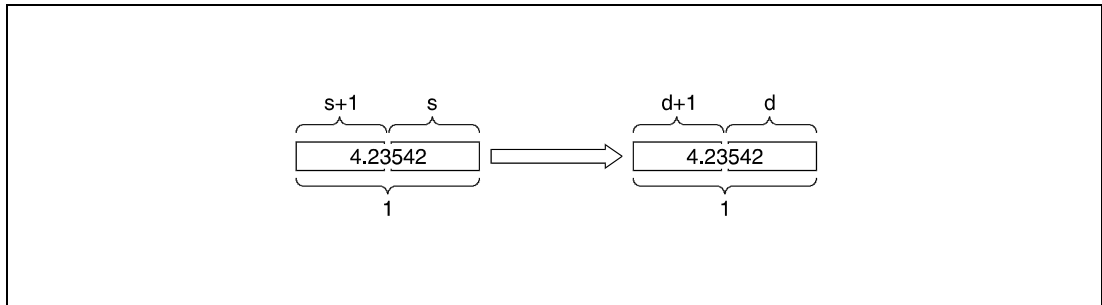


Variabili

Dati impostati	Significato	Tipo dati
s	Dato in virgola mobile o primo numero dell'operando contenente il dato da trasferire.	Numero reale
d	Primo numero dell'operando che memorizza il dato in virgola mobile trasferito.	

Funzioni **Trasferimento dati in virgola mobile****EMOV** **Trasferimento dati in virgola mobile**

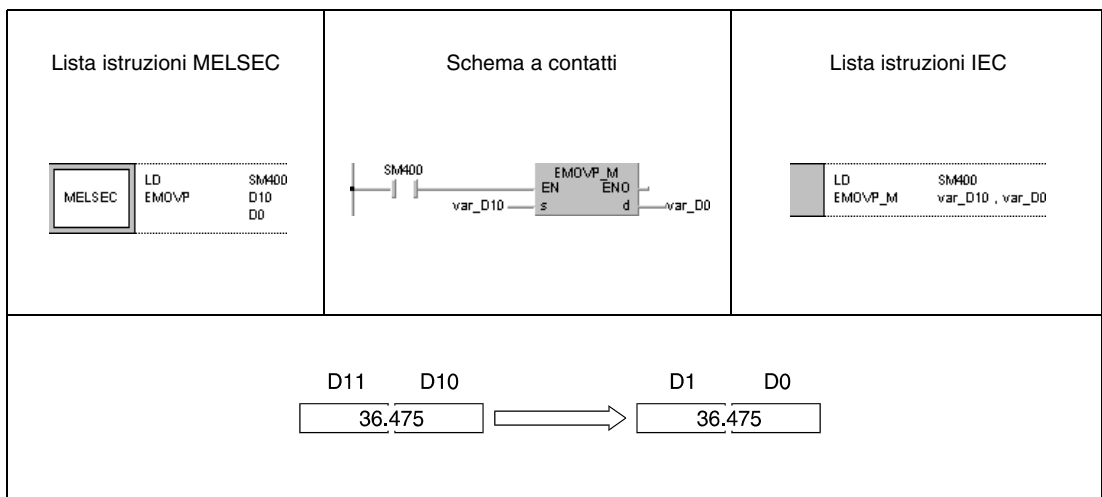
L'istruzione EMOV trasferisce il dato in virgola mobile in s nell'operando indicato da d.



¹ Dati in virgola mobile, tipo di dati per numeri reali

Programma di esempio 1**EMOVP**

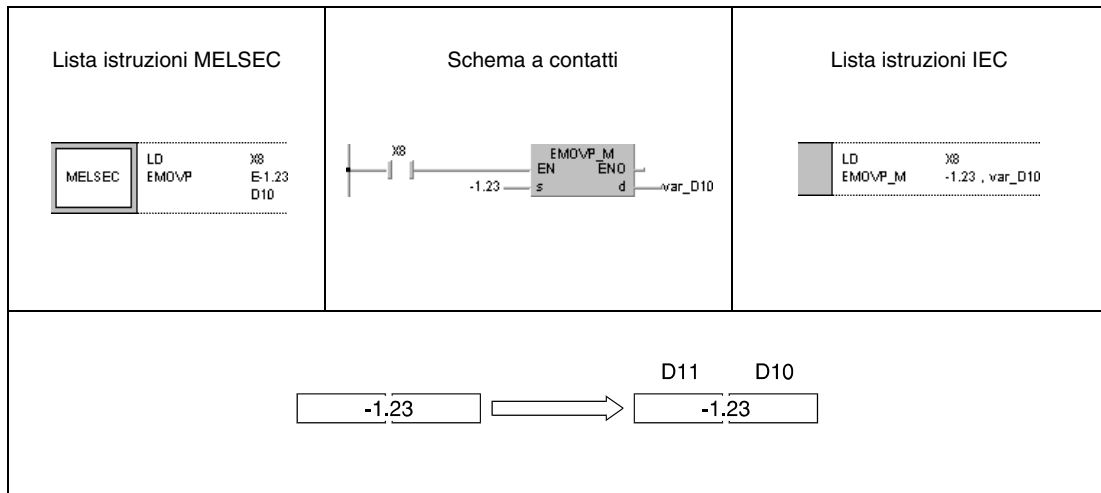
Questo programma trasferisce i dati in virgola mobile da D10 e D11 in D0 e D1, sul fronte di salita di SM400.



Programma di esempio 2

EMOVP

Il programma che segue trasferisce il numero reale 1,23 in D10 e D11 sul fronte di salita di X8.



NOTA

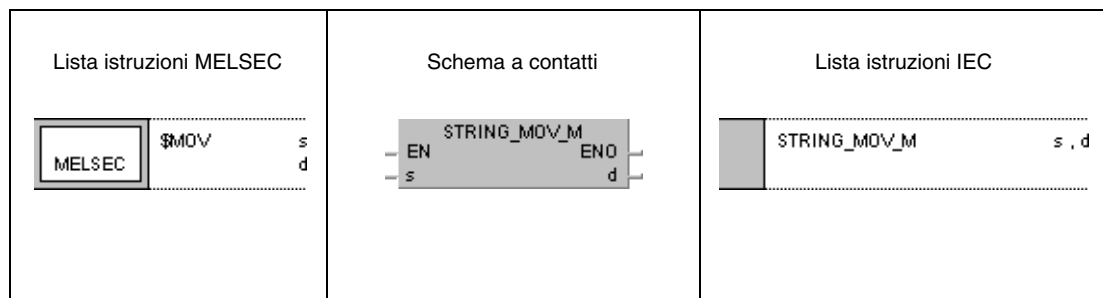
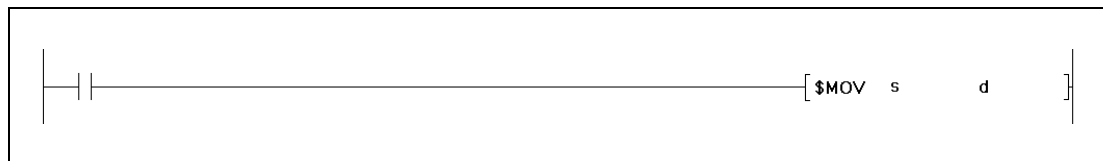
Questi programmi non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provocano messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.4.3 \$MOV, \$MOVP**CPU**

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti \$			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	●	—	SM0	3
d	—	●	●	—	—	—	—	—	—		

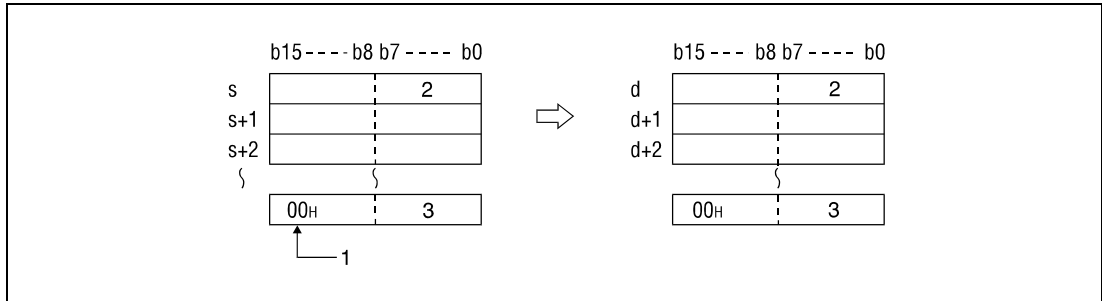
GX IEC Developer**GX Developer****Variabili**

Dati impostati	Significato	Tipo dati
s	Stringa di caratteri o primo numero dell'operando contenente il dato da trasferire.	Stringa caratteri
d	Primo numero dell'operando che memorizza la stringa di caratteri trasferita.	

Funzioni Trasferimento stringhe di caratteri

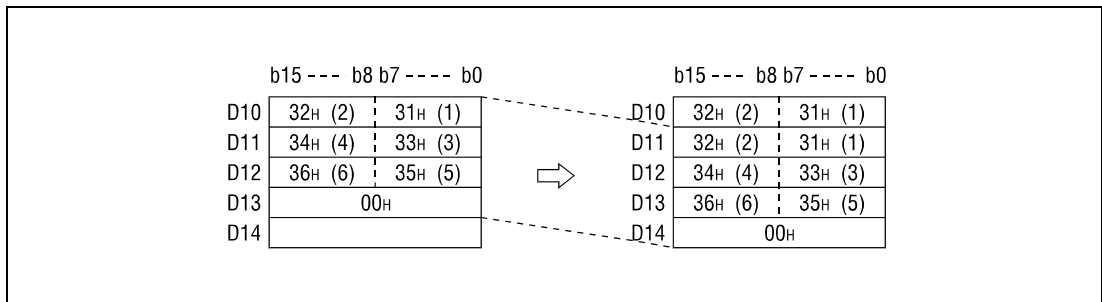
\$MOV Trasferimento stringhe di caratteri

L'istruzione \$MOV trasferisce la stringa di caratteri s in d. L'istruzione trasferisce la stringa dal primo numero dell'operando indicato da s, fino al numero di operando contenente il codice "00H" (fine stringa) in una sola operazione.

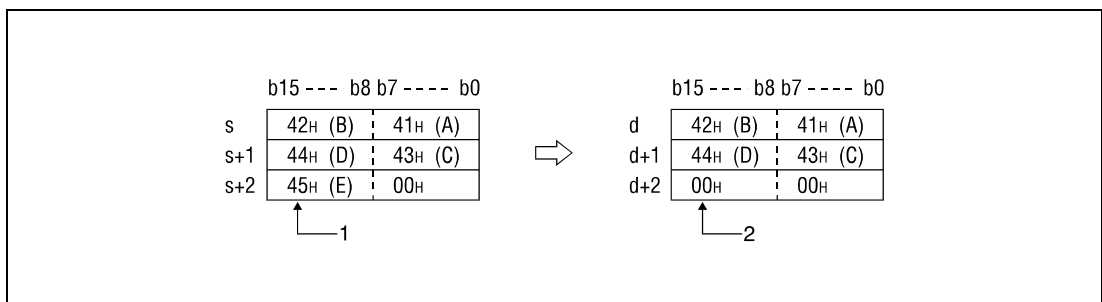


- ¹ Indica la fine della stringa
- ² Primo carattere
- ³ ennesimo carattere

L'istruzione \$MOV viene eseguita senza segnalazione di errore anche se il campo di operandi che contiene la stringa da trasferire (da s a s+n) si sovrappone al campo di operandi che memorizzano il dato trasferito (da d a d+n). Se i dati della stringa da D10 a D13 vengono trasferiti da D11 a D14, l'istruzione \$MOV si comporta come segue:



Se il codice "00H" è contenuto nel byte meno significativo di s+n, i caratteri che seguono vengono omessi. In d+n il codice "00H" viene inserito sia sul byte leggero che sul byte pesante:



- ¹ Carattere non trasferito
- ² "00H" memorizzato automaticamente.

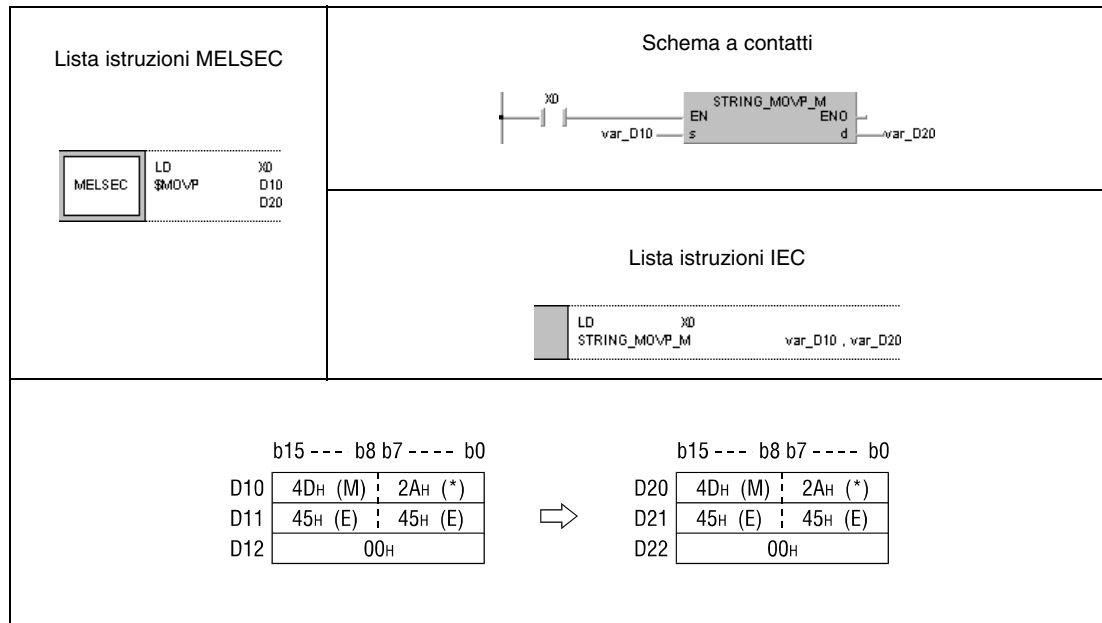
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il codice "00H" non esiste nella stringa indicata da s (codice di errore: 4101).
- I dati della stringa in s non possono essere trasferiti completamente in d.

Esempio di programma

Il programma che segue trasferisce sul fronte di salita di X0 la stringa di caratteri da D10 a D12 in D20 fino a D22.

**NOTA**

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.4.4 CML, CMLP, DCML, DCMLP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Riporto	Errore						
Operandi a bit						Operandi a word (16-bit)						Costanti		Puntatore							Livello					
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H (16#)	P	I	N			M9012	M9010 M9011		
CML																										
s	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●					K1 ↓ K4	5	●		●
d		●	●	●	●	●	●	●	●	●	●	●	●	●	●								●			●
DCML																										
s	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●					K1 ↓ K8	7	●		●
d		●	●	●	●	●	●	●	●	●	●	●	●	●	●								●			●

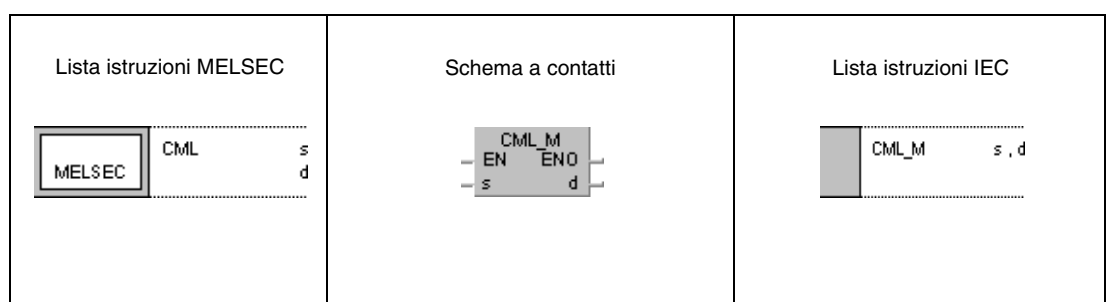
¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

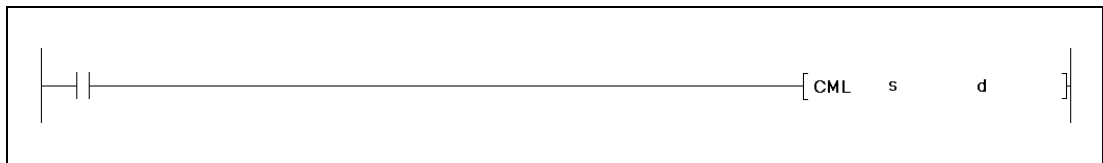
Operandi MELSEC Q

Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word							
s	●	●	●	●	●	●	●	—	—	—	3 ¹⁾
d	●	●	●	●	●	●	—	—	—	—	

¹ Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.
 Se viene usata una CPU QnA oppure una CPU a processore singolo del System Q: 3
 Se viene usata una CPU multiprocessore del System Q con operandi interni a word (tranne registro file ZR) o costanti: 2
 Se viene usata una CPU multiprocessore del System Q con operandi a bit il cui numero operando sia multiplo di 16, con indicazione di cifra K4, e che non usa indicizzazione: 2
 Se viene usata una CPU multiprocessore del System Q con operandi diversi da quelli menzionati:3

GX IEC Developer



GX
Developer

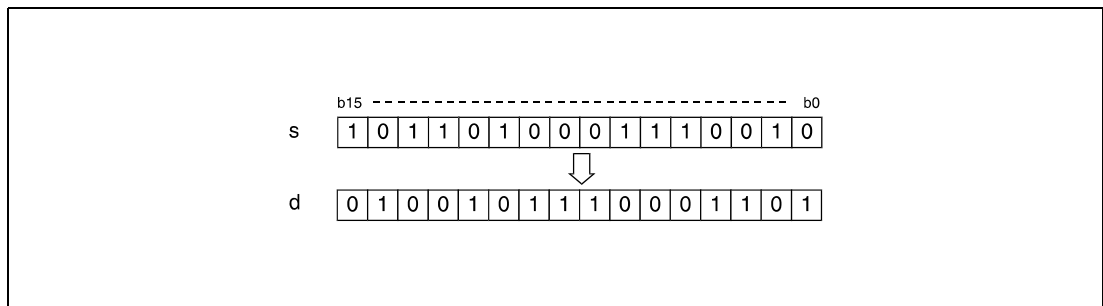
Variabili

Dati impostati	Significato	Tipo dati
s	Dato BIN o primo numero dell'operando contenente i dati da invertire.	BIN 16/32-bit
d	Primo numero dell'operando che memorizza il dati invertiti.	

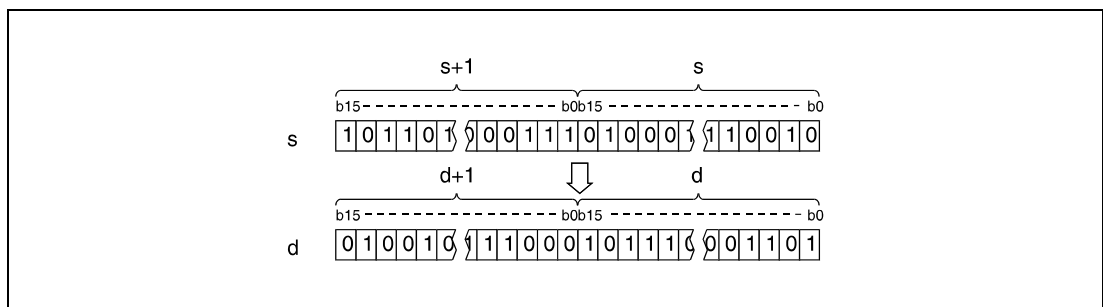
Funzioni

Inversione dati BIN**CML Inversione dati BIN a 16-bit**

Il dato BIN a 16 bit in s viene invertito bit a bit. Il risultato è memorizzato in d.

**DCML Inversione dati BIN a 32-bit**

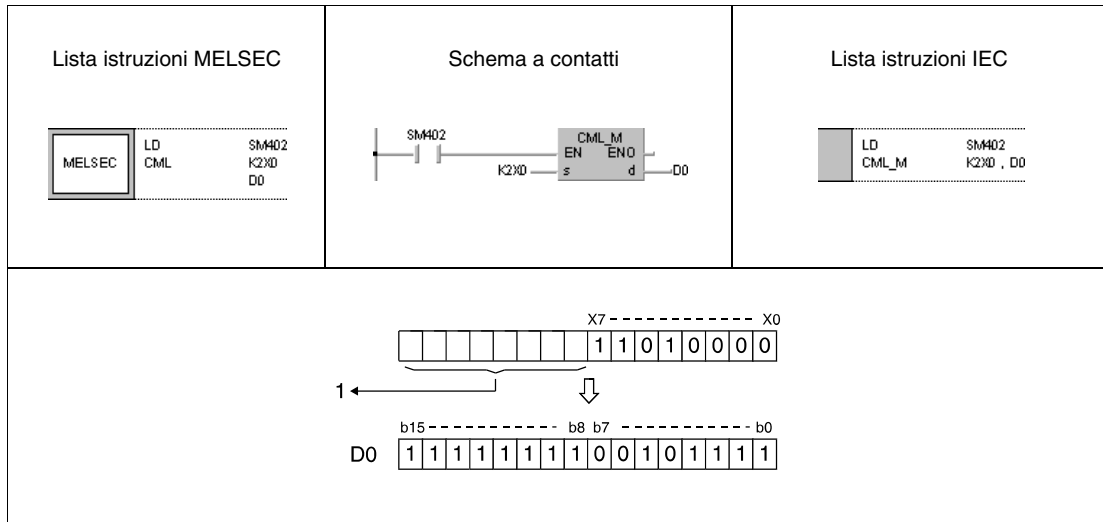
Il dato BIN a 32 bit in s viene invertito bit a bit. Il risultato è memorizzato in d.



Programma di esempio 1

CML

Il programma seguente trasferisce i dati da X0 a X7 in D0 dopo averli invertiti, fino a quando SM402 è uguale a 1.



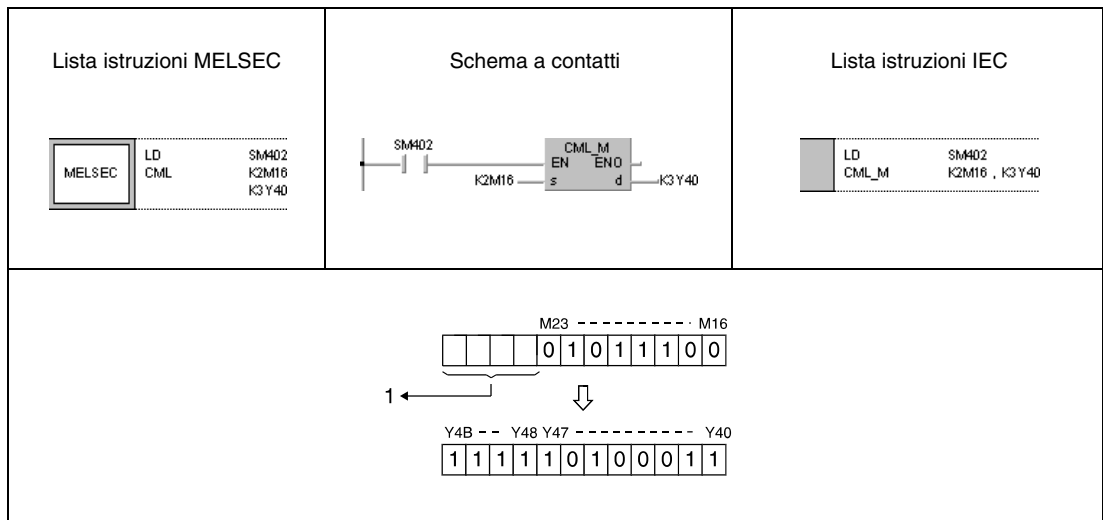
¹ I bit non specificati vengono considerati come 0.

Il numero di bit in s deve essere inferiore al numero di bit in d.

Programma di esempio 2

CML

Il programma seguente trasferisce i dati da M16 a M23 in K3 Y40 (da Y40 a Y4F) dopo averli invertiti, fino a quando SM402 è uguale a 1. Gli operandi da Y48 a Y4B sono impostati a 1, dato che vengono letti come zero.



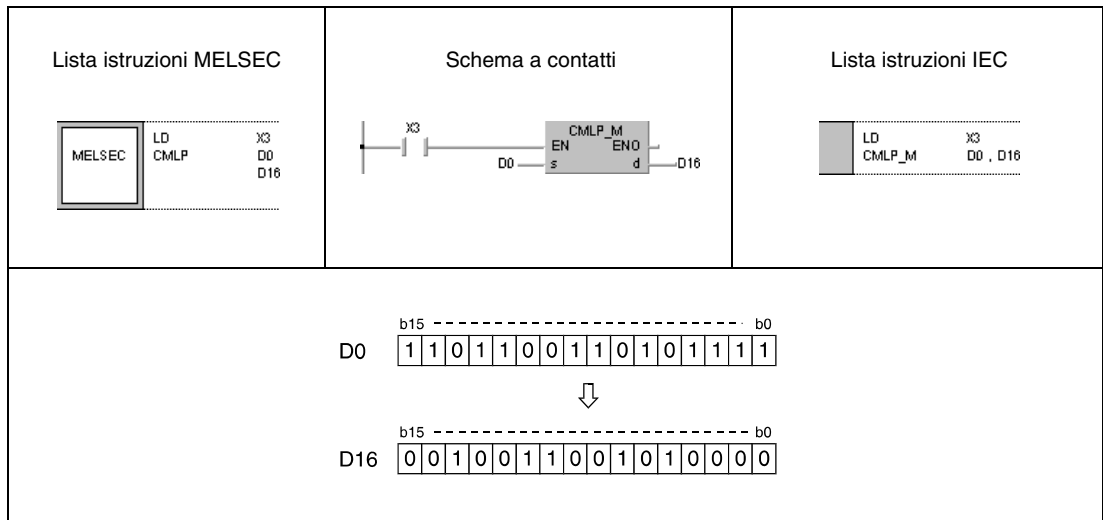
¹ I bit non specificati vengono considerati come 0.

Il numero di bit in s deve essere inferiore al numero di bit in d.

Programma di esempio 3

CMLP

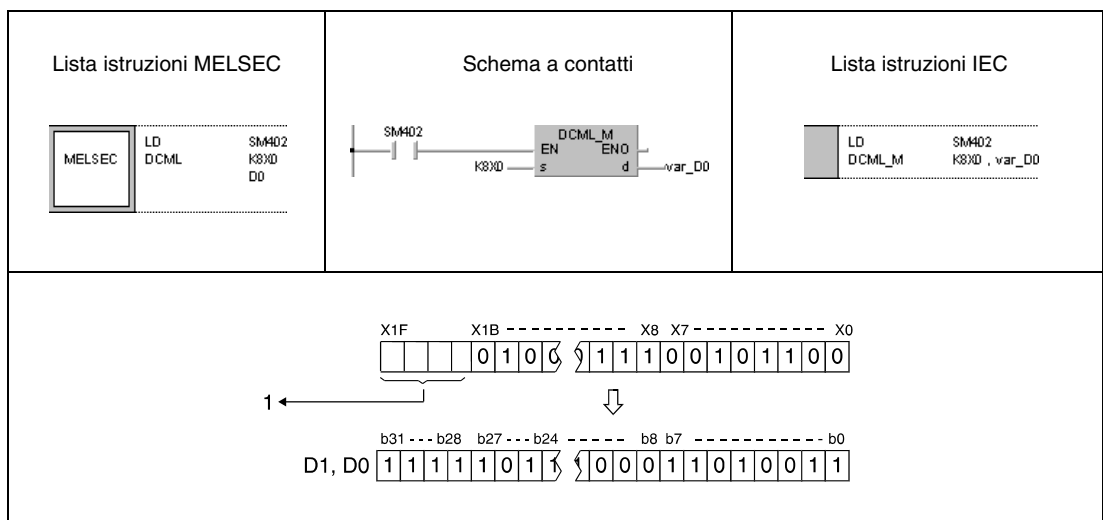
Il programma che segue trasferisce il dato da D0 a D16 dopo averlo invertito, sul fronte di salita di X3.



Programma di esempio 4

DCML

Il programma seguente trasferisce i dati da X0 a X1F in D0 e D1 dopo averli invertiti, fino a quando SM402 è uguale a 1.



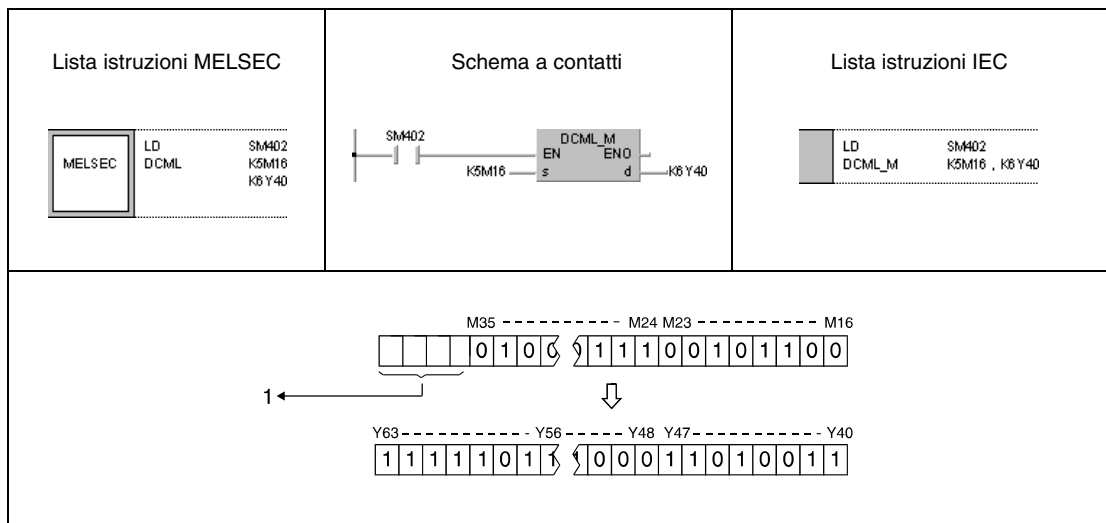
¹ I bit non specificati vengono considerati come 0.

Il numero di bit in s deve essere inferiore al numero di bit in d.

Programma di esempio 5

DCML

Il programma seguente trasferisce i dati da M16 a M35 sulle uscite da Y40 a Y57 dopo averli invertiti, fino a quando SM402 è uguale a 1.

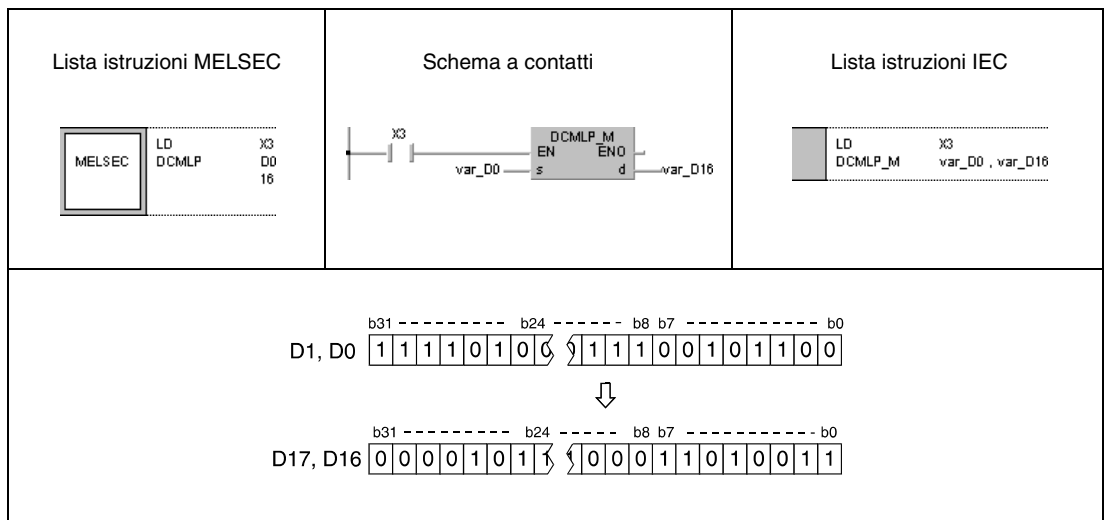


¹ I bit non specificati vengono considerati come 0.

Programma di esempio 6

DCMLP

Il programma che segue trasferisce il dato da D0 e D1 a D16 e D17 dopo averlo invertito, sul fronte di salita di X3.



Il numero di bit in s deve essere inferiore al numero di bit in d.

NOTA

I programmi di esempio 4 e 6 non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provocano messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.4.5 BMOV, BMOVP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

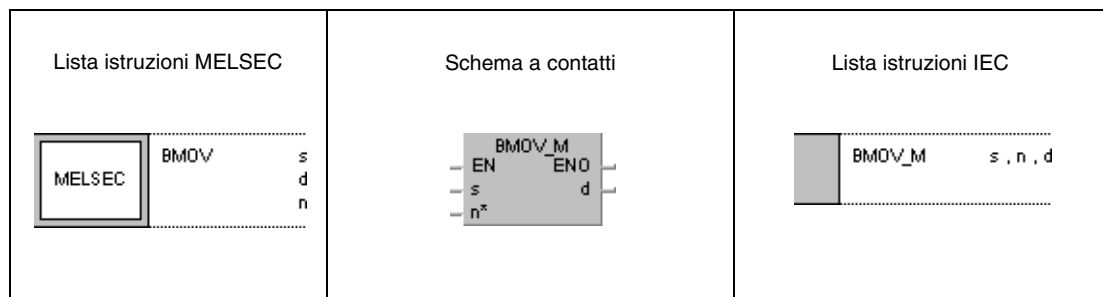
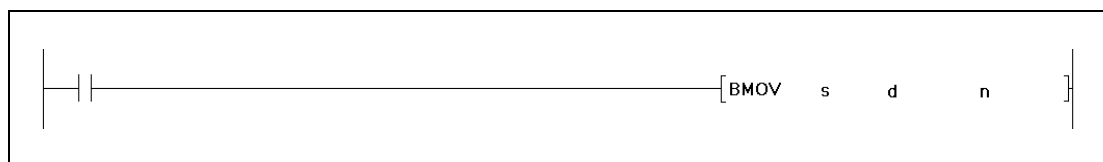
Operandi
MELSEC A

	Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Riporto	Errore				
	Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello									
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z				V	K	H (16#)	P	I	N
s	●	●	●	●	●	●	●	●	●	●	●													
d		●	●	●	●	●	●	●	●	●	●													●
n																	●	●						

¹ Consultare la sezione "Programmazione con CPU AnA, AnAS e AnU" di questo manuale di programmazione per il corrispondente numero di passi.

Operandi
MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	—	—	—	SM0	4
d	●	●	●	●	●	●	—	—	—		
n	●	●	●	●	●	●	●	●	—		

GX IEC
DeveloperGX
Developer

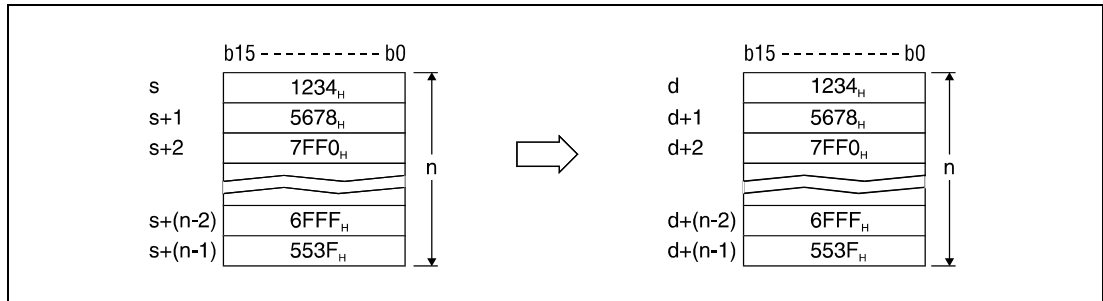
Variabili

Dati impostati	Significato	Tipo dati
s	Primo numero dell'operando che contiene i dati da trasferire	BIN 16-bit
d	Primo numero dell'operando che memorizza i dati trasferiti.	
n	Numero di blocchi dati da trasferire	

Funzioni Trasferimento blocchi dati BIN

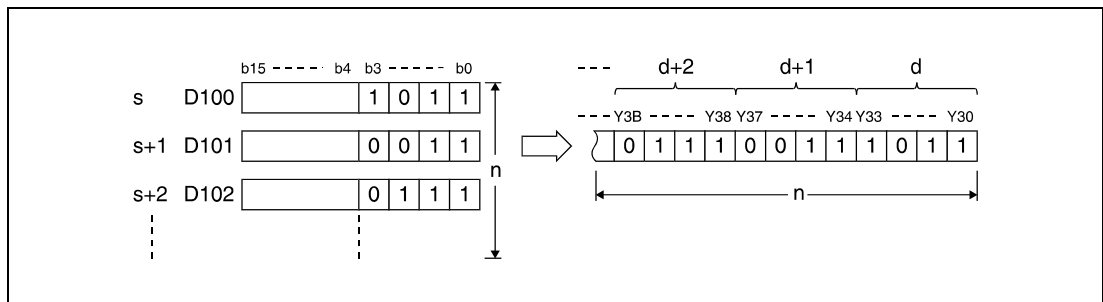
BMOV Trasferimento blocchi dati BIN a 16-bit

L'istruzione BMOV trasferisce un gruppo di blocchi dati in sequenza. Il numero del primo operando che contiene il blocco dati è indicato da s. Il numero dei blocchi dati successivi da trasferire è definito da n. I dati sono trasferiti a partire dall'operando indicato da d.



Un trasferimento può essere eseguito senza errori anche se gli operandi sorgente e destinazione si sovrappongono. Il trasferimento inizia dall'operando corrispondente all'indirizzo più basso s, fino all'operando con l'indirizzo più alto s+(n-1).

Se s è un operando a word e d è un operando a bit, il numero di bit determinati dall'indicazione di cifra dell'operando a bit divengono i bit oggetto dell'operando a word. Se d viene specificato con K1Y30, i bit oggetto dell'operando a word s sono i 4 bit più leggeri.



Se s e d sono operandi a bit, il numero di bit di s e d deve essere uguale.

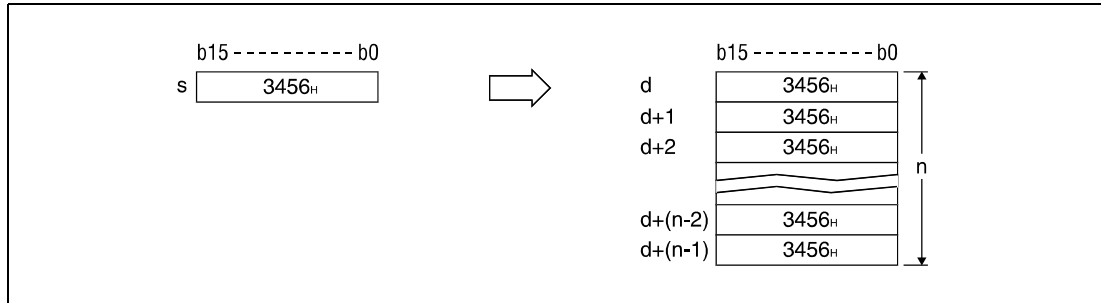
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

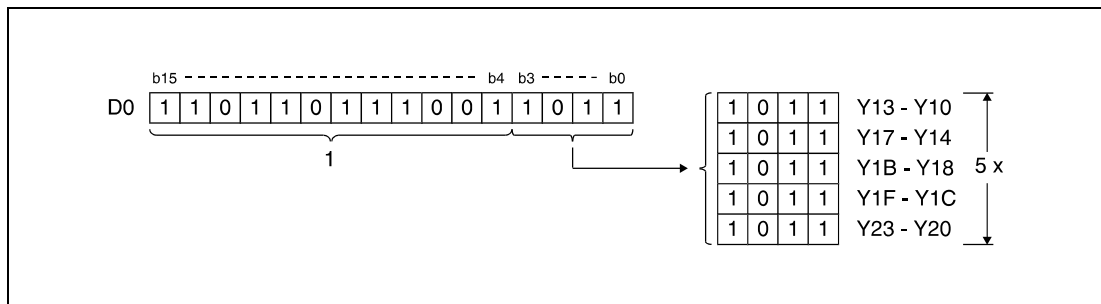
- Il numero di blocchi dati indicato da n supera il numero di dispositivi indicati da s e d (serie Q e System Q = codice di errore: 4101).

Funzioni Trasferimento blocchi dati BIN identici**FMOV Trasferimento blocchi dati BIN a 16-bit identici**

L'istruzione FMOV trasferisce i dati da s a d fino a d+(n-1). Ciascun operando del blocco dati da d a d+(n-1) contiene il valore di s.



Se s è un operando a word e d è un operando a bit, il numero di bit determinati dall'indicazione di cifra dell'operando a bit divengono i bit oggetto dell'operando a word.



¹ Questi bit sono ignorati.

Se s e d sono operandi a bit, il numero di bit di s e d deve essere uguale.

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

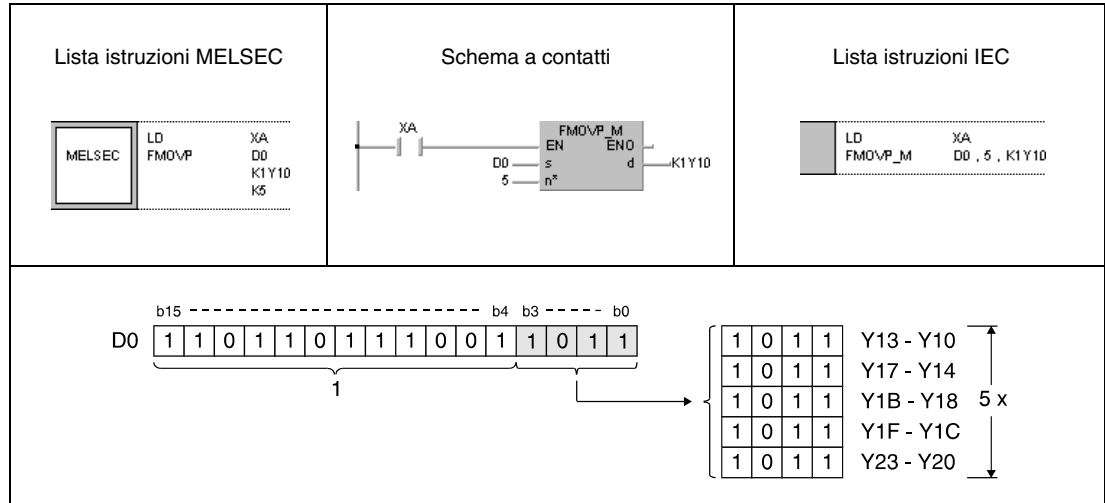
- Il numero di blocchi dati indicato da n supera i numeri dispositivo indicati da s e d (serie Q e System Q = codice di errore: 4101).

Programma di esempio 1

FMOVP

Sul fronte di salita di XA, il programma che segue trasferisce i 4 bit più leggeri (da b0 a b3) di D0 sulle uscite da Y10 a Y23. Il numero di blocchi (5) è definito dalla costante K5.

La figura mostra la struttura dei bit prima e dopo il trasferimento.



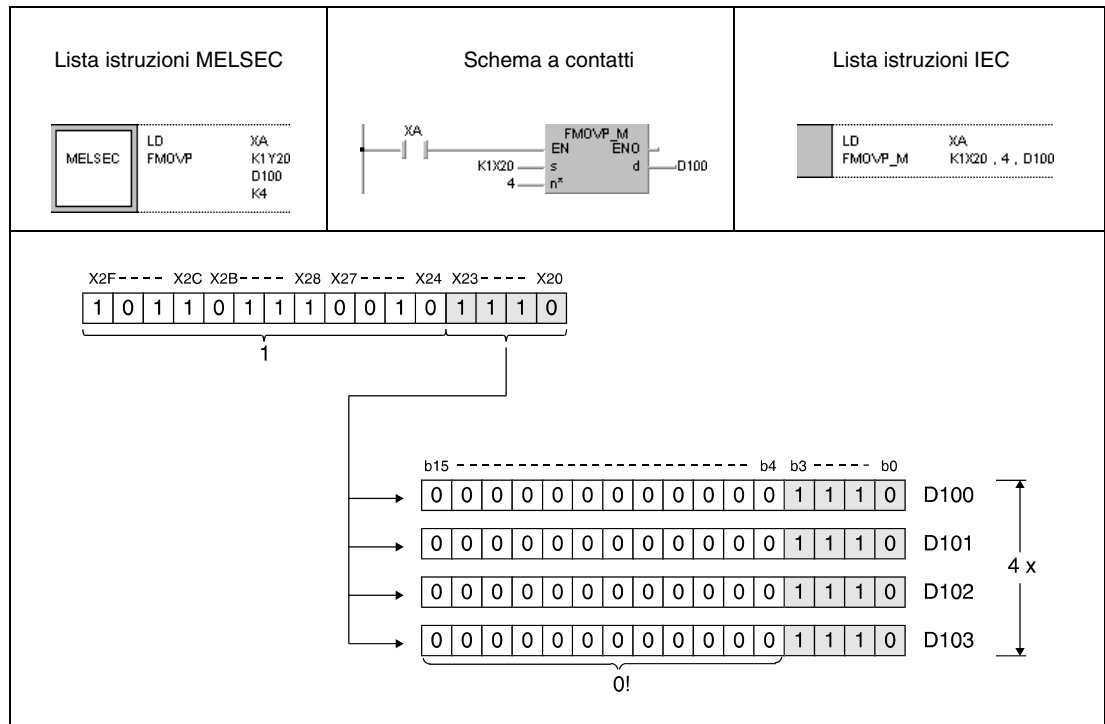
¹ Questi bit sono ignorati.

Programma di esempio 2

FMOVP

Sul fronte di salita di XA, il programma che segue trasferisce i dati da X20 a X23 in D100 fino a D103. Il numero di blocchi (4) da trasferire è definito dalla costante K4.

La figura mostra la struttura dei bit prima e dopo il trasferimento.



¹ Questi bit sono ignorati.

6.4.7 XCH, XCHP, DXCH, DXCHP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

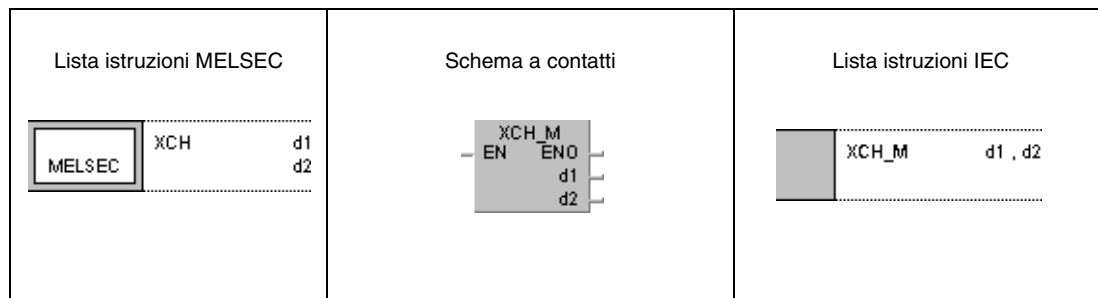
Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Riporto	Errore						
Operandi a bit						Operandi a word (16-bit)						Costanti	Punta-tore	Livello												
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H (16#)	P	I	N	M9012	M9010 M9011				
XCH																										
d1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●						K1 ↓ K4	5	●		●	
d2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●								1			●
DXCH																										
d1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●						K1 ↓ K8	7	●		●	
d2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●								1			●

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

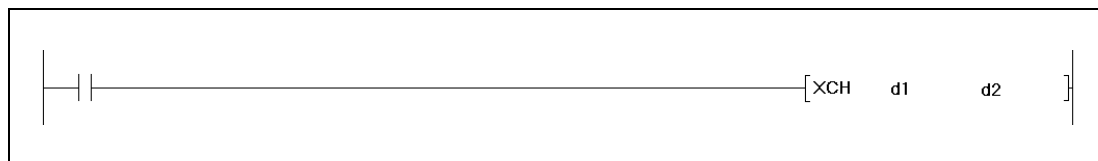
Operandi MELSEC Q

Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word							
d1	●	●	●	●	●	●	—	—	—	3	
d2	●	●	●	●	●	●	—	—	—		

GX IEC Developer



GX Developer



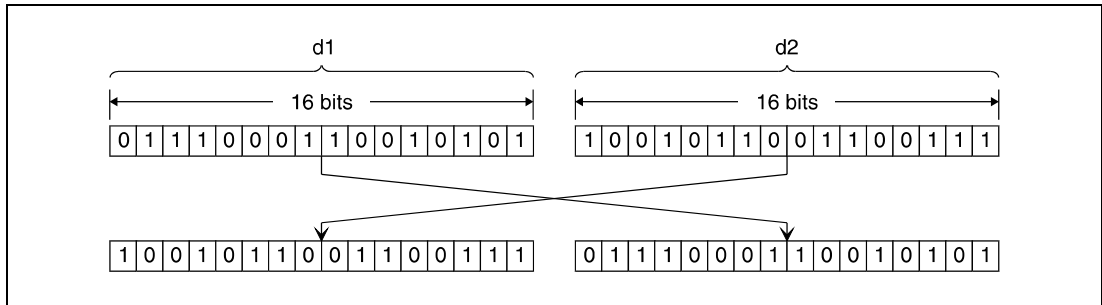
Variabili

Dati impostati	Significato	Tipo dati
d1	Primo numero dell'operando che contiene i dati da scambiare	BIN 16/32-bit
d2		

Funzioni Scambio dati BIN

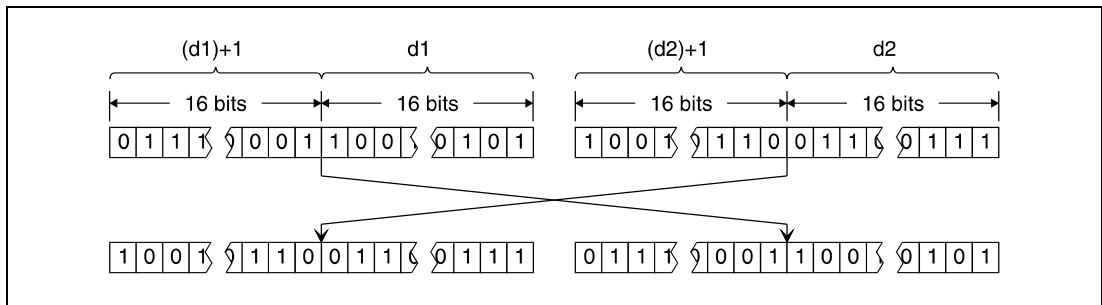
XCH Scambio dati BIN 16-bit

L'istruzione XCH scambia il dato BIN a 16 bit in d1 con il dato BIN a 16 bit in d2.



DXCH Scambio dati BIN 32-bit

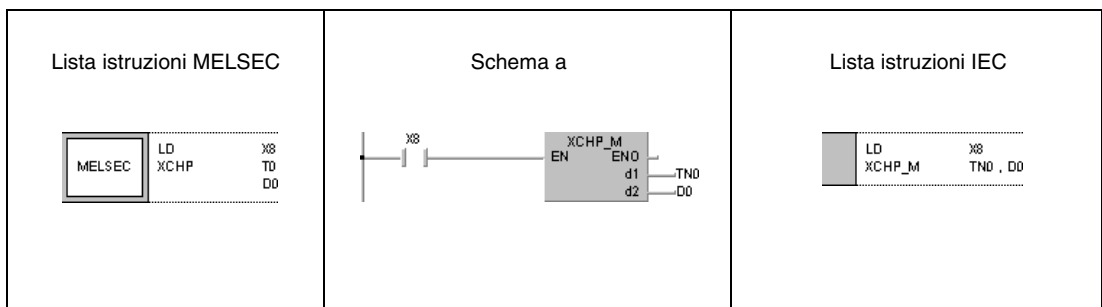
L'istruzione DXCH scambia il dato BIN a 32 bit in (d1)+1, d1 con il dato BIN a 32 bit in (d2)+1, d2.



Programma di esempio 1

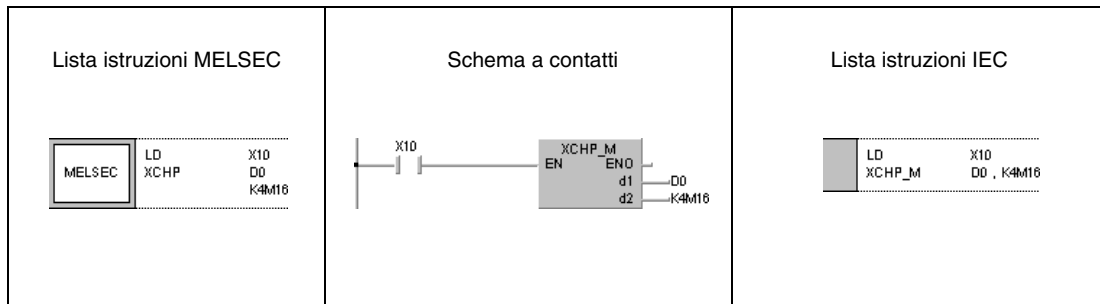
XCHP

Il programma seguente scambia il dato in D0 con il valore attuale di T0 sul fronte di salita di X8.

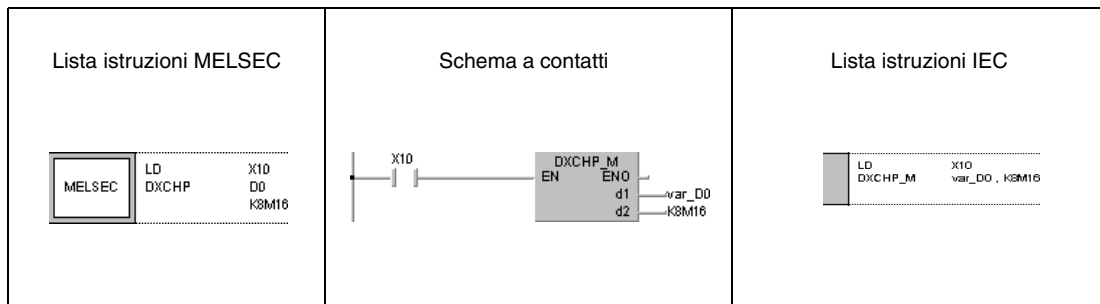


Programma di esempio 2 XCHP

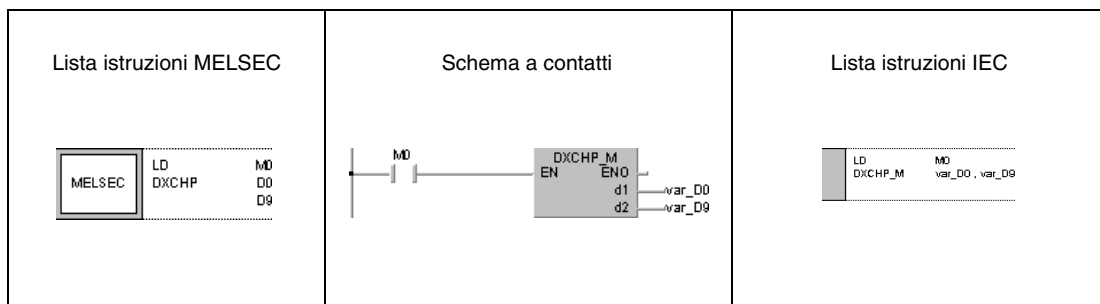
Il programma seguente scambia il dato in D0 con i dati da M16 a M31 sul fronte di salita di X10.

**Programma di esempio 3** DXCHP

Il programma seguente scambia il dato in D0 e D1 con i dati da M16 a M47 sul fronte di salita di X10.

**Programma di esempio 4** DXCHP

Il programma seguente scambia il dato in D0 e D1 con il dato in D9 e D10 sul fronte di salita di M0.

**NOTA**

I programmi di esempio 3 e 4 non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provocano messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.4.8 BXCH, BXCHP

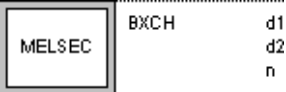
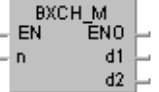
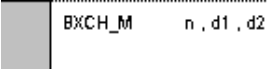
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

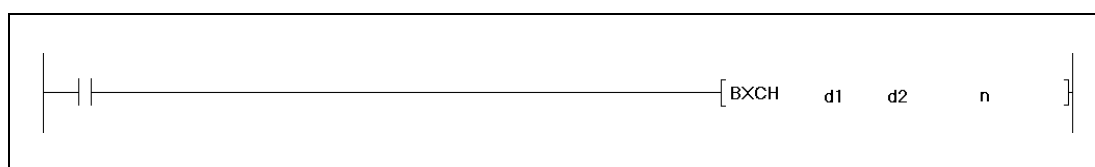
Operandi MELSEC Q

	Operandi utilizzabili									Segnale di errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
d1	—	●	●	—	—	—	—	—	—	SMO	4
d2	—	●	●	—	—	—	—	—	—		
n	●	●	●	●	●	●	●	●	—		

GX IEC Developer

Lista istruzioni MELSEC 	Schema a contatti 	Lista istruzioni IEC 
--	--	---

GX Developer

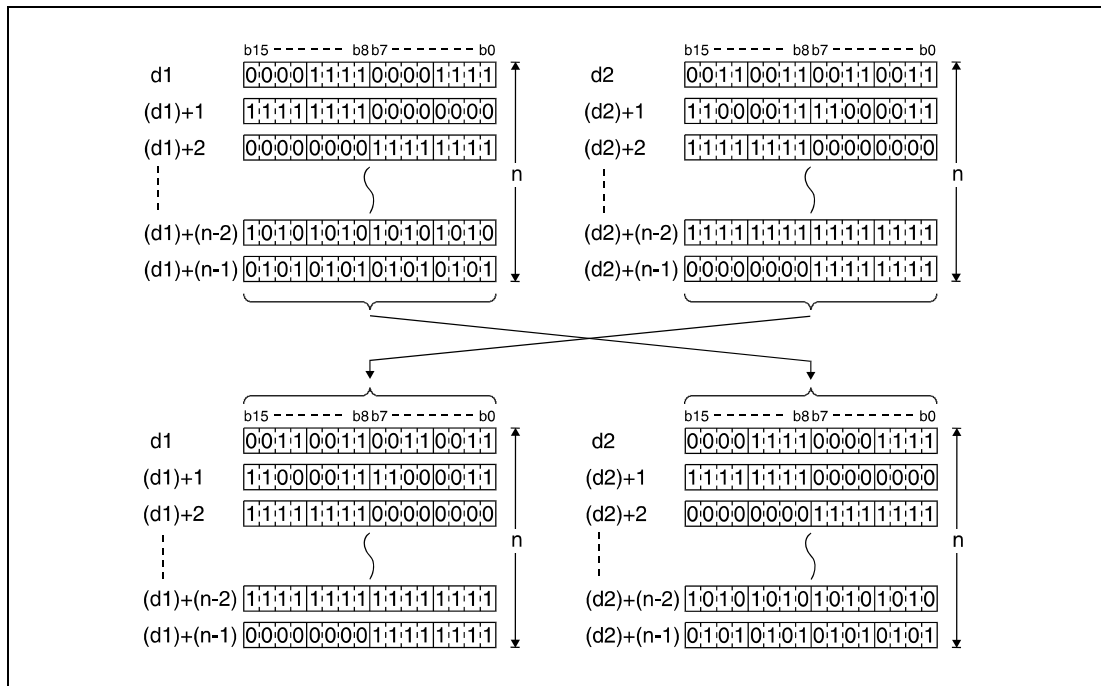


Variabili

Dati impostati	Significato	Tipo dati
d1	Primo numero dell'operando che contiene i dati da scambiare	BIN 16-bit
d2		
n	Numero di scambi	

Funzioni **Scambio blocchi dati BIN****BXCH** **Scambio blocchi dati BIN 16-bit**

L'istruzione BXCH scambia il blocco dati BIN a 16 bit in d1 con il blocco dati BIN a 16 bit in d2.

**Errori di esecuzione**

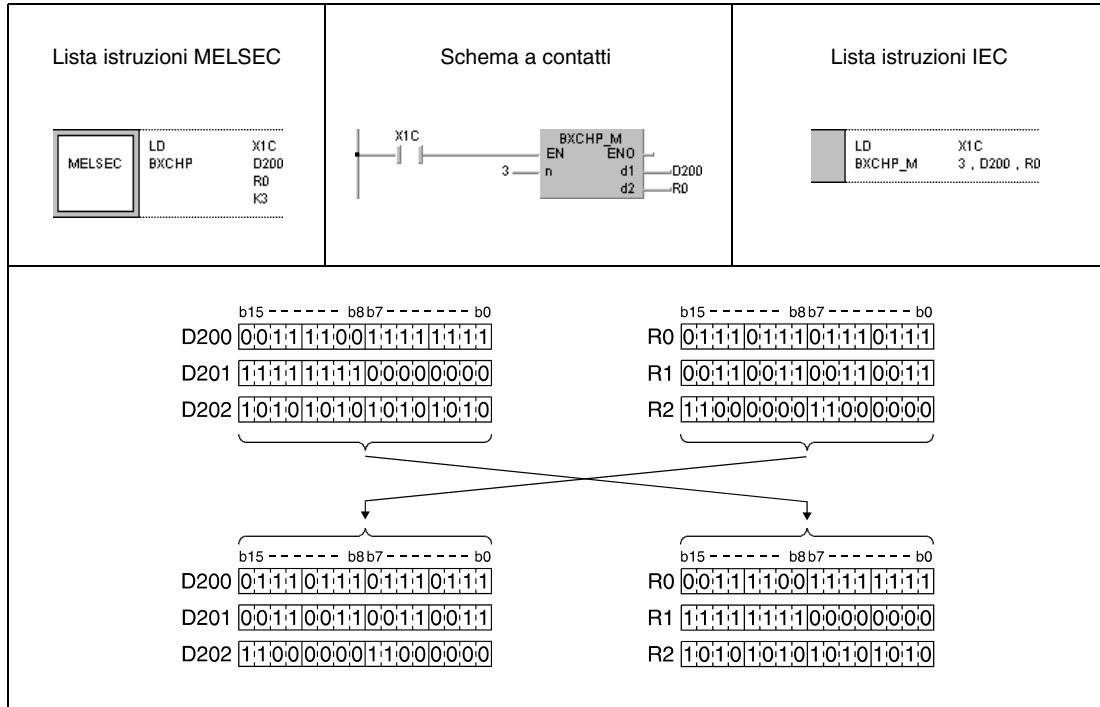
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il numero di blocchi dati indicato da n supera i numeri dispositivo indicati da d1 e d2 (codice di errore: 4101).
- I numeri degli operandi indicati da d1 e d2 sono sovrapposti (codice di errore 4101).

Esempio di programma BXCHP

Il programma seguente scambia i blocchi dati a partire da D200 con i blocchi dati a partire da R0. Il numero di blocchi (3) da scambiare è definito dalla costante K3.

La figura mostra la struttura dei bit prima e dopo il trasferimento.



6.4.9 SWAP, SWAPP

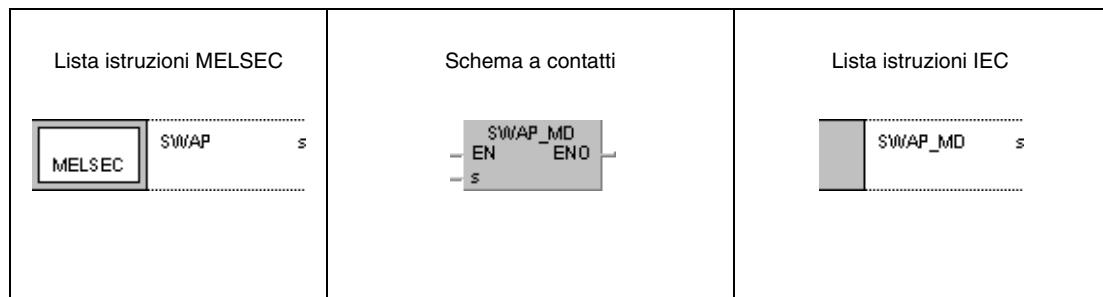
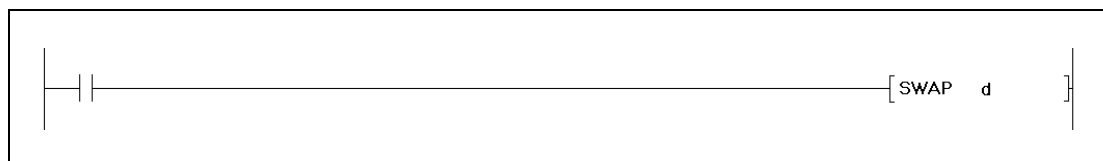
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	●

¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

Operandi
MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	●	—	—	—	3

GX IEC
DeveloperGX
Developer

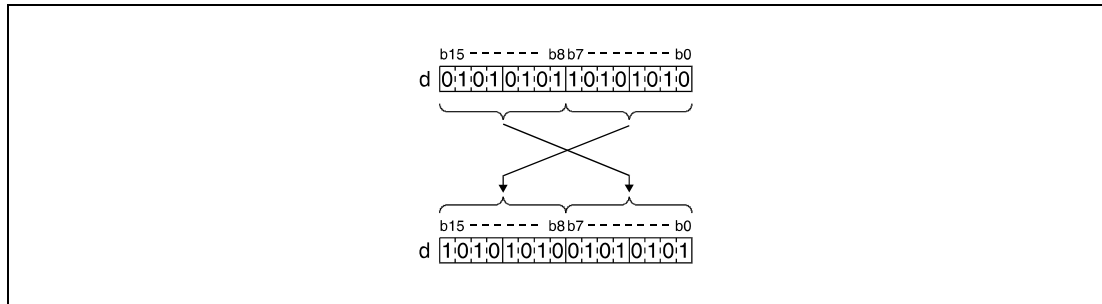
Variabili

Dati impostati	Significato	Tipo dati
s	Primo numero dell'operando che contiene i dati da scambiare	BIN 16-bit

Funzioni **Scambio byte leggero e pesante**

SWAP **Scambio fra byte leggero e byte pesante**

Istruzione SWAP scambia fra loro gli 8 bit leggeri e pesanti (byte leggero e pesante) del dato BIN 16-bit in s.



Esempio di programma

SWAPP

Il programma seguente scambia fra loro gli 8 bit leggeri e pesanti di R10 sul fronte di salita di X10.

<p style="text-align: center;">Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 2px; margin: 5px auto; width: fit-content;"> <p>MELSEC</p> </div> <table style="width: 100%; border-collapse: collapse; margin-top: 5px;"> <tr> <td style="border-right: 1px solid black; padding: 2px;">LD</td> <td style="padding: 2px;">X10</td> </tr> <tr> <td style="border-right: 1px solid black; padding: 2px;">SWAPP</td> <td style="padding: 2px;">R10</td> </tr> </table>	LD	X10	SWAPP	R10	<p>Schema a</p>	<p style="text-align: center;">Lista istruzioni IEC</p> <table style="width: 100%; border-collapse: collapse; margin-top: 5px;"> <tr> <td style="border-right: 1px solid black; padding: 2px;">LD</td> <td style="padding: 2px;">X10</td> </tr> <tr> <td style="border-right: 1px solid black; padding: 2px;">SWAP_P_MD</td> <td style="padding: 2px;">R10</td> </tr> </table>	LD	X10	SWAP_P_MD	R10
LD	X10									
SWAPP	R10									
LD	X10									
SWAP_P_MD	R10									
<p>The diagram shows the state of register R10 before and after the SWAPP instruction. The input value is 'R10 0000000011111111', with the first 8 bits (00000000) and the last 8 bits (11111111) grouped. Arrows indicate the swap of these two groups. The resulting value is 'R10 1111111100000000', where the first 8 bits are now 11111111 and the last 8 bits are 00000000.</p>										

6.5 Istruzioni di salto di programma

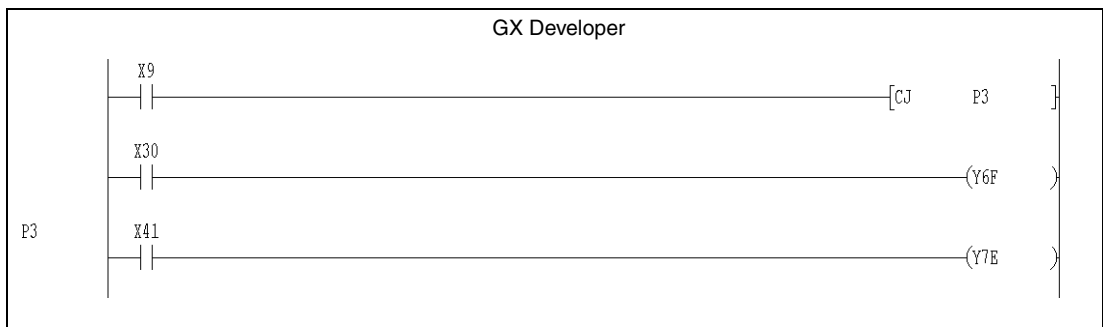
Le istruzioni di salto comprendono una destinazione del salto.

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Salto condizionale	CJ	CJ_M
Salto condizionale dalla scansione successiva	SCJ	SCJ_M
Salto	JMP	JMP_M
Salto a fine programma	GOEND	GOEND_M

La destinazione di un salto viene determinata da un Pointer (puntatore) (GX Developer) o da una Label (etichetta) (GX IEC Developer).

Per i dettagli relativi alla programmazione di una Label, vedi il Manuale di programmazione di GX IEC Developer.

GX Developer		GX IEC Developer	
LD	X9	LD	X9
CJ	P3	JMPC	Label_3
LD	X30	LD	X30
OUT	Y6F	ST	Y6F
P3		Label_3:	LD
LD	X41	ST	X41
OUT	Y7E		Y7E



6.5.1 CJ, SCJ, JMP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Rapporto errore	Segnal e di errore						
Operandi a bit					Operandi a word (16-bit)					Costanti		Punta-tore		Livello											
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H (16#)	P	I	N				M9012	M9010 M9011
p																		●				3 1	●		●

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale di programmazione per il corrispondente numero di passi.

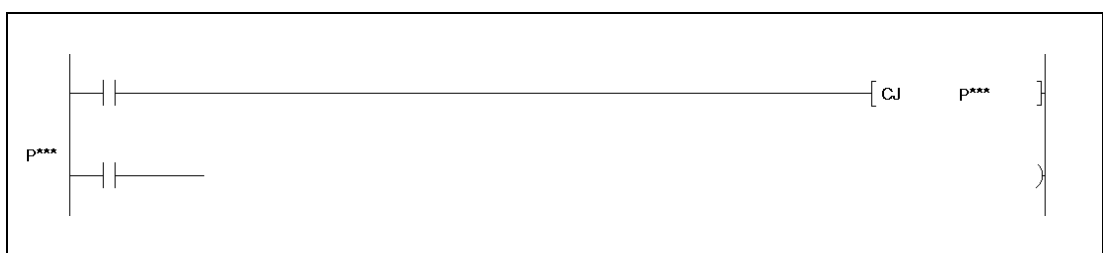
Operandi MELSEC Q

Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word				P			
p	—	—	—	—	—	—	—	●		SM0	2

GX IEC Developer

<p>Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> MELSEC CJ Label </div> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin-top: 5px;"> Label: MELSEC </div>	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> JMPC Label </div> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin-top: 5px;"> Label: </div>
---	--------------------------	--

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
p	Destinazione del salto	Puntatore/Label

Funzioni Istruzioni di salto

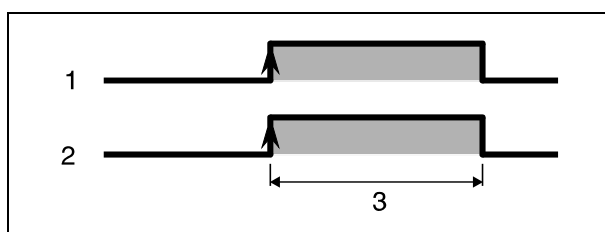
Una istruzione di salto è composta da un comando di salto CJ, SCJ o JMP (**C**onditional **J**ump = salto condizionale, **JuMP** = salto) e da un puntatore (o etichetta) P, che indica la destinazione del salto.

Nella serie A, il numero del puntatore (Label) può variare fra P(Label)0 e P(Label)255. P(Label)255 ha lo stesso significato di una istruzione END e non può essere usata come destinazione di un salto.

Nella serie Q e System Q, il numero di puntatore (Label) può variare fra P(Label)0 e P(Label)4095. Qualsiasi valore P(Label)xx può essere liberamente usato nel programma.

CJ Salto condizionale

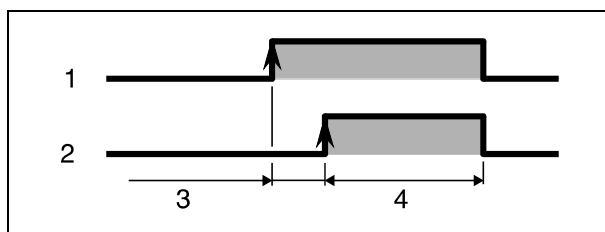
Se la condizione di ingresso è vera, l'istruzione CJ salta per eseguire al parte di programma corrispondente alla destinazione del salto. Se la condizione di ingresso non è vera, viene eseguito il passo di programma successivo.



- ¹ Condizione d'ingresso
- ² Istruzione CJ
- ³ Eseguita ad ogni scansione

SCJ Salto condizionale dalla scansione programma successiva

Se la condizione di ingresso è vera, l'istruzione SCJ salta per eseguire al parte di programma corrispondente alla destinazione del salto, a partire dalla scansione successiva. Se la condizione di ingresso non è vera, viene eseguito il passo di programma successivo.



- ¹ Condizione d'ingresso
- ² Istruzione SCJ
- ³ Una scansione
- ⁴ Eseguita ad ogni scansione

JMP Istruzione di salto

L'istruzione di salto esegue la parte di programma indicata dalla destinazione del salto, indipendentemente dalla condizione di ingresso (salto non condizionale).

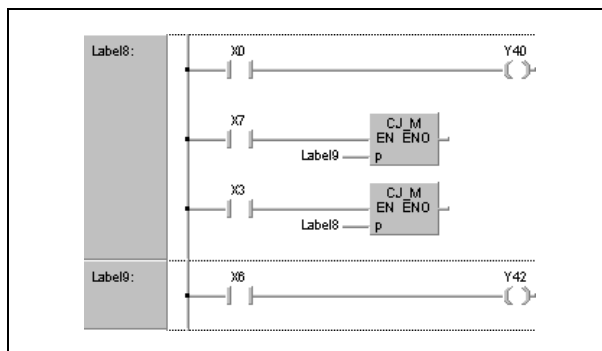
NOTA

Un timer che viene scavalcato da una istruzione CJ, SCJ o JMP continua a gestire con precisione la propria temporizzazione.

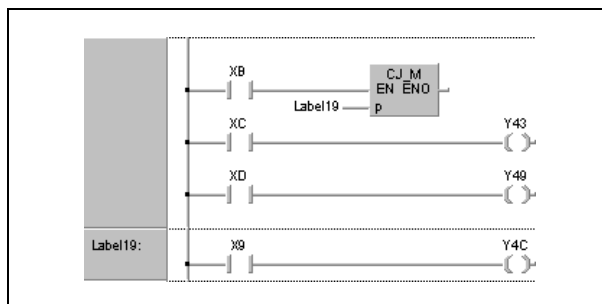
Se una istruzione OUT viene scavalcata da una istruzione di salto, la condizione dell'uscita rimane invariata.

L'esecuzione di una istruzione di salto accorcia il tempo di scansione di un programma, proporzionalmente al numero di passi che vengono saltati (vedi tabelle in appendice).

Le istruzioni CJ, SCJ e JMP possono saltare all'indietro, su destinazioni precedenti a quella contenente l'istruzione di salto. Tuttavia è necessario che il programma esca dal loop prima dell'intervento del timer di watchdog (il programma di esempio seguente esce dal loop quando X7 diventa 1).

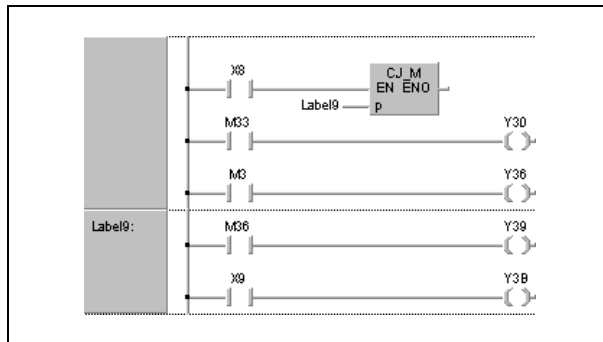


La condizione di un operando scavalcato da una istruzione di salto rimane invariata. Questo è mostrato dall'esempio che segue:



Quando XB diventa 1, questo programma salta alla destinazione Label19. Le condizioni delle uscite Y43 e Y49 restano invariate, anche se XC e XD vengono messi a 1 o a 0.

La destinazione di programma Label9 occupa un passo di programma.



Le istruzioni CJ, SCJ o JMP possono saltare solo su destinazioni contenute all'interno del programma che le contiene.

Se una destinazione di salto si trova all'interno di una sezione esclusa dalla scansione (salto di parti di programma), l'esecuzione del programma riprende dalla prima istruzione utile, dopo la destinazione del salto.

Errori di esecuzione

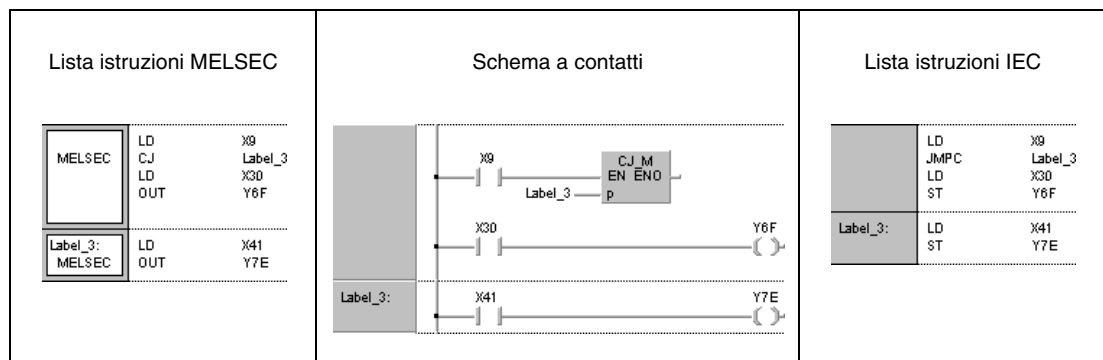
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Si è utilizzato un puntatore definito due volte per indicare la destinazione del salto. (serie Q e System Q = codice di errore 4210).
- La destinazione del salto non è definita nel programma (destinazione o puntatore mancanti) (serie Q e System Q = codice di errore 4210).
- La destinazione del salto è oltre l'istruzione END. (serie Q e System Q = codice di errore 4210).

Programma di esempio 1

CJ

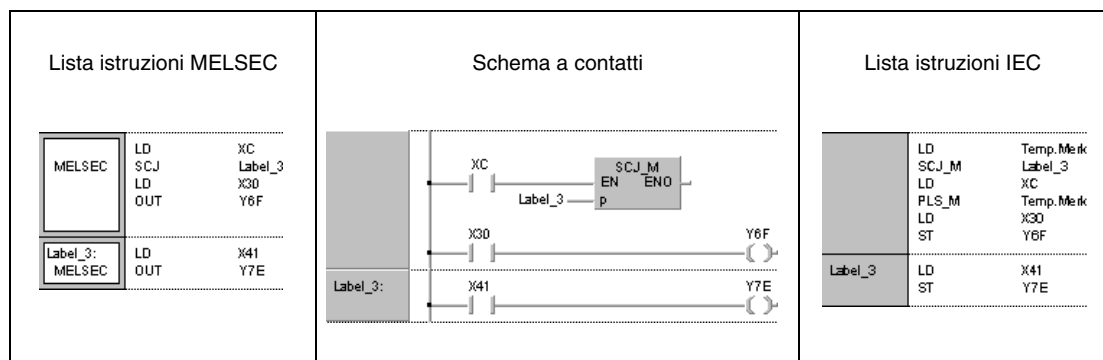
Il programma seguente salta alla destinazione Label_3 se X9 = 1.



Programma di esempio 2

SCJ

Il programma seguente salta alla destinazione Label_3 dalla scansione successiva, se XC = 1.



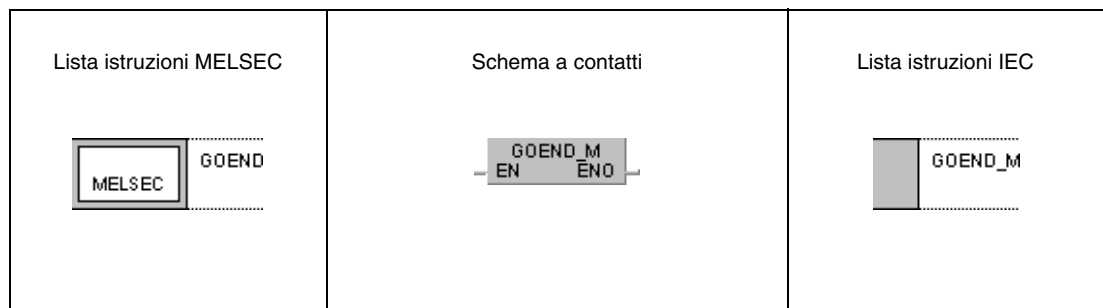
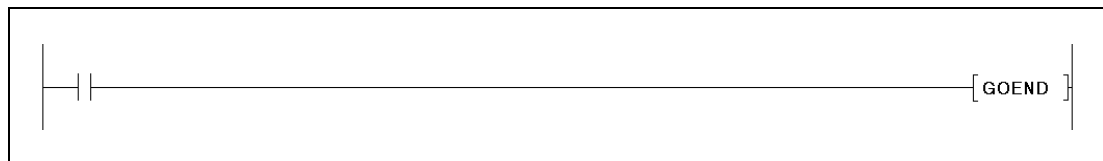
6.5.2 GOEND

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi
MELSEC Q

Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word							
—	—	—	—	—	—	—	—	—	—	SM0	1

GX IEC
DeveloperGX
Developer

Variabili

Dati impostati	Significato	Tipo dati
—	—	—

Funzioni **GOEND Salto a fine programma**

La destinazione di salto dell'istruzione GOEND è l'istruzione FEND o END del programma.

Errori di esecuzione

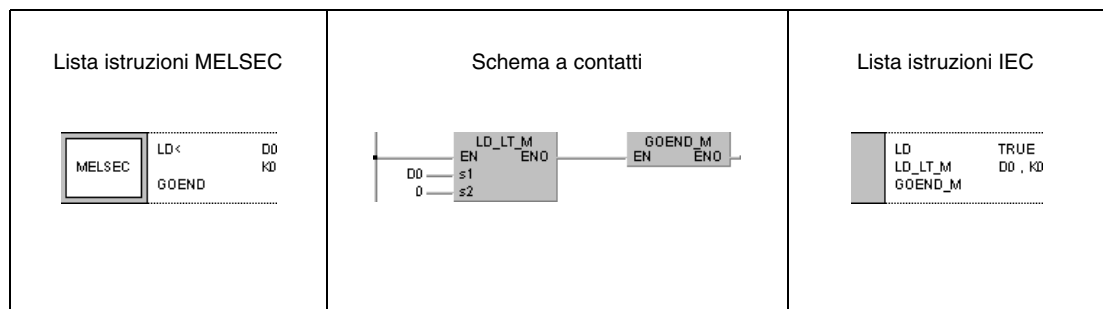
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- L'istruzione GOEND viene eseguita dopo una istruzione CALL o ECALL e prima di una istruzione RET (codice di errore: 4211).
- L'istruzione GOEND viene eseguita dopo una istruzione FOR e prima di una istruzione NEXT (codice di errore: 4200).
- L'istruzione GOEND viene eseguita durante un programma di interruzione, ma prima dell'istruzione IRET (codice di errore: 4221).
- L'istruzione GOEND viene eseguita fra le istruzioni CHKCIR e CHKEND (codice di errore: 4230).
- L'istruzione GOEND viene eseguita fra le istruzione IX e IXEND (codice di errore: 4231).

Esempio di programma

GOEND

Il programma seguente salta all'istruzione END quando il dato in D0 è negativo.



6.6 Istruzioni per controllo esecuzione del programma

Le istruzioni per controllo dell'esecuzione del programma consentono di invocare routine di interruzione. Le interruzioni possono essere abilitate o disabilitate complessivamente o tramite maschere di bit.

La tabella che segue offre una panoramica di queste istruzioni:

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Interruzioni disabilitate	DI	DI_M
Interruzioni abilitate	EI	EI_M
Maschera di bit delle condizioni di esecuzione dei programmi di interruzione	IMASK	IMASK_M
Fine programma di interruzione	IRET	IRET_M

6.6.1 DI, EI, IMASK

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
● ¹	● ¹	●	●	●	●

¹ Con le CPU AnN o AnS, le istruzioni DI/EI possono essere eseguite solo se il relé interno M9053 non è eccitato (0).

Operandi MELSEC A

Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Rapporto errore	Segnal e di errore						
Operandi a bit					Operandi a word (16-bit)					Costanti	Punta-tore		Livello												
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H (16#)	P	I	N			M9012	M9010 M9011	
																						1	1		

¹ Solo istruzioni DI e EI

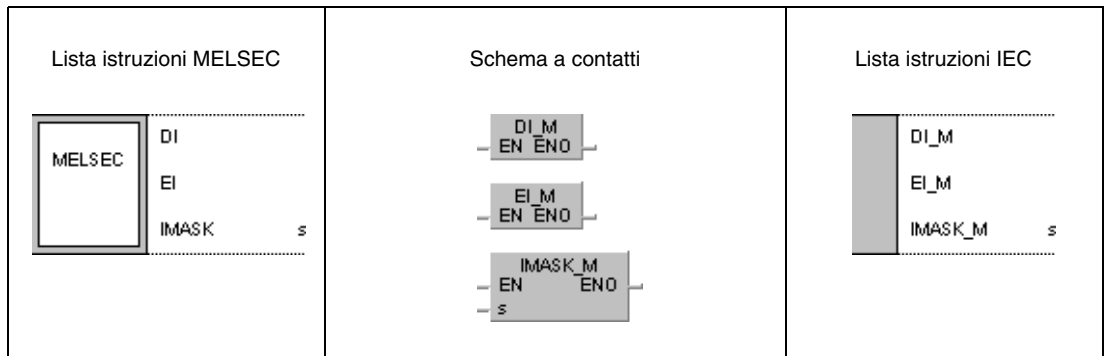
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	—	● ¹	● ¹	—	● ¹	● ¹	—	—	—	—	2 ● ¹
—	—	—	—	—	—	—	—	—	—	—	1 ● ²

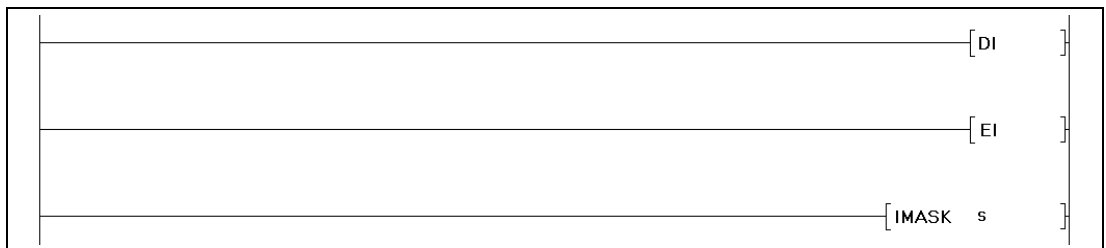
¹ Solo istruzione IMASK

² Solo istruzioni DI e EI

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Maschera di bit contenente le condizioni di esecuzione dello interruzioni, o numero dell'operando che contiene la maschera.	BIN 16-bit

Funzioni Istruzioni controllo interruzioni

Un programma di interruzione è una parte di programma (indicata dall'indirizzo di interruzione lxx) che viene invocata da un segnale esterno di interruzione. Il programma di interruzione viene eseguito in funzione delle istruzioni EI/DI. In una CPU AnN il significato delle istruzioni EI/DI dipende a sua volta dallo stato del relé interno M9053. Le istruzioni determinano le condizioni di esecuzione del programma di interruzione solo se il relé è diseccitato. Se il relé interno M9053 é eccitato, le istruzioni vengono utilizzate per il rinfresco di un collegamento (vedi "Istruzioni di rinfresco").

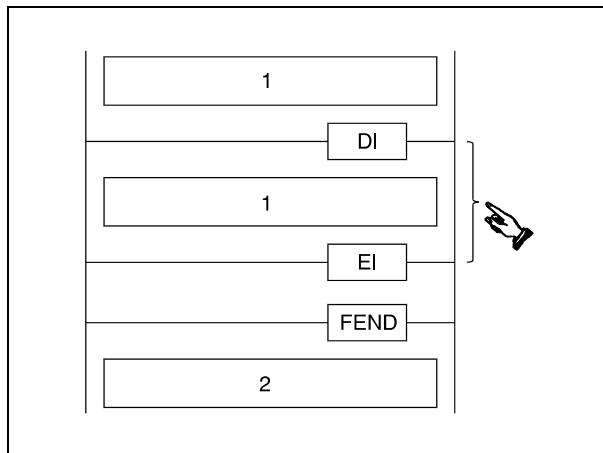
DI Disabilita interruzioni

L'istruzione DI disabilita l'esecuzione del rinfresco di un programma di interruzione fino all'esecuzione di una istruzione EI. Dolo l'accensione o un reset della CPU, lo stato di DI è attivo.

EI Abilita interruzioni

L'istruzione EI abilita l'invocazione del programma di interruzione designato dall'indirizzo di interruzione lxx, oppure abilita l'esecuzione di una istruzione IMASK.

Anche se la condizione di interruzione si verifica fra le istruzioni DI ed EI, il programma di interruzione viene sospeso fino a quando l'intera sezione fra DI e EI non è stata eseguita completamente. Lo schema seguente mostra il funzionamento in GX Developer.



¹ Programma sequenza

² Programma di interruzione

NOTA

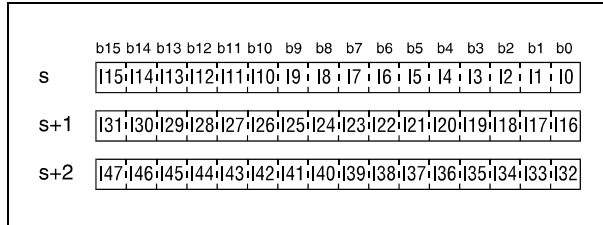
GX Developer inserisce automaticamente l'istruzione FEND. L'evento lxx deve essere allocato ad un task.

IMASK Maschera di bit delle condizioni di esecuzione dei programmi di interruzione (serie Q e System Q)

Nella maschera di bit indicata da s, a ciascun bit è collegato un indirizzo di interruzione. Lo stato di ciascun bit determina se l'interruzione corrispondente può essere eseguita. Se il bit è zero, il programma di interruzione non può essere eseguito. Se il bit è 1, il programma di interruzione verrà essere eseguito.

QnACPU

La tabella seguente mostra l'allocazione dei bit da s a s+2 con i corrispondenti indirizzi di interruzione:

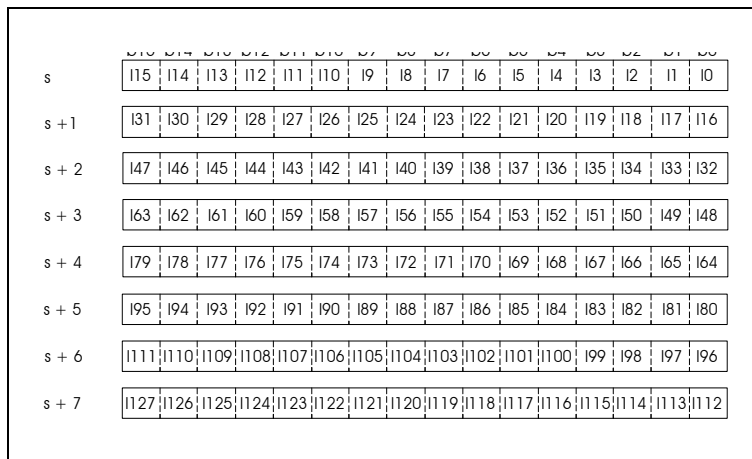


Dopo l'accensione o il reset della CPU con il commutatore RUN/STOP, i bit da b0 a b31 (indirizzi di interruzione da I0 a I31) sono attivi (1), cioè i corrispondenti programmi di interruzione possono essere eseguiti. I bit da b32 a b47 (indirizzi di interruzione da I32 a I47) sono azzerati, cioè i corrispondenti programmi di interruzione non possono essere eseguiti.

Le maschere contenute da s a s+2 vengono memorizzate nei registri speciali da SD715 a SD717.

System Q CPU (singolo processore)

L'allocazione dei bit da s a s+7 con i corrispondenti indirizzi di interruzione viene mostrata di seguito:



Dopo l'accensione o il reset della CPU con il commutatore RUN/STOP, l'esecuzione dei programmi di interruzione da I0 a I31 è abilitata.

Le maschere contenute da s a s+2 vengono memorizzate nei registri speciali da SD715 a SD717. Le maschere contenute da s+3 a s+7 vengono memorizzate nei registri speciali da SD781 a SD785.

Le maschere vengono definite consecutivamente da s a s+7, anche se vengono memorizzate in due serie separate di registri speciali (da SD715 a SD717 e da SD781 a SD785).

**System Q
CPU (multi
processore)**

L'allocazione dei bit da s a s+15 con i corrispondenti indirizzi di interruzione viene mostrata di seguito:

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
s	I15	I14	I13	I12	I11	I10	I9	I8	I7	I6	I5	I4	I3	I2	I1	I0
s + 1	I31	I30	I29	I28	I27	I26	I25	I24	I23	I22	I21	I20	I19	I18	I17	I16
s + 2	I47	I46	I45	I44	I43	I42	I41	I40	I39	I38	I37	I36	I35	I34	I33	I32
s + 3	I63	I62	I61	I60	I59	I58	I57	I56	I55	I54	I53	I52	I51	I50	I49	I48
s + 4	I79	I78	I77	I76	I75	I74	I73	I72	I71	I70	I69	I68	I67	I66	I65	I64
s + 5	I95	I94	I93	I92	I91	I90	I89	I88	I87	I86	I85	I84	I83	I82	I81	I80
s + 6	I111	I110	I109	I108	I107	I106	I105	I104	I103	I102	I101	I100	I99	I98	I97	I96
s + 7	I127	I126	I125	I124	I123	I122	I121	I120	I119	I118	I117	I116	I115	I114	I113	I112
s + 8	I143	I142	I141	I140	I139	I138	I137	I136	I135	I134	I133	I132	I131	I130	I129	I128
s + 9	I159	I158	I157	I156	I155	I154	I153	I152	I151	I150	I149	I148	I147	I146	I145	I144
s + 10	I175	I174	I173	I172	I171	I170	I169	I168	I167	I166	I165	I164	I163	I162	I161	I160
s + 11	I191	I190	I189	I188	I187	I186	I185	I184	I183	I182	I181	I180	I179	I178	I177	I176
s + 12	I207	I206	I205	I204	I203	I202	I201	I200	I199	I198	I197	I196	I195	I194	I193	I192
s + 13	I223	I222	I221	I220	I219	I218	I217	I216	I215	I214	I213	I212	I211	I210	I209	I208
s + 14	I239	I238	I237	I236	I235	I234	I233	I232	I231	I230	I229	I228	I227	I226	I225	I224
s + 15	I255	I254	I253	I252	I251	I250	I249	I248	I247	I246	I245	I244	I243	I242	I241	I240

Dopo l'accensione o il reset della CPU con il commutatore RUN/STOP, l'esecuzione dei programmi di interruzione da I0 a I31 è abilitata. L'esecuzione dei programmi di interruzione da I32 a I255 è disabilitata.

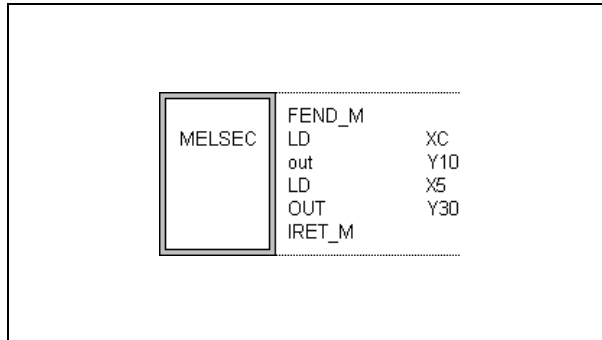
Le maschere contenute da s a s+2 vengono memorizzate nei registri speciali da SD715 a SD717. Le maschere contenute da s+3 a s+15 vengono memorizzate nei registri speciali da SD781 a SD793.

Le maschere vengono definite consecutivamente da s a s+15, anche se vengono memorizzate in due serie separate di registri speciali (da SD715 a SD717 e da SD781 a SD793).

NOTA

Se è necessario usare un contatore all'interno di un programma di interruzione, sono disponibili dei contatori speciali per questo scopo. Le CPU tipo A3H, A3M, AnA, AnAS e AnU non contengono questi contatori speciali.

L'indirizzo di interruzione (pointer di interruzione) che indica il programma di interruzione, occupa un passo.



Con GX Developer o con GX IEC Developer in modo MELSEC, le istruzioni FEDN e IRET devono essere programmate esplicitamente.

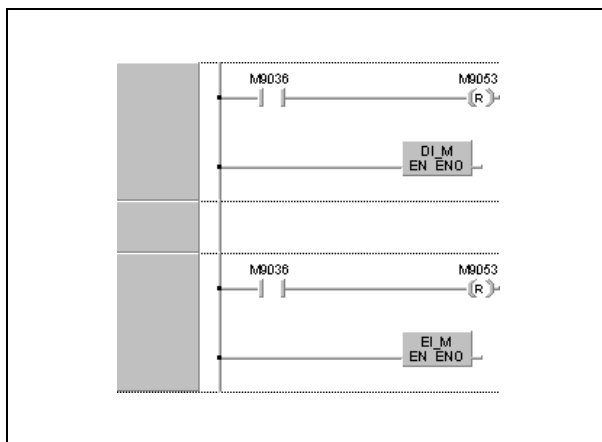
In alternativa all'editor MELSEC, è possibile utilizzare l'editor IEC. L'interruzione viene attribuita ad un task e le istruzioni FEND e IRET vengono inserite automaticamente dal compilatore MEDOC di GX IEC Developer (vedi programma di esempio).

La descrizione delle condizioni di interruzione è contenuta in una diversa sezione di questo manuale.

Durante l'esecuzione di un programma di interruzione, lo stato DI viene attivato internamente, in modo che nessun altro programma di interruzione possa essere attivato contemporaneamente. Un altro programma di interruzione può essere invocato solo dopo la programmazione di una istruzione EI.

Se una istruzione EI o DI è inserita all'interno di una istruzione MC, l'istruzione EI o DI viene eseguita indipendentemente dall'istruzione MC.

Con le CPU AnN o AnS, le istruzioni DI o EI possono essere eseguite solo se il relé interno M9053 non è eccitato (0). Se il relé interno è eccitato, le istruzioni EI o DI si riferiscono alle condizioni di esecuzione di un rinfresco di collegamento. In una CPU AnN o AnS, il relé interno M9053 deve essere azzerato prima di poter elaborare una istruzione EI o DI per il controllo del richiamo di programmi di interruzione.



Esempio di programma EI, DI, IMASK (GX IEC Developer)

Il programma seguente abilita l'esecuzione di un programma di interruzione se X0 è attivo (1). Se X0 non è attivo (0), l'esecuzione del programma di interruzione è disabilitata.

La figura in basso mostra la programmazione dei task in modo IEC. I task invocano i programmi di interruzione I1 e I2.

Interrupt_1 (I1) e Interrupt_2 (I2) sono programmi di interruzione. Non è necessario programmare esplicitamente l'istruzione IRET, inserita automaticamente dal compilatore di GX IEC Developer.

<p style="text-align: center;">Lista istruzioni MELSEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="width: 10%;"></td><td style="width: 10%;">LD</td><td style="width: 10%;">X0</td><td style="width: 10%;"></td><td style="width: 10%;"></td><td style="width: 10%;">Label_1</td></tr> <tr><td>MELSEC</td><td>CJ</td><td></td><td></td><td></td><td></td></tr> <tr><td></td><td>DI</td><td></td><td></td><td></td><td></td></tr> <tr><td colspan="6"> </td></tr> <tr><td>Label_1:</td><td>LD</td><td>X0</td><td></td><td></td><td></td></tr> <tr><td>MELSEC</td><td>CJ</td><td></td><td></td><td></td><td>Label_2</td></tr> <tr><td></td><td>LD</td><td>X0</td><td></td><td></td><td></td></tr> <tr><td></td><td>MOVP</td><td>H6</td><td></td><td></td><td>D10</td></tr> <tr><td></td><td>MOVP</td><td>H0</td><td></td><td></td><td>D11</td></tr> <tr><td></td><td>MOVP</td><td>H0</td><td></td><td></td><td>D12</td></tr> <tr><td></td><td>IMASK</td><td></td><td></td><td></td><td>D10</td></tr> <tr><td></td><td>EI</td><td></td><td></td><td></td><td></td></tr> <tr><td colspan="6"> </td></tr> <tr><td>Label_2:</td><td>LD</td><td>M0</td><td></td><td></td><td></td></tr> <tr><td>MELSEC</td><td>OUT</td><td></td><td></td><td></td><td>Y20</td></tr> </table>		LD	X0			Label_1	MELSEC	CJ						DI											Label_1:	LD	X0				MELSEC	CJ				Label_2		LD	X0					MOVP	H6			D10		MOVP	H0			D11		MOVP	H0			D12		IMASK				D10		EI											Label_2:	LD	M0				MELSEC	OUT				Y20	<p style="text-align: center;">Schema a contatti</p>																																																																			
	LD	X0			Label_1																																																																																																																																																									
MELSEC	CJ																																																																																																																																																													
	DI																																																																																																																																																													
Label_1:	LD	X0																																																																																																																																																												
MELSEC	CJ				Label_2																																																																																																																																																									
	LD	X0																																																																																																																																																												
	MOVP	H6			D10																																																																																																																																																									
	MOVP	H0			D11																																																																																																																																																									
	MOVP	H0			D12																																																																																																																																																									
	IMASK				D10																																																																																																																																																									
	EI																																																																																																																																																													
Label_2:	LD	M0																																																																																																																																																												
MELSEC	OUT				Y20																																																																																																																																																									
<p style="text-align: center;">Lista istruzioni IEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="width: 10%;"></td><td style="width: 10%;">LD</td><td style="width: 10%;">X0</td><td style="width: 10%;"></td><td style="width: 10%;"></td><td style="width: 10%;">Label_1</td></tr> <tr><td></td><td>JMPC</td><td></td><td></td><td></td><td>Label_1</td></tr> <tr><td></td><td>DI_M</td><td></td><td></td><td></td><td></td></tr> <tr><td colspan="6"> </td></tr> <tr><td>Label_1:</td><td>LD</td><td>X0</td><td></td><td></td><td></td></tr> <tr><td></td><td>JMPC</td><td></td><td></td><td></td><td>Label_2</td></tr> <tr><td></td><td>LD</td><td>X0</td><td></td><td></td><td></td></tr> <tr><td></td><td>MOVP_M</td><td>H6</td><td></td><td></td><td>var_D10[10]</td></tr> <tr><td></td><td>MOVP_M</td><td>H0</td><td></td><td></td><td>var_D10[11]</td></tr> <tr><td></td><td>MOVP_M</td><td>H0</td><td></td><td></td><td>var_D10[12]</td></tr> <tr><td></td><td>IMASK_M</td><td></td><td></td><td></td><td>var_D10</td></tr> <tr><td></td><td>EI_M</td><td></td><td></td><td></td><td></td></tr> <tr><td colspan="6"> </td></tr> <tr><td>Label_2:</td><td>LD</td><td>M0</td><td></td><td></td><td></td></tr> <tr><td></td><td>ST</td><td></td><td></td><td></td><td>Y20</td></tr> </table>		LD	X0			Label_1		JMPC				Label_1		DI_M											Label_1:	LD	X0					JMPC				Label_2		LD	X0					MOVP_M	H6			var_D10[10]		MOVP_M	H0			var_D10[11]		MOVP_M	H0			var_D10[12]		IMASK_M				var_D10		EI_M											Label_2:	LD	M0					ST				Y20	<p style="text-align: center;">Task Information</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td colspan="2" style="text-align: center;">Task Attributes</td><td style="text-align: center;">OK</td></tr> <tr><td>Event:</td><td>I1</td><td style="text-align: center;">Cancel</td></tr> <tr><td>Interval:</td><td>0</td><td style="text-align: center;">Comment</td></tr> <tr><td>Priority:</td><td>31</td><td></td></tr> <tr><td colspan="3">Name: Interrupt_1</td></tr> <tr><td colspan="3">Size: 87 Bytes</td></tr> <tr><td>Type:</td><td>TASK</td><td><input checked="" type="checkbox"/> Timer/ Output Control</td></tr> <tr><td colspan="3">Last Change: 06.10.1997 14:59:36</td></tr> <tr><td colspan="3">Security Level</td></tr> <tr><td colspan="3"> <input checked="" type="radio"/> 0 <input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5 <input type="radio"/> 6 <input type="radio"/> 7 </td></tr> <tr><td colspan="3"><input checked="" type="checkbox"/> Allow Read Access for lower Levels</td></tr> </table>	Task Attributes		OK	Event:	I1	Cancel	Interval:	0	Comment	Priority:	31		Name: Interrupt_1			Size: 87 Bytes			Type:	TASK	<input checked="" type="checkbox"/> Timer/ Output Control	Last Change: 06.10.1997 14:59:36			Security Level			<input checked="" type="radio"/> 0 <input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5 <input type="radio"/> 6 <input type="radio"/> 7			<input checked="" type="checkbox"/> Allow Read Access for lower Levels			<p style="text-align: center;">Task Information</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td colspan="2" style="text-align: center;">Task Attributes</td><td style="text-align: center;">OK</td></tr> <tr><td>Event:</td><td>I2</td><td style="text-align: center;">Cancel</td></tr> <tr><td>Interval:</td><td>0</td><td style="text-align: center;">Comment</td></tr> <tr><td>Priority:</td><td>31</td><td></td></tr> <tr><td colspan="3">Name: Interrupt_2</td></tr> <tr><td colspan="3">Size: 87 Bytes</td></tr> <tr><td>Type:</td><td>TASK</td><td><input checked="" type="checkbox"/> Timer/ Output Control</td></tr> <tr><td colspan="3">Last Change: 06.10.1997 15:00:05</td></tr> <tr><td colspan="3">Security Level</td></tr> <tr><td colspan="3"> <input checked="" type="radio"/> 0 <input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5 <input type="radio"/> 6 <input type="radio"/> 7 </td></tr> <tr><td colspan="3"><input checked="" type="checkbox"/> Allow Read Access for lower Levels</td></tr> </table>	Task Attributes		OK	Event:	I2	Cancel	Interval:	0	Comment	Priority:	31		Name: Interrupt_2			Size: 87 Bytes			Type:	TASK	<input checked="" type="checkbox"/> Timer/ Output Control	Last Change: 06.10.1997 15:00:05			Security Level			<input checked="" type="radio"/> 0 <input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5 <input type="radio"/> 6 <input type="radio"/> 7			<input checked="" type="checkbox"/> Allow Read Access for lower Levels		
	LD	X0			Label_1																																																																																																																																																									
	JMPC				Label_1																																																																																																																																																									
	DI_M																																																																																																																																																													
Label_1:	LD	X0																																																																																																																																																												
	JMPC				Label_2																																																																																																																																																									
	LD	X0																																																																																																																																																												
	MOVP_M	H6			var_D10[10]																																																																																																																																																									
	MOVP_M	H0			var_D10[11]																																																																																																																																																									
	MOVP_M	H0			var_D10[12]																																																																																																																																																									
	IMASK_M				var_D10																																																																																																																																																									
	EI_M																																																																																																																																																													
Label_2:	LD	M0																																																																																																																																																												
	ST				Y20																																																																																																																																																									
Task Attributes		OK																																																																																																																																																												
Event:	I1	Cancel																																																																																																																																																												
Interval:	0	Comment																																																																																																																																																												
Priority:	31																																																																																																																																																													
Name: Interrupt_1																																																																																																																																																														
Size: 87 Bytes																																																																																																																																																														
Type:	TASK	<input checked="" type="checkbox"/> Timer/ Output Control																																																																																																																																																												
Last Change: 06.10.1997 14:59:36																																																																																																																																																														
Security Level																																																																																																																																																														
<input checked="" type="radio"/> 0 <input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5 <input type="radio"/> 6 <input type="radio"/> 7																																																																																																																																																														
<input checked="" type="checkbox"/> Allow Read Access for lower Levels																																																																																																																																																														
Task Attributes		OK																																																																																																																																																												
Event:	I2	Cancel																																																																																																																																																												
Interval:	0	Comment																																																																																																																																																												
Priority:	31																																																																																																																																																													
Name: Interrupt_2																																																																																																																																																														
Size: 87 Bytes																																																																																																																																																														
Type:	TASK	<input checked="" type="checkbox"/> Timer/ Output Control																																																																																																																																																												
Last Change: 06.10.1997 15:00:05																																																																																																																																																														
Security Level																																																																																																																																																														
<input checked="" type="radio"/> 0 <input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5 <input type="radio"/> 6 <input type="radio"/> 7																																																																																																																																																														
<input checked="" type="checkbox"/> Allow Read Access for lower Levels																																																																																																																																																														

NOTA

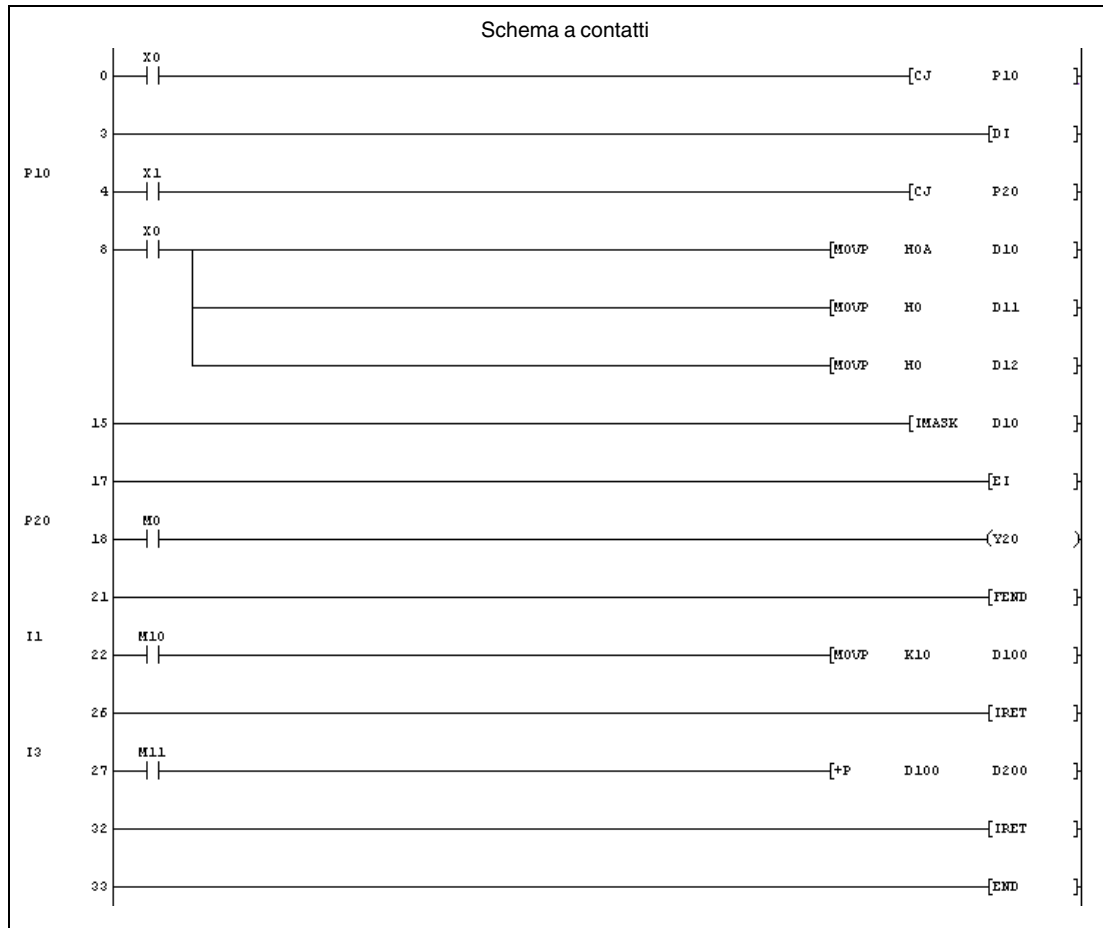
Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

Esempio di programma

EI, DI, IMASK (CPU del System Q, GX Developer)

Il programma seguente abilita l'esecuzione di un programma di interruzione se X0 è attivo (1). Se X0 non è attivo (0), l'esecuzione del programma di interruzione è disabilitata.

I1 e I3 sono programmi di interruzione.



Lista istruzioni			
0	LD	X0	
1	CJ	P10	
3	DI		
4	P10		
5	LD	X1	
6	CJ	P20	
8	LD	X0	
9	MOVP	H0A	D10
11	MOVP	H0	D11
13	MOVP	H0	D12
15	IMASK	D10	
17	EI		
18	P20		
19	LD	M0	
20	OUT	Y20	
21	FEND		
22	I1		
23	LD	M10	
24	MOVP	K10	D100
26	IRET		
27	I3		
28	LD	M11	
29	+P	D100	D200
32	IRET		
33	END		

6.6.2 IRET

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

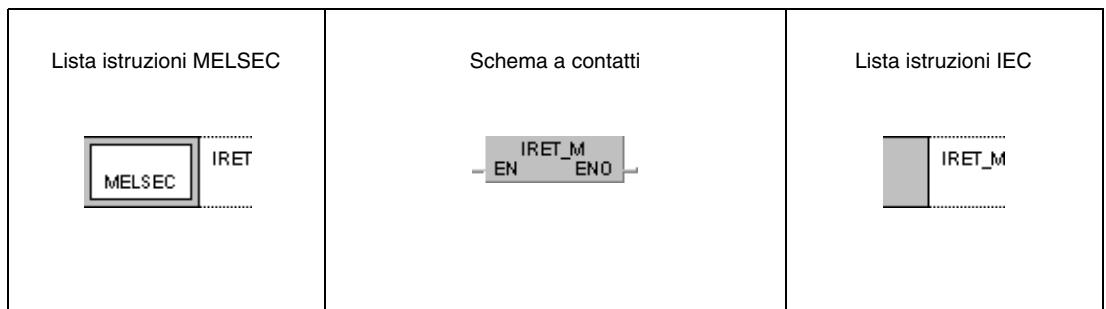
Operandi MELSEC A

Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Rapporto errore	Segnale di errore				
Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello									
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z						V	K	H (16#)	P
																					1		

Operandi MELSEC Q

Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word							
—	—	—	—	—	—	—	—	—	—	—	1

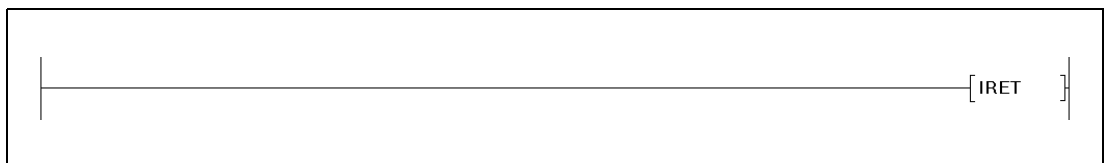
GX IEC Developer



NOTA

Usando l'editor IEC di GX IEC Developer, l'istruzione IRET viene inserita automaticamente nel programma.

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
—	—	—

Funzioni Ritorno da un programma di interruzione al programma principale**IRET Fine di un programma di interruzione**

La fine di un programma di interruzione è indicata dall'istruzione IRET.

L'elaborazione dei contatori continua durante l'interruzione.

Dopo l'esecuzione dell'istruzione IRET, si torna ad eseguire il programma principale.

I seguenti tipi di CPU non possiedono contatori speciali per interruzione: A3H, A3M, AnA, AnAS, AnU.

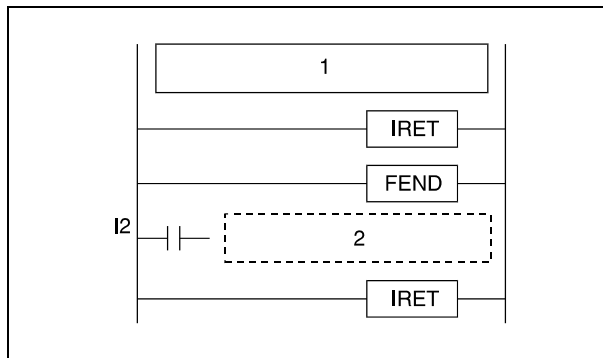
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Non esiste un indirizzo di interruzione corrispondente alla chiamata (serie Q e System Q = codice di errore 4220).
- Se l'istruzione IRET è inserita prima di un programma di interruzione, la CPU termina in quel punto l'elaborazione del programma (serie Q e System Q = codice di errore 4223).
- Istruzione END, FEND, GOEND o STOP inserita fra una chiamata a un programma di interruzione e una istruzione IRET.

NOTA

L'esempio seguente mostra un errore di programmazione!



¹ Programma sequenza

² Programma di interruzione

Esempio di programma

Per l'uso dell'istruzione IRET, vedi gli esempi per le istruzioni EI, DI e IMASK.

6.7 Istruzioni per rinfresco collegamento

Le istruzioni per rinfresco collegamento provvedono al rinfresco dei dati sulle interfacce di ingresso/uscita o delle procedure di trasferimento dati. La tabella che segue offre una panoramica di queste istruzioni:

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Rinfresco parziale I/O (serie Q e System Q)	RFS	RFS_M
	RFSP	RFSP_M
Rinfresco parziale I/O (serie A)	SEG	SEG_M
Istruzione di rinfresco per collegamento e interfaccia dati	COM	COM_M
Condizione di esecuzione o istruzione di rinfresco per collegamento e interfaccia dati	EI	EI_M
	DI	DI_M

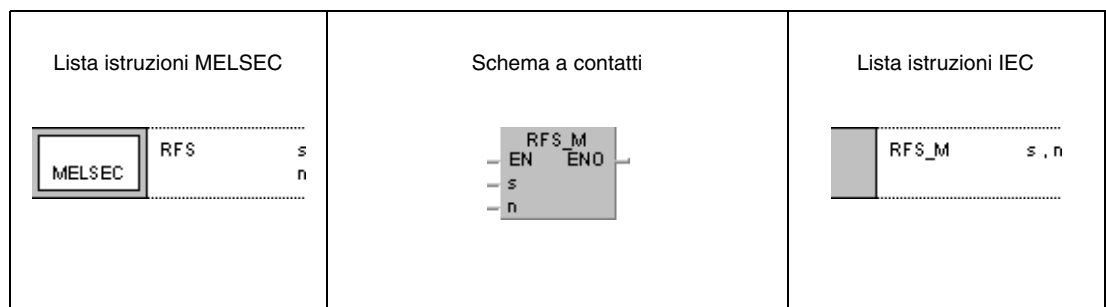
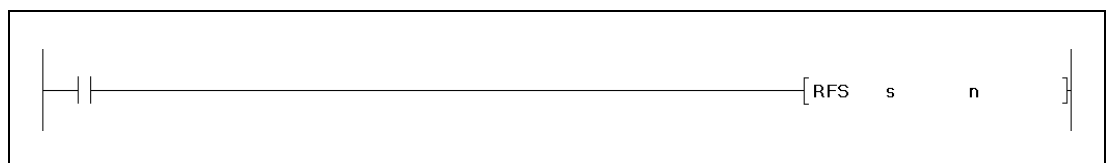
6.7.1 RFS, RFSP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi
MELSEC Q

	Operandi utilizzabili									Segnale di errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s	● ¹	—	—	—	—	—	—	—	—	SM0	3
n	●	●	●	●	●	●	●	●	—		

¹ Solo X e YGX IEC
DeveloperGX
Developer

Variabili

Dati impostati	Significato	Tipo dati
s	Primo numero dell'operando di I/O da rinfrescare	Bit
n	Primo numero dei bit di I/O da rinfrescare	BIN 16-bit

Funzioni Rinfresco parziale I/O (serie Q e System Q)

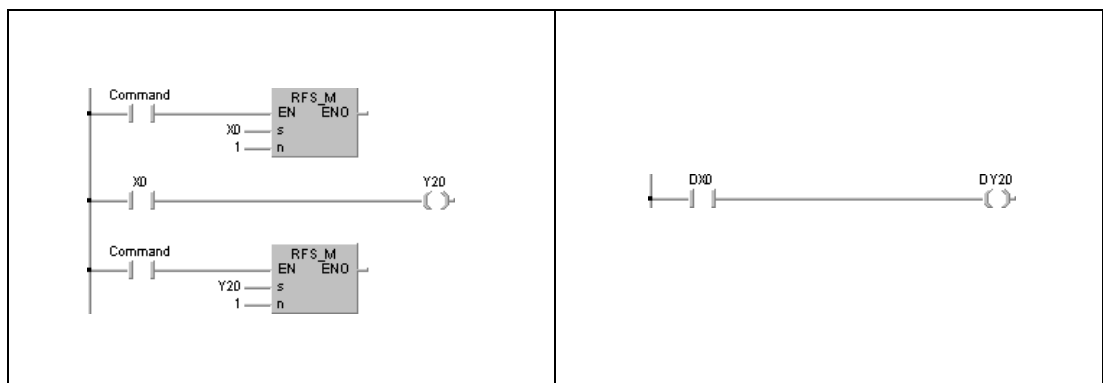
RFS Istruzione di rinfresco

L'istruzione RFS rinfresca gli ingressi e uscite nel campo indicato dei dispositivi di I/O, durante una scansione del programma. Legge dati da una sorgente esterna o scrive dati su un modulo di uscita.

I dati vengono letti in gruppo da una sorgente esterna o scritti in gruppo su un modulo esterno, dopo l'esecuzione dell'istruzione END. Quindi non è possibile emettere un segnale impulsivo per una scansione di programma.

Con l'esecuzione dell'istruzione SEG, gli indirizzi di I/O designati per gli ingressi (X) e le uscite (Y) vengono rinfrescati separatamente. In questo modo è possibile anche emettere segnali impulsivi.

Se vengono utilizzati ingressi/uscite ad accesso diretto (DX/DY), gli ingressi (X) e le uscite (Y) vengono rinfrescate bit a bit.



L'esempio sulla sinistra rinfresca l'ingresso X0 e l'uscita Y20 con una istruzione RFS.

L'esempio sulla destra esegue la stessa funzione tramite DX e DY, senza nessuna istruzione di rinfresco.

Errori di esecuzione

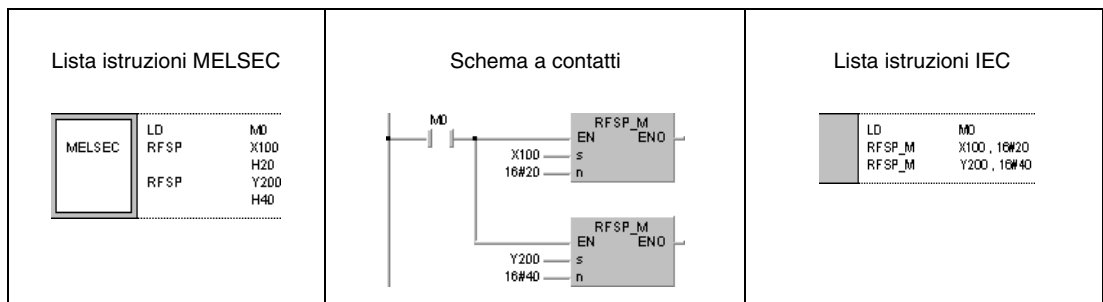
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il numero di bit indicato da n supera il campo degli operandi di ingresso/uscita.

Programma di esempio 1

RFSP

Il programma seguente rinfresca gli ingressi da X100 a X11F e le uscite da Y200 a Y23f, sul fronte positivo di M0.



6.7.2 SEG

CPU

AnS	AnN	An(S)	AnU	QnA(S), Q4AR	System Q
● ¹	● ¹	● ¹	● ¹		

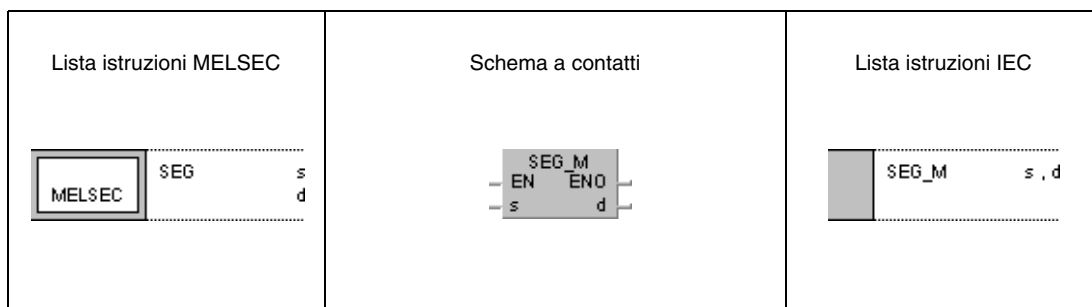
¹ Il rinfresco parziale può essere eseguito solo se il relé interno M9052 è attivo (1).

Operandi MELSEC A

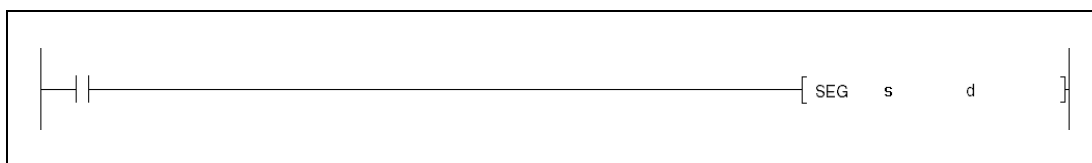
		Operandi utilizzabili														Definizione cifra	Numero di passi	Indice	Rapporto errore	Segnal e di errore								
		Operandi a bit						Operandi a word (16-bit)						Costanti							Puntatore		Livello					
		X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H (16#)	P	I	N				M9012	M9010 M9011	
s		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●					K1 ↓ K8	9	●		●
d			●	●	●	●	●	●	●	●	●	●	●	●	●	●	●								● ¹			

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale di programmazione per il corrispondente numero di passi.

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Primo numero dell'operando di I/O da rinfrescare	Bit
d	Primo numero dei bit di I/O da rinfrescare	BIN 16-bit

Funzioni Rinfresco parziale I/O (serie A)**Note generali**

Il rinfresco parziale può essere eseguito solo se il relé interno M9052 è attivo (1). Se questo relé non è attivo (0), l'istruzione SEG viene utilizzata come decodifica a 7 segmenti.

SEG Rinfresco parziale

L'istruzione SEG abilita il rinfresco di un campo definito di dispositivi di I/O, se la condizione di ingresso è soddisfatta.

L'esecuzione di un rinfresco parziale comporta l'aggiornamento dei dispositivi indicati all'interno di una scansione di programma. I segnali di ingresso continuano ad essere ricevuti e i segnali di uscita inviati sui moduli di uscita.

Un rinfresco parziale modifica lo stato di ingressi (X) e uscite (Y) per una scansione del programma, durante l'elaborazione degli I/O con rinfresco normale.

Con un semplice rinfresco di collegamento, i segnali degli ingressi e uscite vengono elaborati in gruppo, dopo l'esecuzione di una istruzione END. Non è quindi possibile emettere segnali impulsivi in una scansione di programma. Tuttavia, applicando l'istruzione SEG per un rinfresco parziale, i dispositivi di ingresso (X) e uscita (Y) indicati vengono rinfrescati separatamente. In questo modo è possibile anche emettere segnali impulsivi.

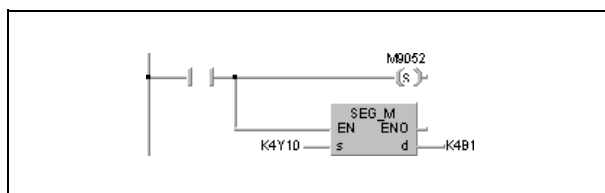
NOTA

Utilizzando una CPU A2C, non è possibile emettere segnali impulsivi per una scansione mentre si sta comunicando con gli altri moduli di I/O, anche se viene eseguito un rinfresco parziale delle uscite (Y) tramite una istruzione SEG. Per i dettagli, fare riferimento al Manuale utente della CPU A2C.

Condizioni di esecuzione**Struttura del programma**

Per prima cosa è necessario attivare il relé M9052. La sorgente dati indicata da s, seguita dall'istruzione SEG, determina il primo numero dell'operando (solo ingressi (X)/uscite (Y)) che si vuole rinfrescare. Il numero di bit di I/O viene indicato in gruppi da 8 bit ciascuno.

Lo schema seguente mostra un esempio di programmazione dell'istruzione SEG.

**Indirizzo primo operando**

Il primo indirizzo viene sempre specificato tramite l'indirizzo del primo operando di ingresso o di uscita (ad es. X0, X10, X20, ecc.).

Se viene impostato un indirizzo compreso fra Yn0 e Yn5 (oppure Xn0 e Xn7), il rinfresco inizia dall'indirizzo Yn0 (Xn0). Se viene impostato un indirizzo compreso fra Yn8 e YnF (oppure Xn8 e XnF), il rinfresco inizia dall'indirizzo Yn8 (Xn8).

Numero bit di I/O

Il numero dei bit di I/O interessati dal rinfresco può essere impostato nel campo da 8 a 2048. L'allocazione in blocchi da 8 bit avviene come segue:

B1 = 8 bit I/O

B2 = 16 bit I/O

...

BA = 80 bit I/O

BB = 88 bit I/O

...

B10 = 128 bit I/O

...

BFF = 2048 bit I/O

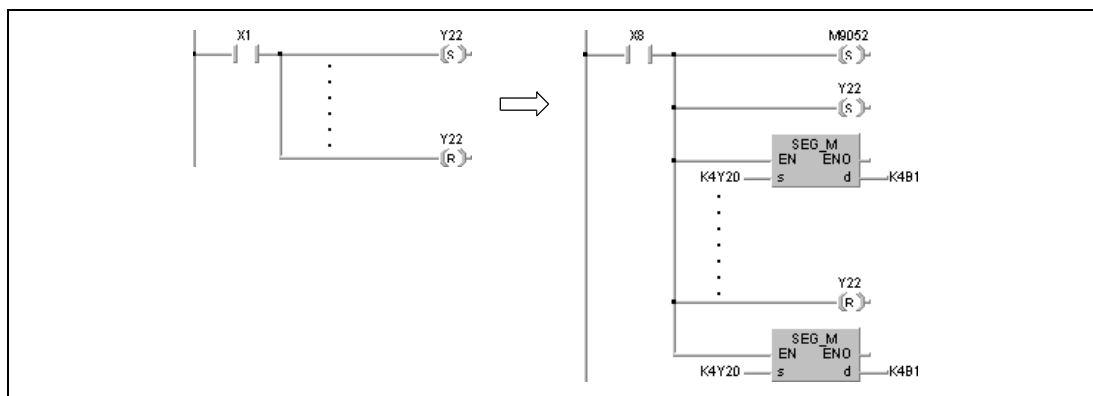
Specificando B0, vengono rinfrescati tutti i bit di I/O del PLC, iniziando dal primo.

Il rinfresco parziale viene eseguito anche se l'istruzione SEG viene eseguita nel modo diretto della CPU. Tuttavia in questo caso le condizioni delle operazioni su ingressi/uscite non vengono modificate.

Le uscite a impulso usando le istruzioni SET e RST in modo diretto dovrebbero essere modificate come mostrato di seguito, quando il controllo degli I/O viene attivato in modo rinfresco.

NOTA

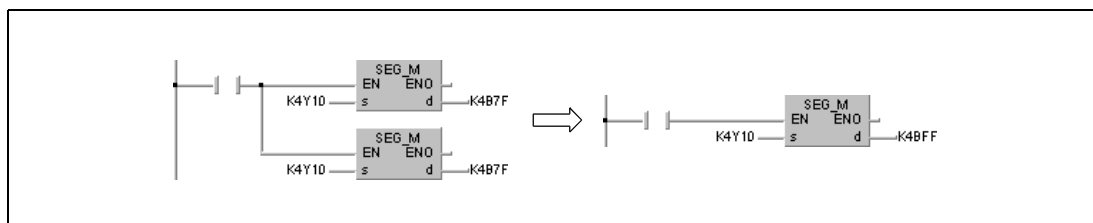
Il programma seguente non può essere elaborato da una CPU A2C.



NOTA

Usando una CPU AnA o AnU, si possono verificare errori nel rinfresco di I-O, se tutti i 2048 bit vengono rinfrescati con una istruzione SEG. In questo caso quindi, il rinfresco deve essere suddiviso in due rinfreschi parziali da 2 x 1024 operandi.

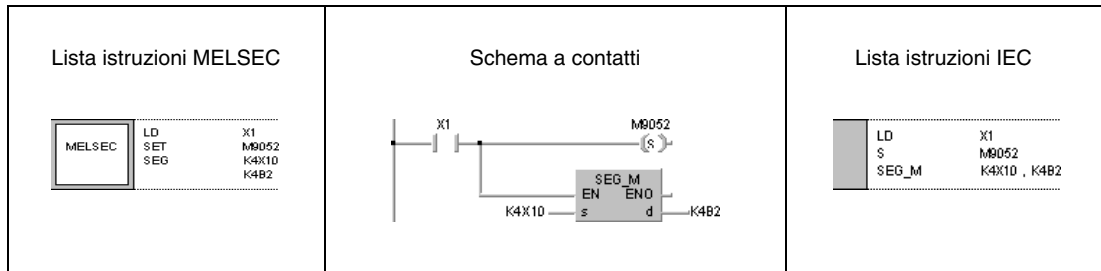
Il programma di esempio che segue mostra la suddivisione del rinfresco di 2048 dispositivi in una scansione di programma, usando una CPU AnA o AnU.



Esempio di programma

SEG

Il programma seguente rinfresca gli ingressi da X10 a X1F.



6.7.3 COM

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

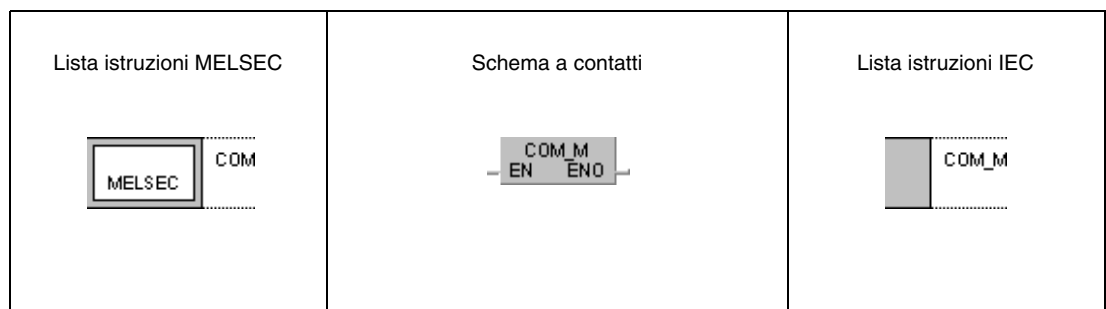
Operandi MELSEC A

Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Rapporto errore	Segnale di errore		
Operandi a bit						Operandi a word (16-bit)						Costanti		Puntatore							Livello	
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V						K	H (16#)
																	3		M9012	M9010 M9011		

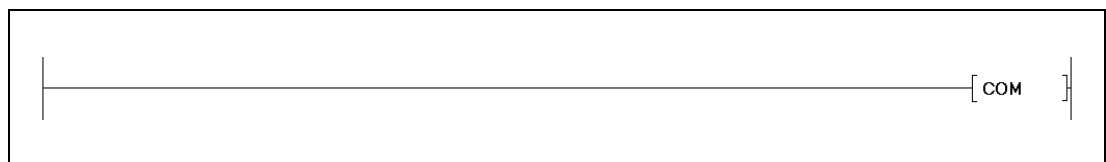
Operandi MELSEC Q

Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word							
—	—	—	—	—	—	—	—	—	—	—	1

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
—	—	—

Funzioni Rinfresco collegamento**COM Istruzione per rinfresco dati collegamento e interfaccia**

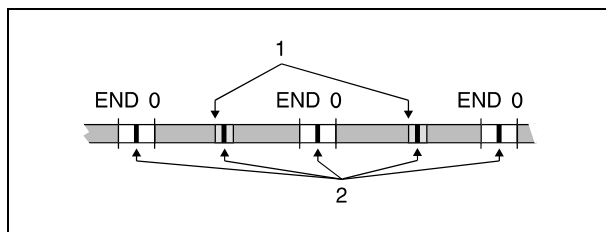
La funzione eseguita dall'istruzione COM su una CPU della serie Q o del System Q dipende dallo stato del relé speciale SM775:

- Se SM775 non è attivo (0), i dati di collegamento e interfaccia sono rinfrescati (rinfresco collegamento) mentre viene eseguita l'elaborazione dati generale (elaborazione END).
- Se SM775 è attiva (1), viene eseguita la sola elaborazione generale (elaborazione END).

Le spiegazioni seguenti si applicano solo alla serie Q/System Q con SM775 non attivo (0) ed alla serie A:

L'istruzione COM viene utilizzata per velocizzare la comunicazione dati con una stazione di I/O remoto. Se il tempo di scansione di una stazione master è più lungo di quello di una stazione locale, l'istruzione COM consente l'elaborazione corretta degli ingressi ricevuti e dei dati di uscita.

A seguito dell'esecuzione dell'istruzione COM, la CPU sospende temporaneamente il programma a sequenza, esegue l'elaborazione generale (elaborazione END), e rinfresca il collegamento e i dati di interfaccia (rinfresco collegamento).



¹ Istruzione COM

² Elaborazione dati generali/rinfresco collegamento

L'istruzione COM può essere utilizzata nel programma un qualsiasi numero di volte. Occorre tuttavia considerare che il tempo di scansione del programma viene ogni volta incrementato della durata relativa alla elaborazione dati generali ed al rinfresco del collegamento.

NOTA

L'elaborazione dati generali esegue le funzioni seguenti:

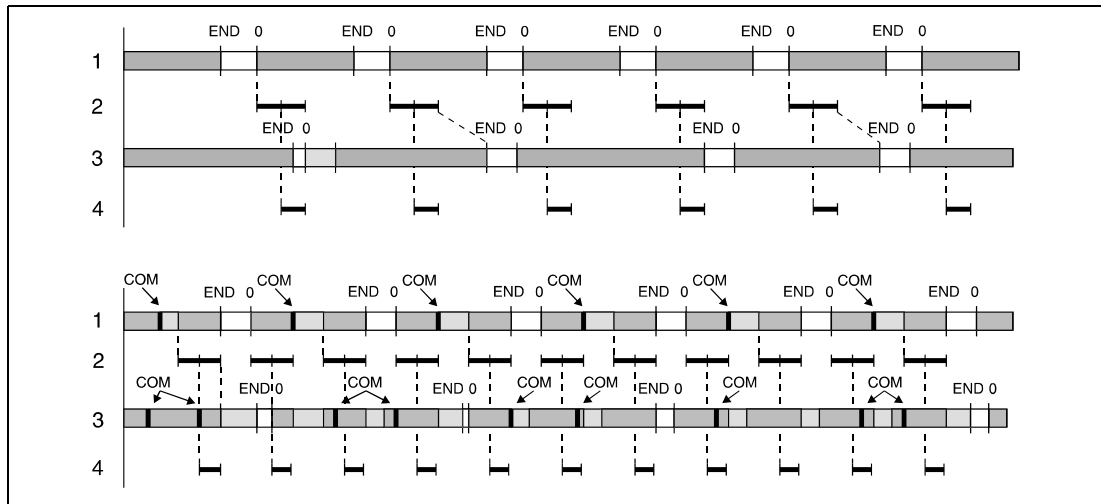
- Comunicazione fra PLC e dispositivi periferici.
- Monitoraggio di altre stazioni.
- Lettura di buffer di memoria di altri moduli funzione speciali, tramite modulo di collegamento a computer.

Il rinfresco del collegamento esegue le funzioni seguenti:

- Rinfresco del CC-Link
- Rinfresco automatico dei moduli funzione intelligenti
- Rinfresco di MELSECNET/10 e MELSECNET/H
- Rinfresco automatico della memoria condivisa multi-CPU (solo per CPU multiprocessore del System Q con versione funzionale B o successiva)

Condizioni di esecuzione

La parte superiore della figura mostra gli eventi di comunicazione dati senza istruzione COM. La parte inferiore della figura mostra gli eventi di comunicazione dati utilizzando l'istruzione COM.



- 1 Programma stazione master
- 2 Comunicazione dati
- 3 Programma stazione locale
- 4 Stazione I/O remoto, rinfresco I/O

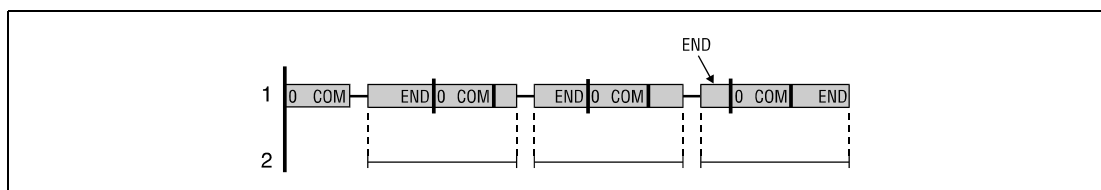
La comunicazione dati fra collegamenti viene velocizzata nel programma di sequenza della stazione master tramite l'istruzione COM, perché il numero di eventi di comunicazione con la stazione di I/O remoto aumenta.

I dati possono non essere ricevuti correttamente se il tempo di scansione del programma della stazione locale è più lungo di quello della stazione master. In questo caso si raggiunge una comunicazione dati sicura utilizzando l'istruzione COM nel programma di sequenza della stazione locale.

Programmando istruzioni COM nel programma di sequenza della stazione locale, si ottiene un rinfresco automatico ogni volta che la stazione locale riceve il comando della stazione master fra le istruzioni seguenti:

- Passo 0 e istruzione COM
- Istruzione COM e istruzione COM
- Istruzione COM e istruzione END

Se il tempo di scansione del collegamento è più lungo del tempo di scansione del programma della stazione master, la comunicazione non può essere velocizzata, anche programmando l'istruzione COM nella stazione master.



- 1 Programma di sequenza della stazione master
- 2 Tempo di scansione collegamento della stazione slave

6.7.4 EI, DI

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
● ¹	● ¹				

¹ Il rinfresco del collegamento può essere eseguito solo se M9053 è attivo (1).

Operandi MELSEC A

Operandi utilizzabili														Definizione cifra	Numero di passi	Indice	Riporto errore	Segnal e di errore								
Operandi a bit							Operandi a word (16-bit)												Costanti	Punta- tore	Livello					
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H (16#)	P	I	N				M9012	M9010 M9011	
																									1	

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC																
<div style="border: 1px solid black; padding: 5px; width: fit-content;">MELSEC</div> <table border="0"> <tr><td>LD</td><td>M9036</td></tr> <tr><td>SET</td><td>M9053</td></tr> <tr><td>LD</td><td>TRUE</td></tr> <tr><td>DI</td><td></td></tr> </table>	LD	M9036	SET	M9053	LD	TRUE	DI			<table border="0"> <tr><td>LD</td><td>M9036</td></tr> <tr><td>S</td><td>M9053</td></tr> <tr><td>LD</td><td>TRUE</td></tr> <tr><td>DI_M</td><td></td></tr> </table>	LD	M9036	S	M9053	LD	TRUE	DI_M	
LD	M9036																	
SET	M9053																	
LD	TRUE																	
DI																		
LD	M9036																	
S	M9053																	
LD	TRUE																	
DI_M																		
<div style="border: 1px solid black; padding: 5px; width: fit-content;">MELSEC</div> <table border="0"> <tr><td>LD</td><td>M9036</td></tr> <tr><td>SET</td><td>M9053</td></tr> <tr><td>LD</td><td>TRUE</td></tr> <tr><td>EI</td><td></td></tr> </table>	LD	M9036	SET	M9053	LD	TRUE	EI			<table border="0"> <tr><td>LD</td><td>M9036</td></tr> <tr><td>S</td><td>M9053</td></tr> <tr><td>LD</td><td>TRUE</td></tr> <tr><td>EI_M</td><td></td></tr> </table>	LD	M9036	S	M9053	LD	TRUE	EI_M	
LD	M9036																	
SET	M9053																	
LD	TRUE																	
EI																		
LD	M9036																	
S	M9053																	
LD	TRUE																	
EI_M																		

Variabili

Dati impostati	Significato	Tipo dati
—	—	—

Funzioni Condizioni di esecuzione per il rinfresco del collegamento

Note generali

La funzione eseguita da un rinfresco di collegamento (vedi istruzione COM) dipende dalle istruzioni EI/DI. Sulle CPU AnN o A2C, la funzione delle istruzioni EI/DI dipende dallo stato del relé interno M9053. Solo se questo relé è attivo, queste istruzioni servono a definire le condizioni di esecuzione del rinfresco di collegamento. Le istruzioni determinano invece le condizioni di esecuzione dei programmi di interruzione se il relé è diseccitato.

DI Disabilita esecuzione rinfresco collegamento

L'istruzione DI disabilita l'esecuzione del rinfresco di un collegamento fino all'esecuzione di una istruzione EI. Dopo l'accensione o un reset della CPU, lo stato del rinfresco collegamento è attivo.

Il rinfresco collegamento è sempre abilitato durante l'elaborazione END.

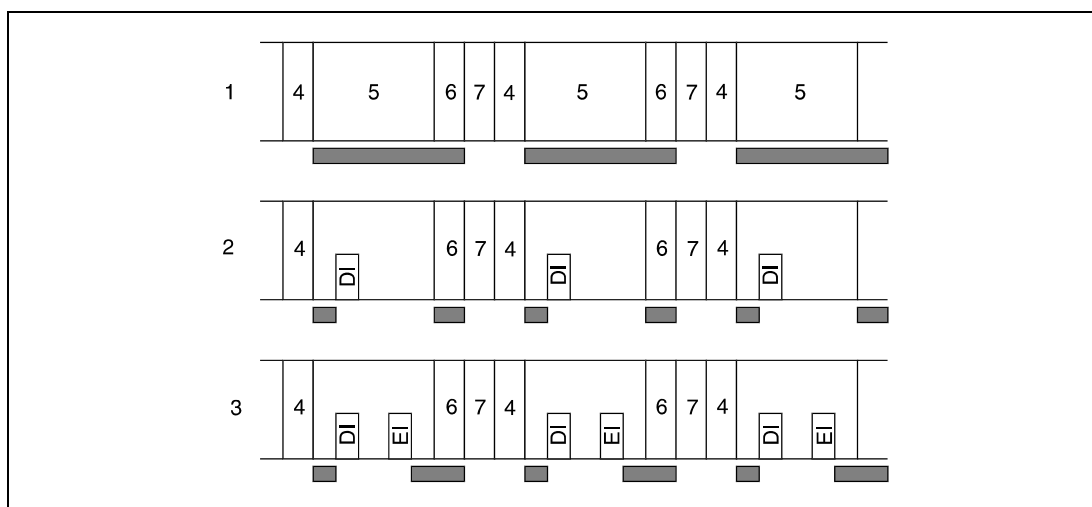
EI Abilita esecuzione rinfresco collegamento

Dopo l'attivazione dell'istruzione EI, viene abilitata l'esecuzione del rinfresco di un collegamento.

Condizioni di esecuzione

La figura seguente mostra le condizioni di esecuzione per le istruzioni EI/DI.

Le righe grigie indicano l'attività del collegamento dati. Non esiste nessuna attesa per periodo di scansione costante, se questo non è stato specificato. In modo diretto, non è possibile il rinfresco I/O.



¹ Esecuzione programma senza istruzioni EI/DI

² Esecuzione programma con istruzioni DI

³ Esecuzione programma con istruzioni EI

⁴ Rinfresco I/O

⁵ Programma di sequenza

⁶ Elaborazione END

⁷ Periodo di attesa per ciclo costante

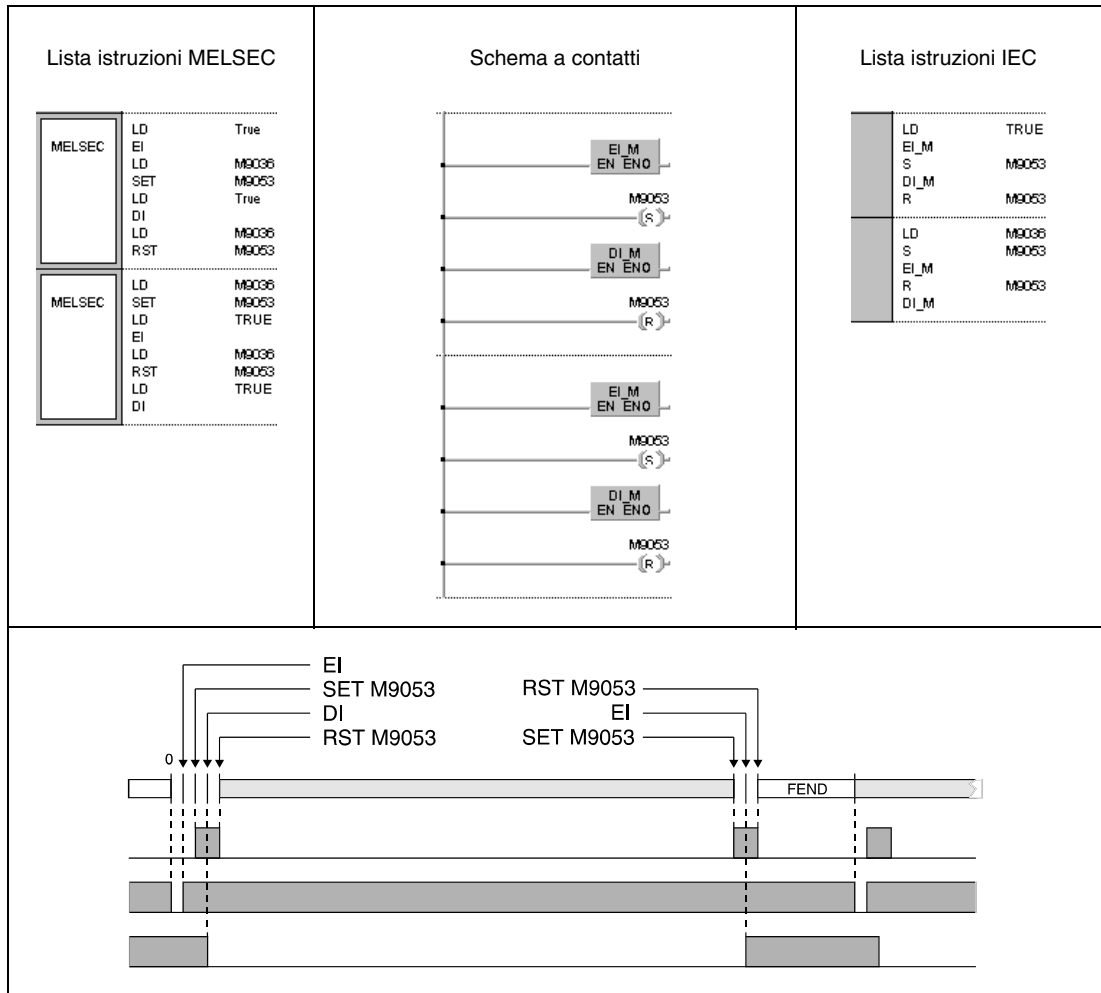
L'elaborazione avviene se è soddisfatta la condizione di esecuzione.

La funzione delle istruzioni EI/DI dipende dallo stato del relé interno M9053. Dopo l'esecuzione di una istruzione EI/DI, lo stato di M9053 non può essere modificato.

Se una istruzione EI o DI è posta all'interno di una istruzione MC, questa viene elaborata separatamente dall'esecuzione dell'istruzione MC.

Esempio di programma EI

Il programma che segue disabilita il rinfresco del collegamento fino a quando non viene eseguita l'istruzione EI, subito prima dell'istruzione FEND. Il richiamo di un programma di interruzione è possibile in qualsiasi momento. La figura mostra l'esecuzione del programma nel tempo.



6.8 Altre istruzioni utili

Le istruzioni riportate nella tabella seguente consentono la programmazione di timer e contatori speciali, contatori di impulsi e uscite impulsive. Sono anche comprese istruzioni per il posizionamento di tavole rotanti e per l'implementazione di matrici di ingressi.

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Ingresso singolo contatore avanti/indietro	UDCNT1	UDCNT1_M
Ingresso doppio contatore avanti/indietro	UDCNT2	UDCNT2_M
Timer programmabile (apprendimento)	TTMR	TTMR_M
Timer funzione speciale	STMR	STMR_M
	STMRH	STMRH_M
Posizionamento di tavole rotanti	ROTC	ROTC_M
Segnale rampa	RAMP	RAMP_M
Contatore di impulsi	SPD	SPD_M
Uscita a impulsi con numero uscite imposto	PLSY	PLSY_M
Modulazione larghezza impulso	PWM	PWM_M
Implementazione matrici di ingressi	MTR	MTR_M

6.8.1 UDCNT1

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

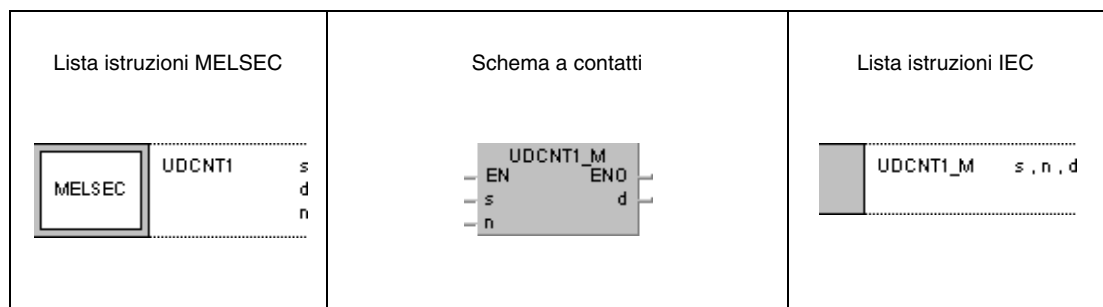
Operandi MELSEC Q

	Operandi utilizzabili									Segnale di errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s	● ¹	—	—	—	—	—	—	—	—	4	
d	—	● ²	—	—	—	—	—	—	—		
n	●	●	●	●	●	●	●	●	—		

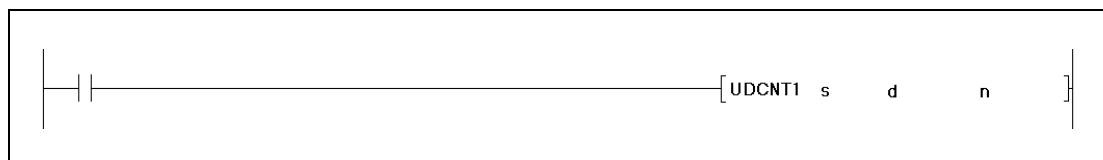
¹ Solo X

² solo C

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s	s+0: Numero operando per ingresso di conteggio (segnale impulsivo, fase). s+1: Imposta conteggio avanti o indietro 0 = conteggio avanti 1 = conteggio indietro	bit	Array [1..2] di BOOL
d	Numero del contatore che esegue l'istruzione UDCNT1.	BIN 16-bit (solo contatore)	ANY16
n	Impostazione	BIN 16-bit	ANY16

Funzioni **contatore avanti/indietro a fase singola**

UDCNT1 **Istruzione di conteggio**

Quando l'ingresso indicato da s+0 (array_s [0]) passa da 0 a 1, l'uscita attuale del contatore indicato da d viene aggiornata. Vengono quindi contati solo i fronti di salita.

LA direzione di conteggio viene determinata dallo stato dell'ingresso indicato da s+1 (array_s [1]):

Se la condizione d'ingresso è 0, gli impulsi dell'ingresso indicato da s+0 (Array_s [0]) vengono sommati al valore attuale di conteggio.

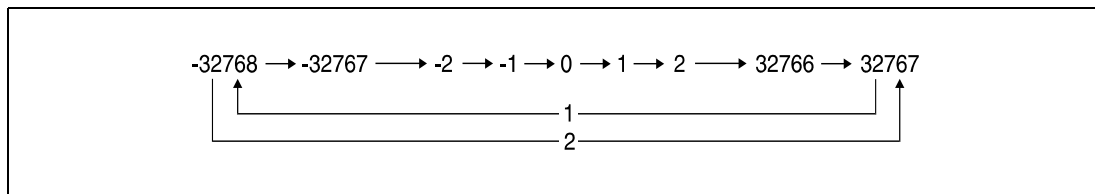
Se la condizione d'ingresso è 1, gli impulsi dell'ingresso indicato da s+0 (Array_s [0]) vengono sottratti dal valore attuale di conteggio.

La logica del conteggio è la seguente:

Durante il conteggio avanti, il contatto del contatore indicato da d viene attivato (1) se il valore attuale di conteggio è identico al valore impostato con n. L'elaborazione del conteggio continua anche mentre il contatto del contatore è attivo (vedi esempio di programma).

Durante il conteggio indietro, il contatto del contatore viene disattivato (0) se il valore attuale di conteggio è uguale a n-1 (vedi esempio di programma).

Il contatore indicato da d è un contatore ciclico. Se il conteggio raggiunge 32767 e viene incrementato, il contatore passa a -32768. Se il contatore contiene -32768 e viene decrementato, il contatore passa a 32767. La figura seguente mostra il conteggio ciclico:



- ¹ Conteggio avanti
- ² Conteggio indietro

L'istruzione UDCNT1 si attiva quando la condizione di esecuzione è vera e si disattiva quando la condizione di esecuzione è falsa. Se il contatore viene riavviato, il conteggio continua dal valore raggiunto prima dell'arresto.

L'istruzione RST azzerà il contatore indicato da d ed apre il contatto corrispondente.

NOTA

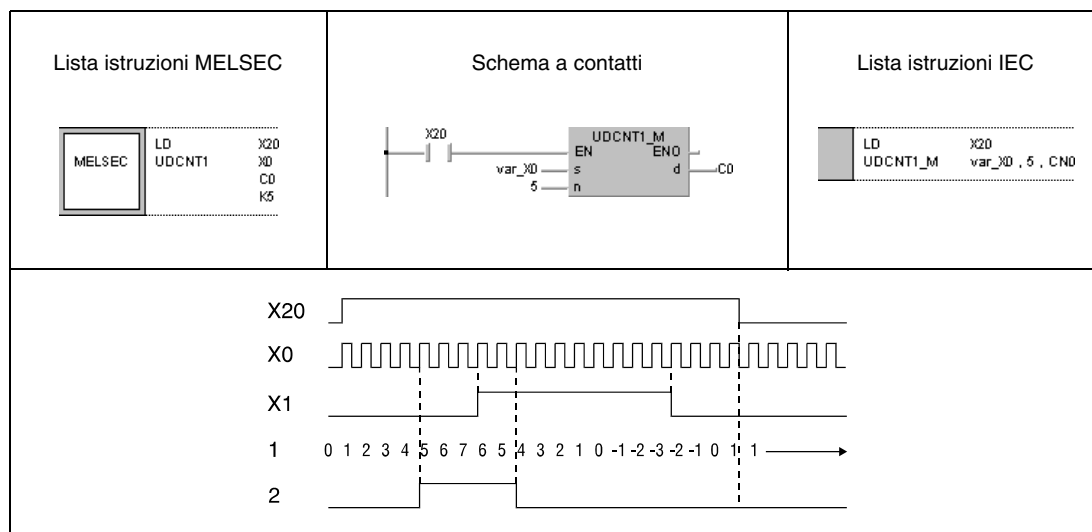
Il processo di conteggio di una istruzione UDCNT1 viene eseguito durante una interruzione della CPU (1 ms per CPU multiprocessore del System Q, 5 ms per QnACPU). Per questa ragione è possibile contare con precisione solo impulsi con durate di stato rispettivamente di 1ms e 5 ms.

Il valore impostato non può essere modificato durante il conteggio (ingresso corrispondente a s+0 (Array_s [0]) attivo). Per modificare l'impostazione, è necessario disattivare l'ingresso indicato da s+0 (Array_s [0]) .

I contatori oggetto di una istruzione UDCNT1 non possono essere utilizzati contemporaneamente da altre istruzioni. In questo caso il valore restituito può non essere preciso.

Esempio di programma**UDCNT1**

Se X20 è attivo, il programma seguente utilizza il contatore C0 (avanti/indietro) per contare il numero di fronti di salita di X0.



¹ Contatto

² del contatore C0

NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.8.2 UDCNT2

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

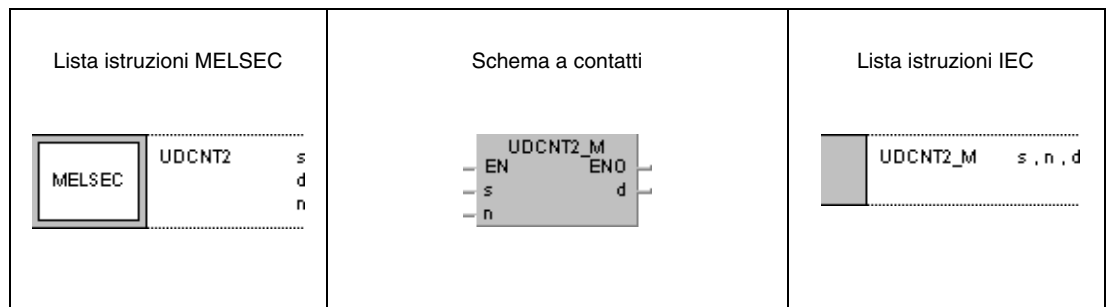
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	● ¹	—	—	—	—	—	—	—	4		
d	—	● ²	—	—	—	—	—	—			
n	●	●	●	●	●	●	●	—			

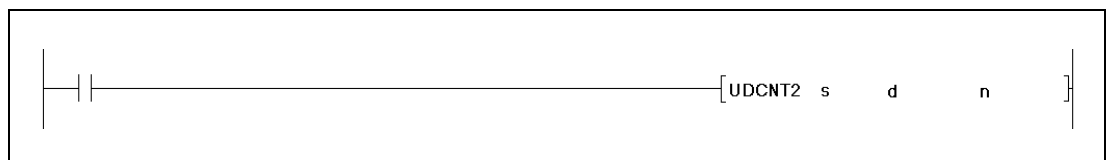
¹ Solo X

² solo C

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s	s+0: Numero operando per ingresso di conteggio (segnale impulsivo, fase A). s+1: Numero operando per ingresso di conteggio (segnale impulsivo, fase B).	Bit	Array [1..2] di BOOL
d	Numero del contatore che esegue l'istruzione UDCNT2.	BIN 16-bit (solo contatore)	ANY16
n	Impostazione	BIN 16-bit	ANY16

Funzioni **contatore avanti/indietro a fase doppia****UDCNT2** **Istruzione di conteggio**

Il valore di conteggio del contatore indicato da d cambia in funzione dello stato dei due ingressi s+0 (array_s [0]) e s+1 (array_s [1]).

La direzione di conteggio viene determinata come segue:

Se l'ingresso s+0 (array_s[0]) è attivo (1) e l'ingresso s+1 (array_s[1]) passa da 0 a 1 il valore di conteggio viene incrementato di 1.

Se l'ingresso s+0 (array_s[0]) è attivo (1) e l'ingresso s+1 (array_s[1]) passa da 1 a 0 il valore di conteggio viene decrementato di 0.

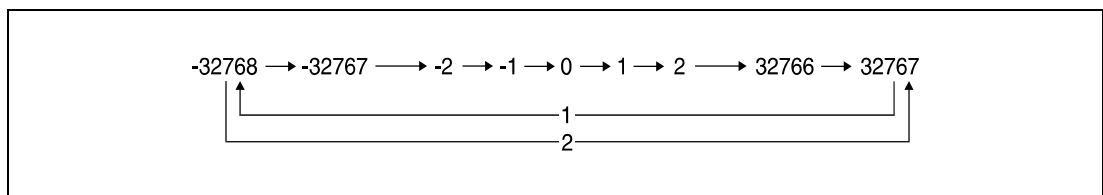
Se l'ingresso s+0 (array_s[0]) non è attivo (0), non viene eseguita nessuna funzione di conteggio.

La logica del conteggio è la seguente:

Durante il conteggio avanti, il contatto del contatore indicato da d viene attivato (1) se il valore attuale di conteggio è identico al valore impostato con n. L'elaborazione del conteggio continua anche mentre il contatto del contatore è attivo (vedi esempio di programma).

Durante il conteggio indietro, il contatto del contatore viene disattivato (0) se il valore attuale di conteggio è uguale a n-1 (vedi esempio di programma).

Il contatore indicato da d è un contatore ciclico. Se il conteggio raggiunge 32767 e viene incrementato, il contatore passa a -32768. Se il contatore contiene -32768 e viene decrementato, il contatore passa a 32767. La figura seguente mostra il conteggio ciclico:



¹ Conteggio avanti

² Conteggio indietro

L'istruzione UDCNT2 si attiva quando la condizione di esecuzione è vera e si disattiva quando la condizione di esecuzione è falsa. Se il contatore viene riavviato, il conteggio continua dal valore raggiunto prima dell'arresto.

L'istruzione RST azzerà il contatore indicato da d ed apre il contatto corrispondente.

NOTA

Il processo di conteggio di una istruzione UDCNT2 viene eseguito durante una interruzione della CPU (1 ms per CPU multiprocessore del System Q, 5 ms per QnACPU). Per questa ragione è possibile contare con precisione solo impulsi con durate di stato rispettivamente di 1ms e 5 ms.

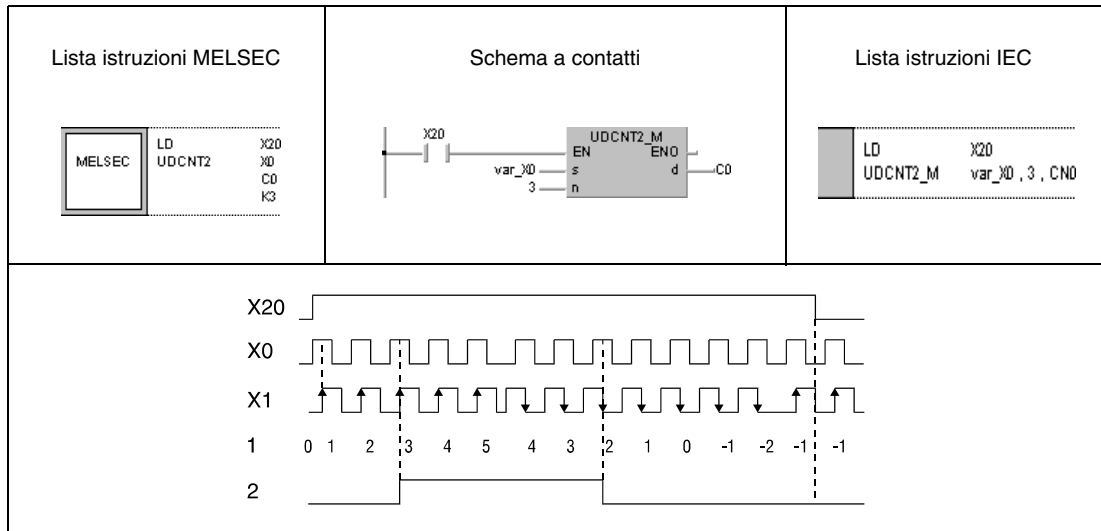
Il valore impostato non può essere modificato durante il conteggio (ingresso corrispondente a s+0 (Array_s [0]) attivo). Per modificare l'impostazione, è necessario disattivare l'ingresso indicato da s+0 (Array_s [0]).

I contatori oggetto di una istruzione UDCNT2 non possono essere utilizzati contemporaneamente da altre istruzioni. In questo caso il valore restituito può non essere preciso.

Esempio di programma

UDCNT2

Se X20 è attivo, il programma che segue utilizza il contatore C0. La logica e la direzione di conteggio dipendono dalle condizioni di X0 e X1.



NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.8.3 TTMR

CPU

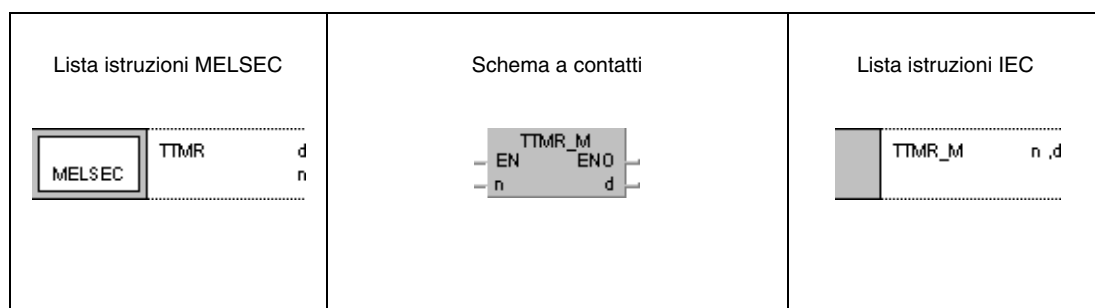
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

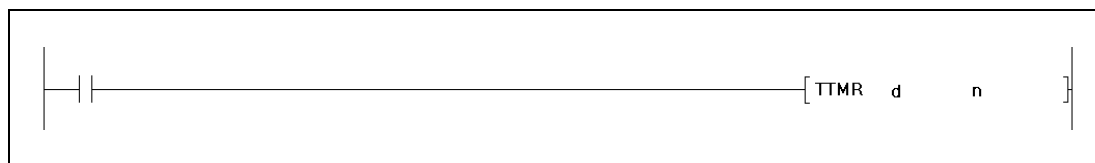
Operandi MELSEC Q

	Operandi utilizzabili									Segnale di errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
d	—	●	●	—	—	—	—	—	—	SM0	3
n	—	●	●	●	●	●	●	●	—		

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
d	d+0: Operando che memorizza il valore della misura	BIN 16-bit	Array [1..2] di ANY16
	d+1: Per uso interno della CPU.		
n	Moltiplicatore valore della misura		ANY16

Funzioni **Timer programmabile (apprendimento)**

TTMR Istruzione timer

Un timer programmato con l'istruzione TTMR misura la durata di un segnale di ingresso in secondi. Il valore della misura viene moltiplicato per n e memorizzato in (array_d [0]+[1]).

Con il fronte di salita dell'ingresso, gli operandi d+0 (array_d [0]) e d+1 (array_d [1]) vengono azzerati.

I moltiplicatori espressi da n, sono i seguenti:

n = 0, moltiplicatore 1

n = 1, moltiplicatore 10

n = 2, moltiplicatore 100

NOTA

La misura di tempo viene eseguita durante l'esecuzione dell'istruzione TTMR. Se una istruzione JMP, o altra istruzione simile, scavalca l'istruzione TTMR, le misure di tempo non sono precise.

Il moltiplicatore n non deve cambiare durante l'esecuzione dell'istruzione TTMR. Una variazione provoca errori di misura.

L'istruzione TTMR viene utilizzata anche per programmi di tipo lento.

L'operando indicato da d+1 (array_d [1]) viene utilizzato dalla CPU. Una variazione provoca errori di misura.

Errori di esecuzione

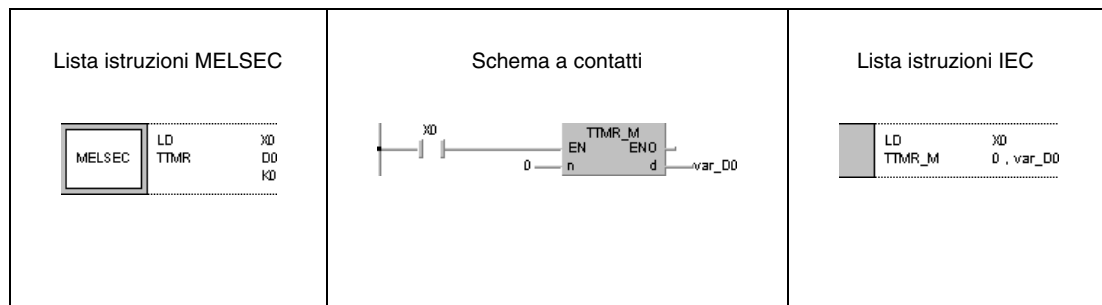
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il valore indicato da n supera il campo valori ammesso da 0 a 2 (codice di errore: 4100).

Esempio di programma

TTMR

Se X0 è attivo, il programma seguente misura il tempo in secondi (moltiplicatore = 1). Il risultato è memorizzato in D0.



NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.8.4 STMR, STMRH

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

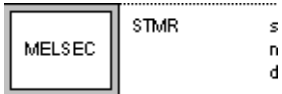
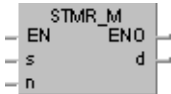
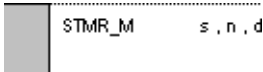
¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

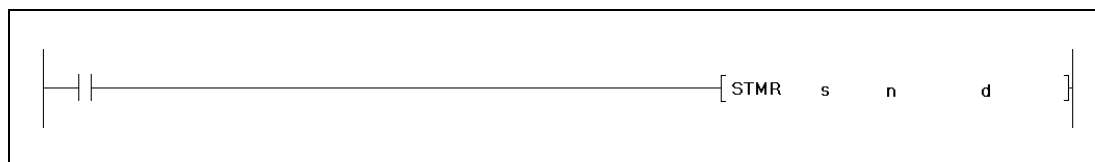
	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti			Altro
	Bit	Word		Bit	Word						
s	—	● ¹	—	—	—	—	—	—	—	3	
n	●	—	—	—	—	—	—	—	—		
d	—	●	●	●	●	●	●	●	—		

¹ Può essere usato solo con dati timer (T).

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
---	---	---

GX Developer



Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s	Numero del timer.	BIN 16-bit (solo timer)	ANY16
n	Impostazione tempo.	BIN 16-bit	ANY16
d	d+0: Uscita timer OFF ritardata. d+1: Uscita timer impulsiva dopo OFF (impostata da fronte di discesa) d+2: Uscita timer impulsiva dopo ON (impostata da fronte di salita) d+3: Uscita timer ON ritardata.	Bit	Array [1.0.4] di BOOL

Funzioni	Timer speciale
	STMR Istruzione per timer lenti
	STMRH Istruzione per timer veloci

L'istruzione STMR utilizza le uscite indicate da d+0 fino a d+3 (da array_d [0] a array_d [3]) per eseguire quattro diverse funzioni di temporizzazione:

Uscita timer ritardata su OFF (d+0) (array_d [0])

L'uscita indicata da d+0 (array_d [0]) si attiva (1) sul fronte di salita della condizione di esecuzione. L'uscita viene disattivata (0) nuovamente dopo il fronte di discesa della condizione di esecuzione e dopo il periodo di tempo indicato da n.

Uscita timer impulsiva dopo OFF (impostata da fronte di discesa, d+1 (array_d [1]))

L'uscita indicata da d+1 (array_d [1]) si attiva (1) sul fronte di discesa della condizione di esecuzione. Dopo il periodo di tempo indicato da n o sul fronte di salita della condizione di esecuzione, l'uscita viene disattivata (0) nuovamente.

Uscita timer impulsiva dopo ON (impostata da fronte di salita, d+2 (array_d [2]))

L'uscita indicata da d+2 (array_d [2]) si attiva (1) sul fronte di salita della condizione di esecuzione. Dopo il periodo di tempo indicato da n o sul fronte di discesa della condizione di esecuzione, l'uscita viene disattivata (0) nuovamente.

Uscita timer ritardata su ON (d+3 (Array [3]))

L'uscita indicata da d+3 (array_d [3]) si attiva (1) sul fronte di salita della condizione di esecuzione. Questo corrisponde al ritardo all'eccitazione indicato da n. L'uscita d+3 viene attivata anche dal fronte di discesa della condizione di esecuzione e disattivata (0) dopo il periodo indicato da n.

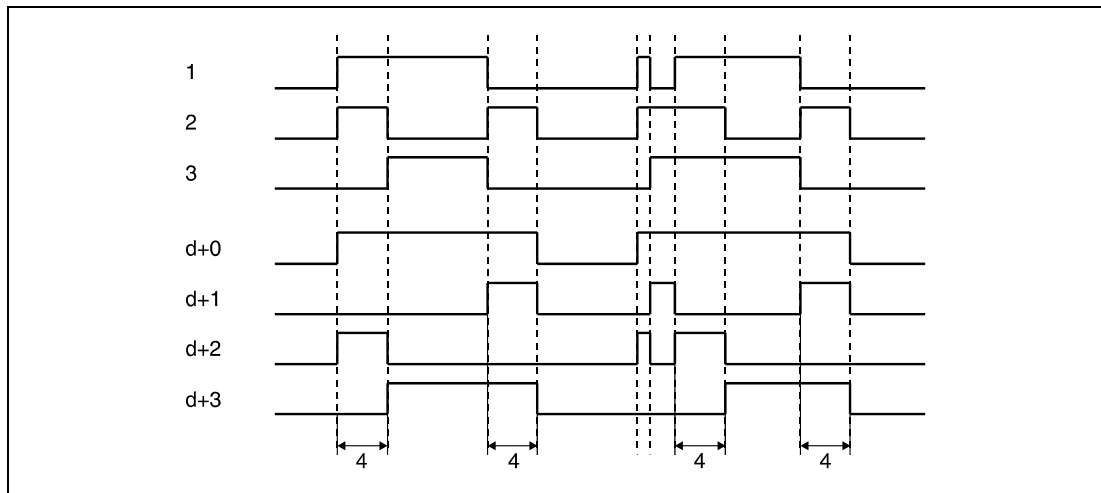
La bobina del timer indicato da s viene attivata con il fronte di salita della condizione di esecuzione ed inizia la misura del tempo indicato da n.

La bobina misura il tempo fino a quando il valore misurato corrisponde al tempo impostato, poi cade.

Se la condizione di esecuzione si disattiva prima che sia trascorso il tempo impostato con n, la bobine del timer rimane eccitata e la misura di tempo viene sospesa in quel punto.

Se la condizione di esecuzione viene attivata nuovamente, il valore viene azzerato e la misura del tempo riprende.

Il contatto del timer indicato da s viene attivato sia dal fronte di discesa della condizione di esecuzione con bobina timer attiva, che dal fronte di discesa della bobina del timer con condizione di esecuzione attiva. Il contatto del timer si apre sul fronte di discesa della condizione di esecuzione con bobine del timer disattivata. Il contatto del timer è disponibile solo per uso interno alla CPU.



- ¹ Condizione di esecuzione
- ² Bobina del timer indicato da s
- ³ Contatto del timer indicato da s
- ⁴ Tempo impostato n

La misura di tempo viene eseguita durante l'esecuzione dell'istruzione STMR. Se una istruzione JMP, o altra istruzione simile, scavalca l'istruzione STMR, le misure di tempo non sono precise.

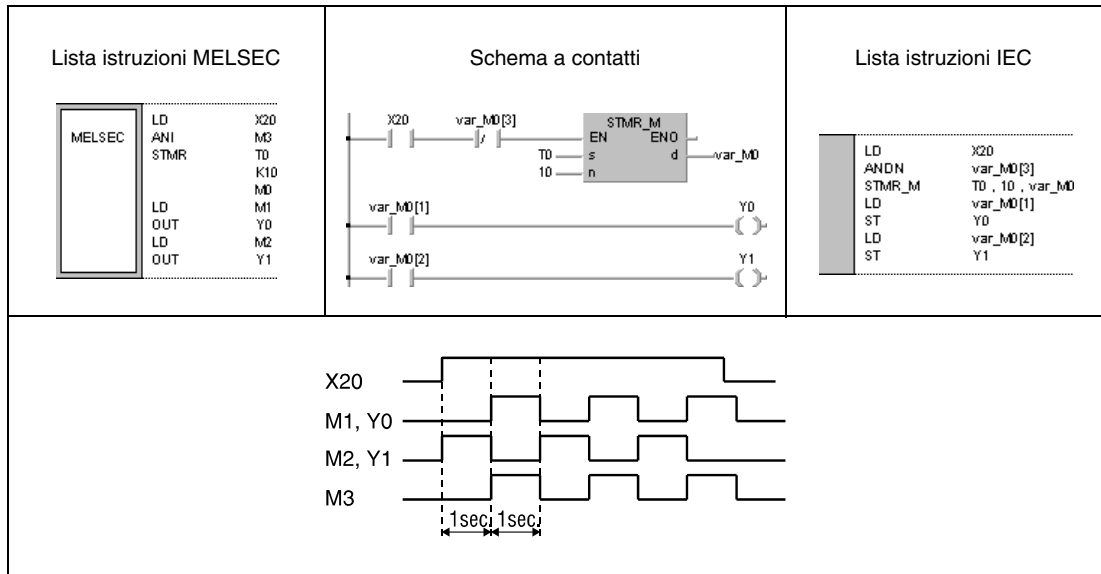
Il tempo effettivo indicato da d può essere calcolato moltiplicando il tempo impostato n con l'unità di tempo per i contatori lenti (valore di default + 100 ms).

La costante n può variare nel campo fra 1 e 32767.

Il timer indicato da s non può essere usato per una istruzione OUT. Se su usano contemporaneamente le istruzioni OUT e STMR sullo stesso timer, l'istruzione STMR non funziona correttamente.

Esempio di programma STMR

Se X20 è attivo, il programma seguente attiva alternativamente le uscite Y0 e Y1 per 1 secondo ciascuna. Il timer utilizzato è un timer a 100 ms. Il periodo di tempo da 1 secondo viene calcolato moltiplicando K10 con 100 ms.



NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.8.5 ROTC

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

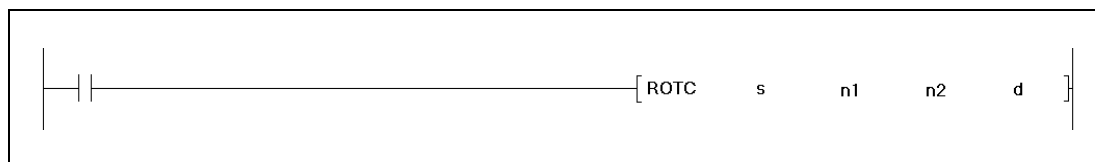
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	—	SM0	5	
n1	●	●	●	●	●	●	●	—			
n2	●	●	●	●	●	●	●	—			
d	●	—	—	—	—	—	—	—			

GX IEC Developer

<p>Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; display: inline-block; margin-bottom: 5px;">MELSEC</div> <p>ROTC s n1 n2 d</p>	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p> <div style="border: 1px solid black; padding: 5px; display: inline-block; margin-bottom: 5px;"></div> <p>ROTC_M s , n1 , n2 , d</p>
--	--------------------------	--

GX Developer



Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s	s+0: Misura numero di giri della tavola (solo uso interno).	BIN 16-bit	Array [1..3] di ANY16
	s+1: Numero di posizione		
	s+2: Numero del settore.		
n1	Numero settori (divisioni) della tavola (da 2 a 32767).		ANY16
n2	Numero settori lenti (da 0 a n1).	ANY16	
d	d+0: Fase A segnale di ingresso	Bit	Array [1.0.8] di BOOL
	d+1: Fase B segnale di ingresso		
	d+2: Segnale d'ingresso rilevamento posizione di zero.		
	d+3: Segnale di uscita avanti veloce (solo uso interno).		
	d+4: Segnale di uscita avanti lento (solo uso interno).		
	d+5: Segnale di uscita Stop (solo uso interno).		
	d+6: Segnale di uscita indietro veloce (solo uso interno).		
d+7: Segnale di uscita indietro lento (solo uso interno).			

Funzioni Istruzione di posizionamento per tavole rotanti**ROTC Istruzione di posizionamento**

L'istruzione ROTC ruota il settore indicato da s+2 (array_s [2]) su una tavola con un numero di settori (divisioni) definito, indicato da n1, fino alla posizione indicata da s+1 (array_s [1]).

Posizioni e settori della tavola rotante sono numerati in senso antiorario.

Il valore di s+0 (array_s [0]) viene utilizzato internamente dal sistema per determinare la posizione del settore in funzione della posizione di zero. Questo valore non deve essere modificato per non compromettere la precisione di posizionamento della tavola.

Il valore di n2 determina il numero di settori della tavola da percorrere a bassa velocità. Questo valore deve essere inferiore a quello indicato da n1.

Gli ingressi delle fasi A/B indicati da d+0 (array_d [0]) e d+1 (array_d [1]) rilevano la direzione di rotazione. Entrambi gli ingressi ricevono impulsi. Se l'ingresso d+0 (array_d [0]) della fase A è attivo, la direzione di rotazione viene determinata dal fronte dell'impulso sull'ingresso d+1 (array_d [1]) della fase B:

Se la fase B presenta un fronte di salita, in quel momento la tavola ruota in senso orario (verso destra).

Se la fase B presenta un fronte di discesa, in quel momento la tavola ruota in senso antiorario (verso sinistra).

L'ingresso definito da d+2 (array_d [2]) rileva la posizione di zero. Questo ingresso è attivo se il settore raggiunge posizione 0. Se questo ingresso è attivo durante l'esecuzione di una istruzione ROTC, il valore in s+0 (array_s [0]) viene azzerato. Per ottenere un posizionamento accurato, azzerare il valore in s+0 (array_s [0]) prima di eseguire l'istruzione ROTC.

I dati da d+3 (array_d [3]) a d+7 (array_d [7]) contengono i segnali di uscita per il funzionamento della tavola rotante. L'attivazione dei segnali di uscita dipende dal risultato corrente dell'operazione svolta dall'istruzione ROTC.

Se il risultato di tutte le operazioni è 0 prima di eseguire una istruzione ROTC, le uscite indicate da d+3 (array_d [3]) a d+7 (array_d [7]) sono azzerate senza posizionare la tavola. Le uscite vengono azzerate anche se la condizione di esecuzione viene disattivata.

Una istruzione ROTC può essere eseguita solo una volta in un programma. L'utilizzo ripetuto dell'istruzione in un programma provoca un funzionamento non corretto dell'istruzione.

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il valore specificato da s+0 (array_s [0]) a s+2 (array_s [2]) o in n2 è maggiore di quello in n1 (codice di errore: 4100).

Esempio di programma ROTC

Nel programma che segue i contatti X0, X1 (encoder incrementale) e X2 indirizzano i relé interni per il rilevamento della direzione di rotazione e della posizione di zero da M0 (var_M0 array [0]) a M2 (var_M0 array [2]). Il contatto X2 viene attivato se il settore 0 è posto sulla posizione 0 (rilevamento posizione di zero).

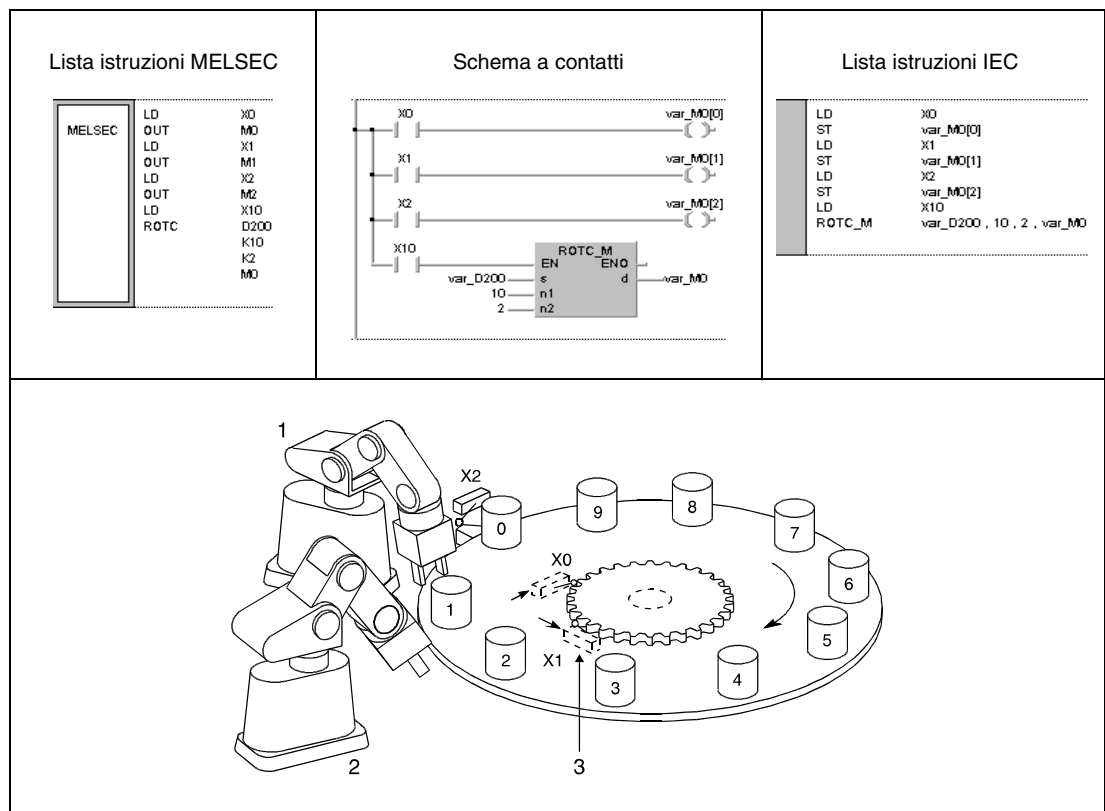
La tavola rotante considerata dall'esempio è divisa in dieci settori.

L'elemento (setto) da muovere e la stazione (posizione) devono essere specificati in D201 (var_D200 array [1]) e D202 (var_D200 array [2]) prima dell'esecuzione dell'istruzione ROTC.

A causa del valore n1=10, il contatto del registro contatore emette 10 impulsi per ogni rotazione (divisione). Il valore n2=2 specifica il numero delle divisioni da percorrere in lento.

Ad esempio, se il registro D201 (var_D200 array [1]) contiene il valore 0 e il registro D202 (var_D200 array [2]) contiene il valore 3, la tavola rotante muove l'elemento 3 (setto 3) sulla stazione 0 (posizione 0) lungo il percorso più breve (senso orario). I settori da 1 a 3 vengono percorsi a bassa velocità.

Per una allocazione dei singoli registri e relé interni o elementi di array rispettivamente con le funzioni corrispondenti, vedi la tabella che segue l'esempio.



- 1 Stazione 0 (posizione 0)
- 2 Stazione 1 (posizione 1)
- 3 Encoder incrementale

Registro dati	Significato	Nota
D200 (var_D200 Array [0])	Registro contatore	
D201 (var_D200 Array [1])	Posizione della stazione	Questi valori vengono scritti nei registri dati D201 (var_D200 array [1]) e D202 (var_D200 array [2]) tramite una istruzione MOV.
D202 (var_D200 Array [2])	Posizione dell'elemento	
M0 (var_M0 Array [0])	Segnale fase A	I relé interni da M0 (var_M0 array [0]) a M2 (var_M0 array [2]) corrispondono agli ingressi da X0 a X2 (vedi programma di esempio).
M1 (var_M0 Array [1])	Segnale fase B	
M2 (var_M0 Array [2])	Segnale rilevamento posizione di zero.	
M3 (var_M0 Array [3])	Rotazione avanti veloce	Dopo l'attivazione di X10, l'istruzione ROTC viene attivata ed i relé interni da M3 (var_M0 array [3]) a M7 (var_M0 array [7]) vengono assegnate delle funzioni specifiche. Quando X10 viene disattivato, anche i relé interni vengono disattivati.
M4 (var_M0 Array [4])	Rotazione avanti lento	
M5 (var_M0 Array [5])	Segnale di arresto	
M6 (var_M0 Array [6])	Rotazione indietro veloce	
M7 (var_M0 Array [7])	Rotazione indietro lento	

NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.8.6 RAMP

CPU

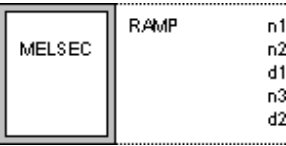
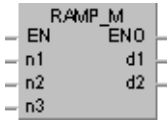
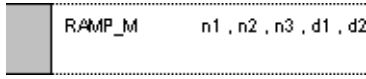
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

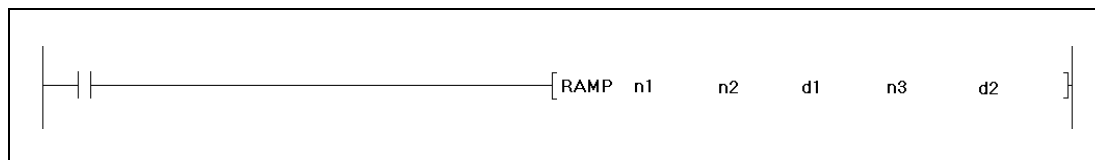
Operandi MELSEC Q

	Operandi utilizzabili									Segnale di errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
n1	●	●	●	●	●	●	●	●	—	—	6
n2	●	●	●	●	●	●	●	●	—	—	
d1	●	●	●	●	●	●	●	—	—	—	
n3	●	●	●	●	●	●	●	●	—	—	
d2	●	—	—	—	—	—	—	—	—	—	

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
---	---	--

GX Developer



Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
n1	Valore iniziale dell'operazione.	BIN 16-bit	ANY16
n2	Valore finale dell'operazione.		ANY16
d1	(d1)+0: Operando che memorizza il valore attuale.		Array [1.0.2] di ANY16
	(d1)+1: Operando che contiene il numero dei passi eseguiti (solo uso interno).		
n3	Numero di passi da eseguire.		ANY16
d2	(d2)+0: Bit da attivare al termine.	Bit	Array [1.0.2] di BOOL
	(d2)+1: Bit che determina la memorizzazione del risultato dell'operazione.		

Funzioni Segnale rampa

RAMP Istruzione per la variazione graduale del valore di un operando

Una istruzione RAMP modifica gradualmente il contenuto di (d1)+0 dal valore iniziale indicato da n1, fino al valore finale indicato da n2.

Il numero di passi in cui deve essere suddivisa la variazione è indicato da n3.

Il numero di passi già compiuti è contenuto in (d1)+1 (array_d1 [1]) per uso interno del sistema.

Al termine dell'operazione, l'operando indicato da (d2)+0 (array_d2 [0]) viene attivato.

Lo stato dell'operando (d2)+0 (array_d2 [0]) ed il contenuto dell'operando (d1)+0 (array_d1 [0]) dipendono dallo stato dell'operando (d2)+1 (array_d2 [1]):

Se l'operando (d2)+1 (array_d2 [1]) non è attivo, l'operando (d2)+0 (array_d2 [0]) viene disattivato durante la scansione successiva e l'istruzione RAMP inizia una nuova operazione, partendo dal valore attualmente contenuto in (d1)+0 (array_d1 [0]).

Se l'operando (d2)+1 (array_d2 [1]) è attivo, l'operando (d2)+0 (array_d2 [0]) rimane attivo ed il valore in (d1)+0 (array_d1 [0]) non viene modificato (memoria).

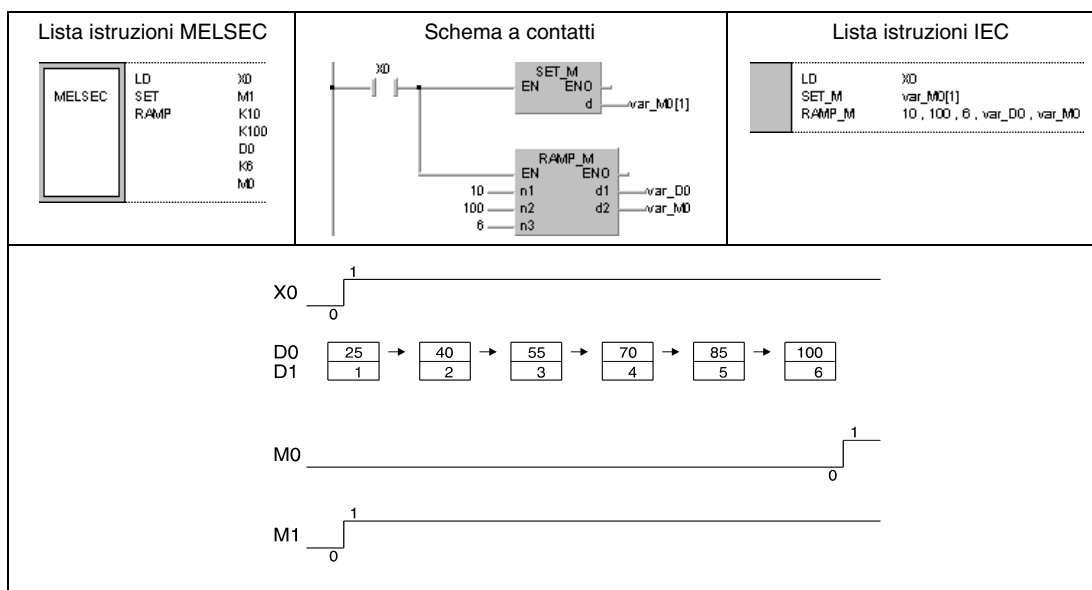
Se la condizione di esecuzione viene disattivata durante l'operazione, il contenuto in (d1)+0 (array_d1 [0]) non cambia. Se la condizione di esecuzione viene riattivata, l'istruzione RAMP cambia il contenuto in (d1)+0 (array_d1 [0]) memorizzato prima del reset.

I valori n1 e n2 non devono cambiare durante l'esecuzione dell'istruzione.

Esempio di programma

RAMP

Il programma seguente incrementa da 10 a 100 il contenuto di D0 in 6 passi e memorizza il contenuto in D0 al termine dell'operazione.



NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

6.8.7 SPD

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

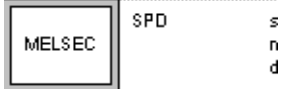
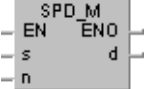
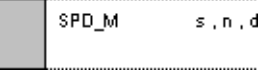
¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

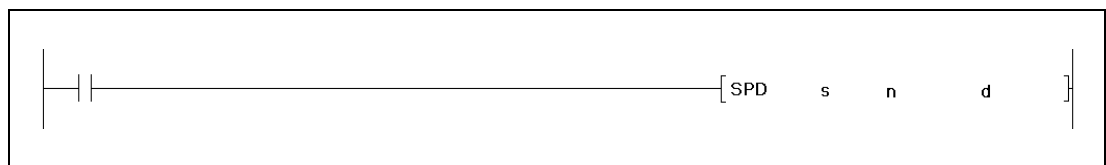
	Operandi utilizzabili									Segnale di errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti	Altro		
	Bit	Word		Bit	Word						
s	● ¹	—	—	●	●	●	●	●	—	—	4
n	●	●	●	—	—	—	—	—	—	—	
d	—	●	●	●	●	●	●	●	—	—	

¹ Solo X

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
---	---	--

GX Developer



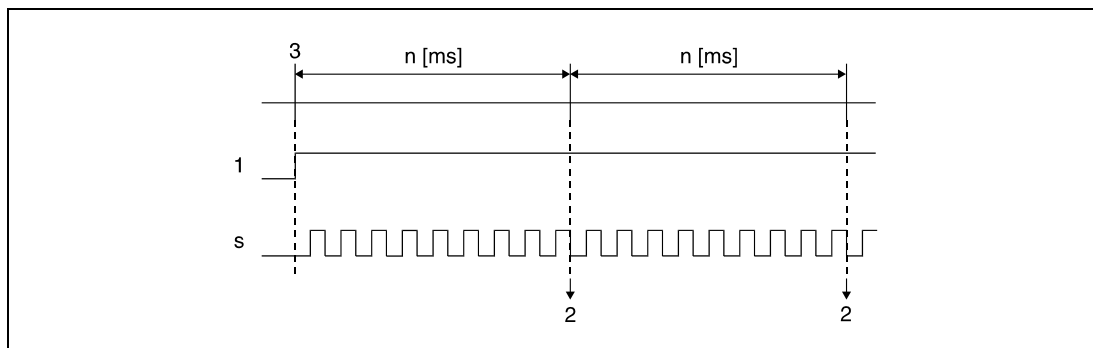
Variabili

Dati impostati	Significato	Tipo dati
s	Segnale impulsivo d'ingresso	Bit
n	Tempo di misura (unità: ms).	BIN 16-bit
d	Primo numero dell'operando che memorizza il risultato.	

Funzioni Misura densità di impulso

SPD Misura densità di impulso

L'istruzione SPD conta gli impulsi sull'ingresso indicato da s per il periodo di tempo specificato da n. Il risultato della misura viene memorizzato in d.



- 1 Condizione di esecuzione.
- 2 Il risultato della misura è memorizzato in d.
- 3 Inizio misura.

Fintanto che la condizione di esecuzione è attiva, la misura riparte di nuovo da 0 quando il tempo di misura è trascorso. Per arrestare la misura SPD si deve disattivare la condizione di ingresso.

L'istruzione SPD memorizza i dati degli operandi designati nell'area di lavoro della CPU, ed effettua le operazioni di conteggio durante l'interruzione di sistema da 5 ms. Per questa ragione, il numero di volte in cui si può usare l'istruzione SPD è limitato. Le istruzioni SPD che superano questo limite non vengono eseguite.

Nota

L'elaborazione del conteggio degli impulsi utilizzati con l'istruzione SPD viene eseguita durante l'interruzione di sistema. Per poter contare correttamente gli impulsi è quindi necessario fare in modo che la durata di stato degli impulsi sia uguale o maggiore al periodo dell'interruzione di sistema della CPU. Il periodo di interruzione è 1 ms per una CPU multiprocessore del System Q e 5 ms per una QnA-CPU.

Se si utilizza una CPU del System Q, l'istruzione SPD non viene eseguita se n=0.
 Se si utilizza una CPU QnA, l'istruzione SPD non viene eseguita se n = 0 o n non multiplo di 5.
 L'istruzione SPD può essere usata fino a sei volte fra tutti i programmi in esecuzione. La settima istruzione SPD o successive non vengono elaborate.

Esempio di programma

SPD

Se X10 è attivo, il programma che segue conta gli impulsi di X0 in un periodo di 500 ms. Il risultato viene memorizzato di D0.

<p>Lista istruzioni MELSEC</p> <pre style="border: 1px solid black; padding: 5px; margin: 10px auto; width: 80%;"> LD X10 SPD X0 K500 D0 </pre>	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p> <pre style="border: 1px solid black; padding: 5px; margin: 10px auto; width: 80%;"> LD X10 SPD_M X0 , 500 , D0 </pre>
---	--------------------------	---

6.8.8 PLSY

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

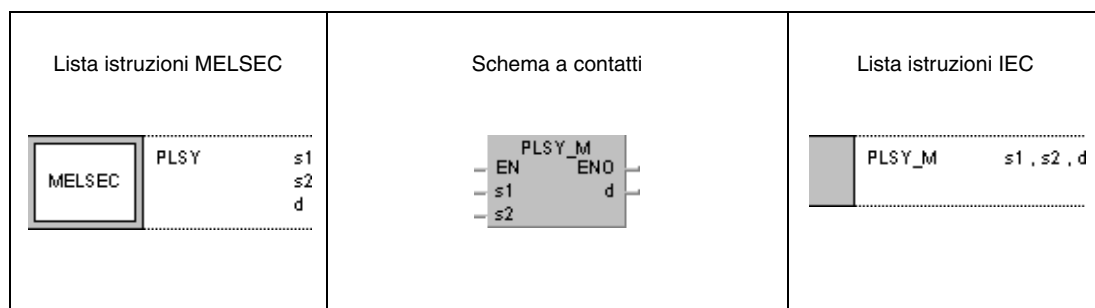
¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

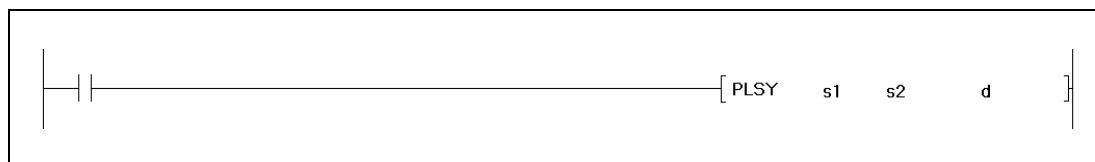
	Operandi utilizzabili									Segnale di errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti	Altro		
	Bit	Word		Bit	Word						
s1	●	●	●	●	●	●	●	●	—	—	4
s2	●	●	●	●	●	●	●	●	—	—	
d	● ¹	—	—	—	—	—	—	—	—	—	

¹ Solo Y

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s1	Operando che memorizza le impostazioni di frequenza.	BIN 16-bit
s2	Operando che memorizza le impostazioni del numero di impulsi di uscita.	
d	Operando che memorizza l'uscita	Bit

Funzioni Uscita impulsiva con numero di impulsi regolabile

PLSY Istruzione per uscita impulsi

L'istruzione PLSY emette il numero di impulsi specificato da s2 con la frequenza specificata da s1, sull'uscita indicata da d.

Il campo di frequenza in s1 può variare fra 1 e 100 Hz. Se s1 contiene il valore 0, l'istruzione PLSY emette un segnale continuo.

Il numero di impulsi contenuto in s2 può variare da 1 a 32767.

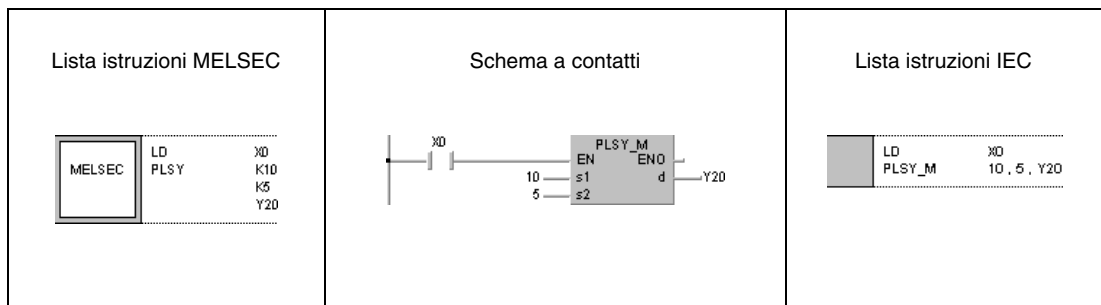
L'operando d può indicare solo uscite corrispondenti a moduli di uscita.

L'uscita del treno di impulsi inizia con il fronte di salita della condizione di esecuzione dell'istruzione PLSY. Durante l'emissione del treno di impulsi la condizione di esecuzione non deve essere disabilitata. La disabilitazione della condizione di esecuzione sospende l'emissione degli impulsi.

L'istruzione PLSY memorizza i dati degli operandi designati nell'area di lavoro della CPU, ed effettua le operazioni di emissione degli impulsi durante l'interruzione di sistema. Per questa ragione l'istruzione PLSY può essere usata solo una volta nel programma. Il periodo di interruzione è 1 ms per una CPU multiprocessore del System Q e 5 ms per una CPU QnA.

Esempio di programma

PLSY
Se X0 è attivo, il programma seguente emette cinque impulsi a 10 Hz su Y20.



6.8.9 PWM

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

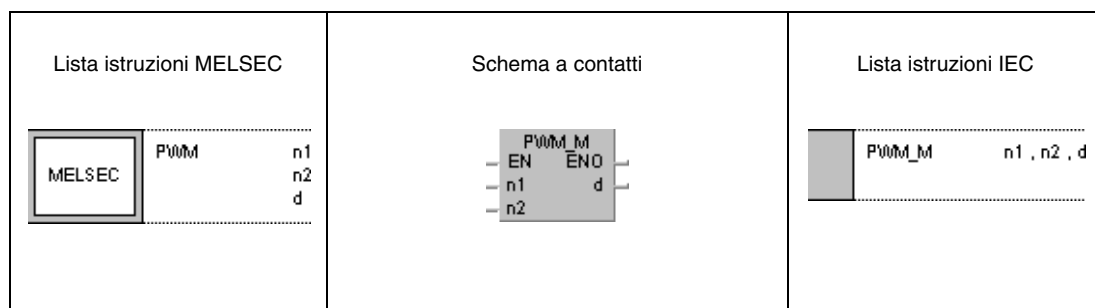
¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

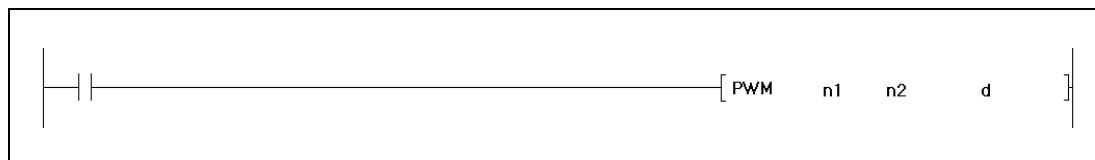
	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti			Altro
	Bit	Word		Bit	Word						
n1	●	●	●	●	●	●	●	—	—	4	
n2	●	●	●	●	●	●	●	—	—		
d	● ¹	—	—	—	—	—	—	—	—		

¹ Solo Y

GX IEC Developer



GX Developer



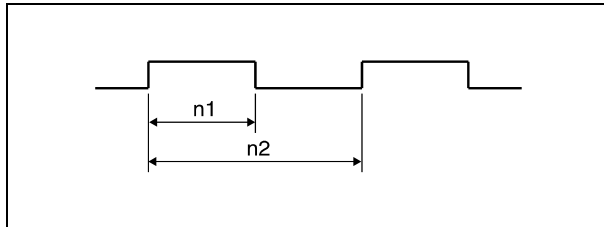
Variabili

Dati impostati	Significato	Tipo dati
n1	Indirizzo dell'operando che contiene l'impostazione del tempo di ON.	BIN 16-bit
n2	Indirizzo dell'operando che contiene l'impostazione del tempo di ciclo.	
d	Indirizzo dell'operando che contiene la destinazione di uscita.	Bit

Funzioni Modulazione larghezza impulso

PWM Istruzione di modulazione

L'istruzione PWM emette impulsi con il tempo di ciclo specificato da n2 e con il tempo di ON specificato da n1, verso l'uscita indicata da d.



I tempi espressi da n1 e n2 possono variare fra 1 e 65535 ms se si utilizza una CPU multiprocessore del System Q. Con una CPU QnA, il campo dei valori ammessi per n1 e n2 varia da 5 a 65535 ms. Il valore impostato in n1 deve essere inferiore a quello in n2.

Notas

L'istruzione PWM trasporta i dati indicati dagli operandi nell'area di lavoro della CPU ed effettua la gestione dell'uscita durante l'interruzione di sistema (1 ms per CPU del System Q, 5 ms per CPU QnA). Per questa ragione l'istruzione PWM può essere usata solo una volta nel programma.

L'istruzione non viene eseguita nei casi seguenti:

- se n1 e n2 sono entrambi 0
- se n2 è inferiore o uguale a n1
- se n1 e n2 non sono multipli di 5 (solo se si utilizza una CPU QnA)

Esempio di programma

PWM

Se X0 è attivo, il programma che segue emette impulsi con un tempo di ciclo di 1 secondo, con tempo di ON di 100 ms, su Y20.

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
<pre> LD X0 PWM K100 K1000 Y20 </pre>		<pre> LD X0 PWM_M 100, 1000, Y20 </pre>

6.8.10 MTR

CPU

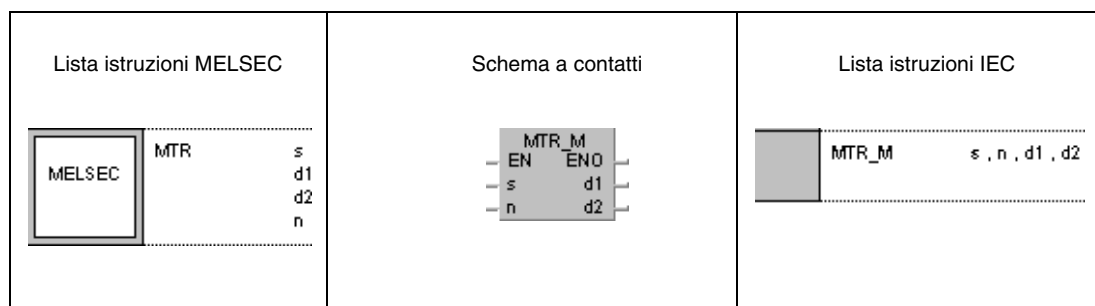
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

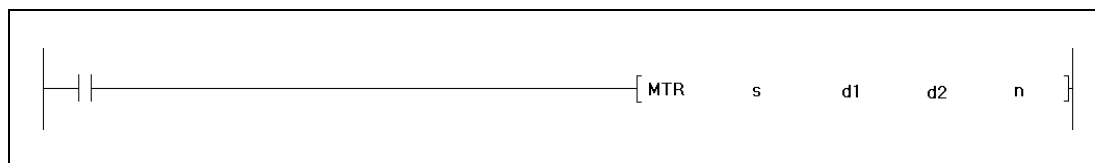
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti			Altro
	Bit	Word		Bit	Word						
s	●	—	—	—	—	—	—	—	SM0	5	
d1	●	—	—	—	—	—	—	—			
d2	●	—	—	—	—	—	—	—			
n	●	●	●	●	●	●	●	—			

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Primo operando d'ingresso	Bit
d1	Primo operando d'uscita	
d2	Primo numero dell'operando che memorizza la matrice di ingressi.	
n	Numero di righe degli ingressi	BIN 16-bit

Funzioni **Implementazione matrice di ingressi****MTR** **Istruzione per la lettura di n righe di dati in una matrice d'ingresso.**

L'istruzione MTR legge informazioni a 16 bit iniziando dall'operando indicato da s. Il numero di ripetizioni (righe) è indicato da n. I valori dei dati letti sono memorizzati a partire dall'operando indicato da d2. In questo modo si implementa una matrice da 16 bit su n righe.

Ad ogni scansione può essere letta una riga (16 bit).

Il processo di lettura viene ripetuto continuamente dalla prima all'ultima riga.

Dato il formato della matrice di ingresso (16 bit x n righe), l'operando indicato da d2 deve avere spazio per le n righe da 16 bit per memorizzare i dati.

Le righe vengono selezionate iniziando dall'uscita indicata da d1. L'uscita corrispondente a ciascuna riga di 16 bit da leggere, viene attivata o disattivata automaticamente dal sistema. Il numero di uscite corrisponde al numero di righe. Per questo, ogni riga può essere indirizzata con precisione dal sistema.

Gli indirizzi degli operandi indicati da s, d1 e d2 devono essere divisibili per 16.

Il numero di righe n può variare fra 2 e 8.

Notare che l'istruzione MTR opera direttamente sui dati di ingresso e uscita.

Errori di esecuzione

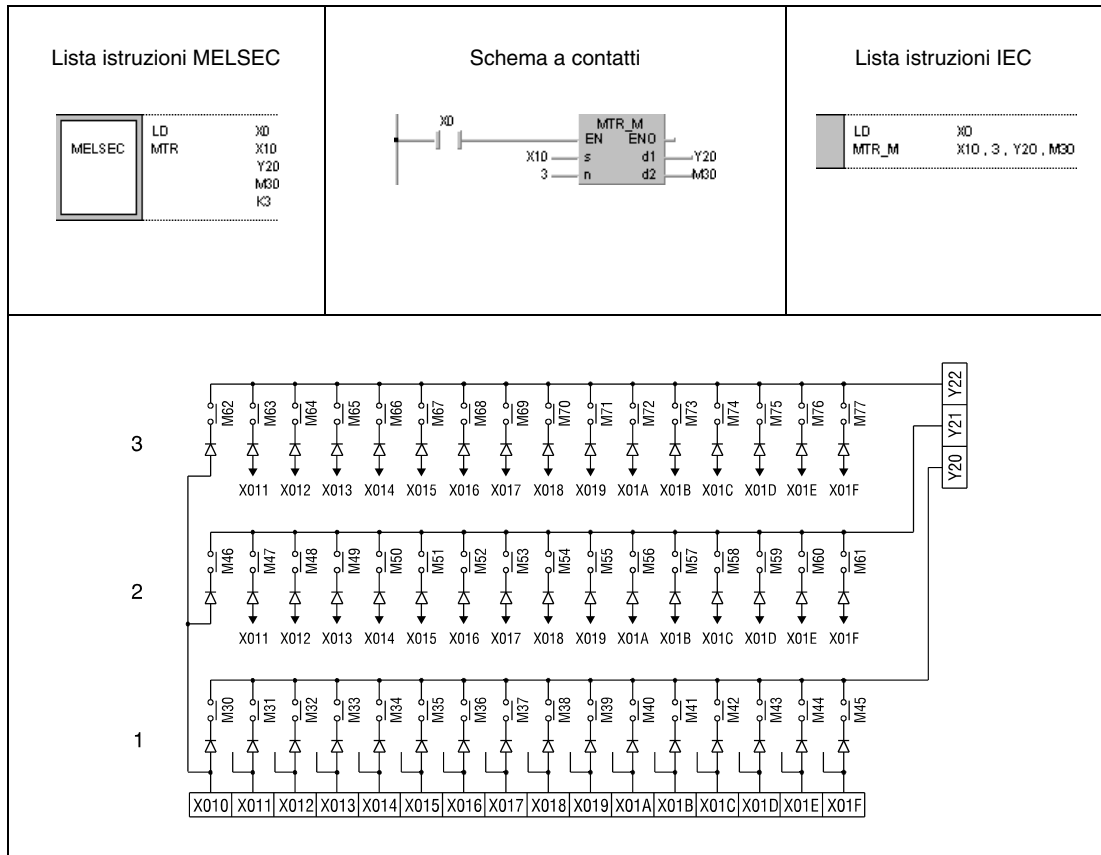
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Gli indirizzi degli operandi indicati da s, d1 e d2 devono essere divisibili per 16 (codice di errore: 4101).
- L'operando indicato da s eccede il campo di ingresso attuale (codice di errore: 4101).
- L'operando indicato da s eccede il campo di uscita attuale (codice di errore: 4101).
- Lo spazio della matrice di n righe da 16 bit eccede il campo operando rilevante per s2 (codice di errore: 4101).
- Il valore di n non è compreso fra 2 e 8 (codice di errore: 4100).

Esempio di programma

MTR

Se X0 è attivo, il programma seguente legge gli ingressi da X10 a X1F per tre volte e memorizza i risultati da M30 a M77. Viene costruita una matrice di 3 righe da 16 bit. Le righe vengono indirizzate tramite le uscite da Y20 a Y22.



- 1 Prima riga
- 2 Seconda riga
- 3 Terza riga

7 Istruzioni applicative, parte 2

Le istruzioni applicative, parte seconda, sono funzioni specifiche per diverse funzioni speciali. La tabella seguente mostra la suddivisione di queste funzioni:

Istruzione	Significato
Istruzioni per operazioni logiche	AND / OR logico, OR / NOR esclusivo logico,
Istruzioni di rotazione	Rotazione destra / sinistra dati a 16 e 32 bit
Istruzioni di scorrimento	Scorrimento dati di bit o parole
Istruzioni di elaborazione bit	Set, reset, e test di bit
Istruzioni per elaborazione dati	Ricerca, codifica e decodifica dati di operandi specificati, disgrega e aggrega dati
Istruzioni per programmi strutturati	Operazione ripetuta, chiamata subroutine di programma, richiamo subroutine fra file di programma, commutazione fra programma principale e sottoprogramma, chiamate programmi microcomputer, indicizzazione schema a contatti, memorizzazione valori di indicizzazione in tabelle dati
Istruzioni per operazioni su tabelle dati	Scrittura e lettura dati da una tabella, cancellazione e inserimento di blocchi dati in una tabella
Istruzioni di accesso a buffer di memoria	Accesso a buffer di memoria da parte di moduli funzione speciali o moduli remoti
Istruzioni di visualizzazione	Invio di caratteri ASCII sull'uscita di un modulo o su un display a LED
Istruzioni per debug e diagnosi guasti	Controllo guasti, set e reset di memorie di stato, traccia campionamento, traccia programma
Istruzioni di elaborazione stringhe di caratteri	Elaborazione stringhe di caratteri (codice ASCII)
Istruzioni per funzioni speciali	Funzioni trigonometriche, radice quadrata e calcolo esponenziale con dati BCD e dati in virgola mobile
Istruzioni per controllo dati	Controllo limite inferiore e superiore e memorizzazione dei dati controllati
Istruzioni di commutazione file registri	Commutazione fra blocchi di registri e file registri
Istruzioni per orologio	Scrittura e lettura dei dati orologio
Istruzioni per dispositivi periferici	Invio messaggi e acquisizione tasti da unità periferiche
Istruzioni di programma	Seleziona diversi modi di esecuzione del programma
Altre istruzioni	Reset timer di watchdog (WDT), set e reset riporto, generazione impulso, lettura diretta da registri file ad accesso indiretto, ingresso tasti numerici da tastiera, salvataggio o ripristino in blocco di registri indice, scrittura dei file registri su EEPROM

7.1 Istruzioni per operazioni logiche

Le istruzioni logiche consentono la programmazione di collegamenti logici, quali somme o prodotti logici.

Funzione	Istruzione MELSEC in editor MELSEC	Istruzione MELSEC in editor IEC
AND (prodotto logico)	WAND	WAND_M, WAND_3_M
	WANDP	WANDP_M, WANDP_3_M
	DAND	DAND_M, DAND_3_M
	DANDP	DANDP_M, DANDP_3_M
	BKAND	BKAND_M
	BKANDP	BKANDP_M
OR (somma logica)	WOR	WOR_M, WOR_3_M
	WORP	WORP_M, WORP_3_M
	DOR	DOR_M, DOR_3_M
	DORP	DORP_M, DORP_3_M
	BKOR	BKOR_M
	BKORP	BKORP_M
OR esclusivo (XOR)	WXOR	WXOR_M, WXOR_3_M
	WXORP	WXORP_M, WXORP_3_M
	DXOR	DXOR_M, DXOR_3_M
	DXORP	DXORP_M, DXORP_3_M
	BKXOR	BKXOR_M
	BKXORP	BKXORP_M
NOR esclusivo (XNR)	WXNR	WXNR_M, WXNR_3_M
	WXNRP	WXNRP_M, WXNRP_3_M
	DXNR	DXNR_M, DXNR_3_M
	DXNRP	DXNRP_M, DXNRP_3_M
	BKXNR	BKXNR_M
	BKXNRP	BKXNRP_M

Le istruzioni logiche sono elaborate bit a bit, come dati binari. Le due condizioni (0 e 1) vengono collegate e il risultato del collegamento viene inviato all'indirizzo di destinazione.

NOTA

Utilizzare istruzioni IEC con gli editor IEC

La tabella seguente mostra i risultati dei collegamenti logici degli stati 0 e 1. A e B sono variabili d'ingresso, mentre Y è la variabile di uscita.

Collegamento logico	Dettagli di elaborazione	Funzionamento Espressione	Esempio		
			A	B	Y
AND logico	L'uscita Y è attivata a 1 solo se entrambi gli ingressi A e B sono attivi (1).	$Y = A \times B$	0	0	0
			0	1	0
			1	0	0
			1	1	1
OR logico	L'uscita Y è attivata a 1, se almeno uno dei due ingressi A o B è attivo (1).	$Y = A + B$	0	0	0
			0	1	1
			1	0	1
			1	1	1
OR esclusivo logico (XOR)	L'uscita Y è attiva a 1, se gli ingressi A e B sono diversi, mentre viene disattivata (0) se A e B sono uguali.	$Y = \bar{A} \times B + A \times \bar{B}$	0	0	0
			0	1	1
			1	0	1
			1	1	0
NOR esclusivo logico (XNR)	L'uscita Y è attiva a 1, se gli ingressi A e B sono uguali, mentre viene disattivata (0) se A e B sono diversi.	$Y = (\bar{A} + B) (A + \bar{B})$	0	0	1
			0	1	0
			1	0	0
			1	1	1

² Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.

Se viene usata una CPU QnA:

4

Se viene usata una CPU multiprocessore del System Q con operandi interni a word (tranne file registri ZR) o costanti:

6

Se viene usata una CPU multiprocessore del System Q con operandi a bit il cui numero operando sia multiplo di 16, la cui designazione di cifra è K4, e che non usa indicizzazione:

6

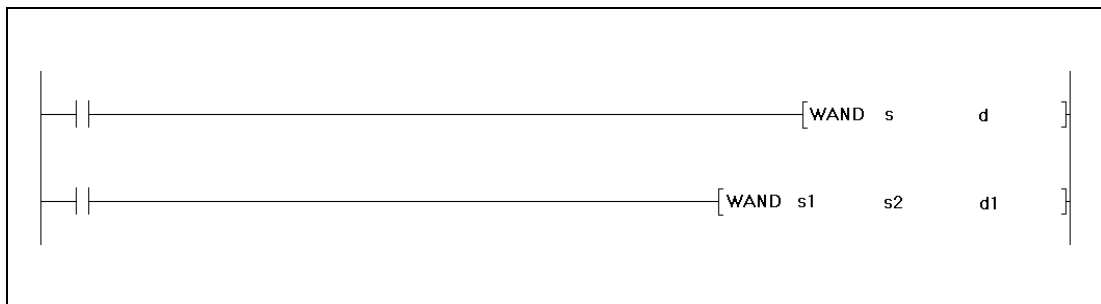
Se viene usata una CPU del System Q con operandi diversi da quelli menzionati:

4

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC											
<table border="1"> <tr> <td>MELSEC</td> <td>WAND</td> <td>s</td> <td>d</td> </tr> <tr> <td>MELSEC</td> <td>WAND</td> <td>s1</td> <td>s2</td> <td>d1</td> </tr> </table>	MELSEC	WAND	s	d	MELSEC	WAND	s1	s2	d1		<table border="1"> <tr> <td>WAND_3_M</td> <td>s1, s2, d1</td> </tr> </table>	WAND_3_M	s1, s2, d1
MELSEC	WAND	s	d										
MELSEC	WAND	s1	s2	d1									
WAND_3_M	s1, s2, d1												

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Dato del prodotto logico, o primo indirizzo dell'operando che contiene questo dato.	BIN 16/32-bit
d		
s1	Dato per il prodotto logico, o primo indirizzo dell'operando che contiene questo dato.	
s2		
d1 (per DAND d)	Primo indirizzo di operando che contiene il risultato dell'operazione logica	

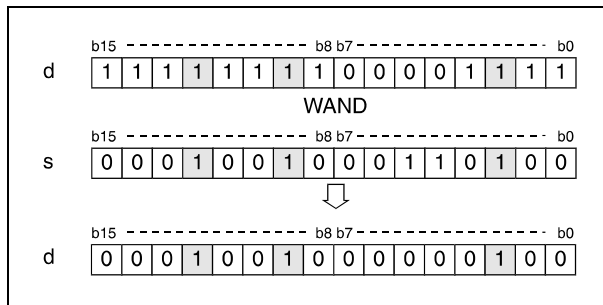
Funzioni AND logico

WAND Dati 16-bit

L'operazione di AND logico fornisce il prodotto logico di due variabili.

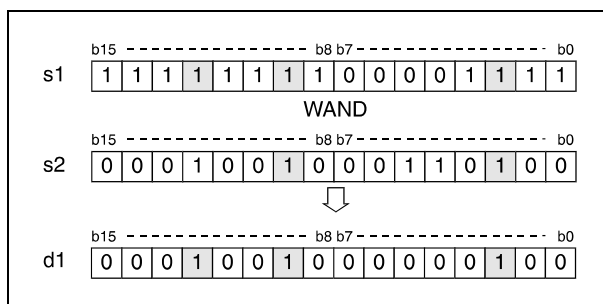
● Variante 1:

I dati a 16 bit indicati da s e d formano il prodotto logico calcolato bit a bit. Il risultato viene inviato all'operando indicato da d.



● Variante 2:

I dati a 16 bit indicati da s1 e 2 formano il prodotto logico calcolato bit a bit. Il risultato viene inviato all'operando indicato da d1.

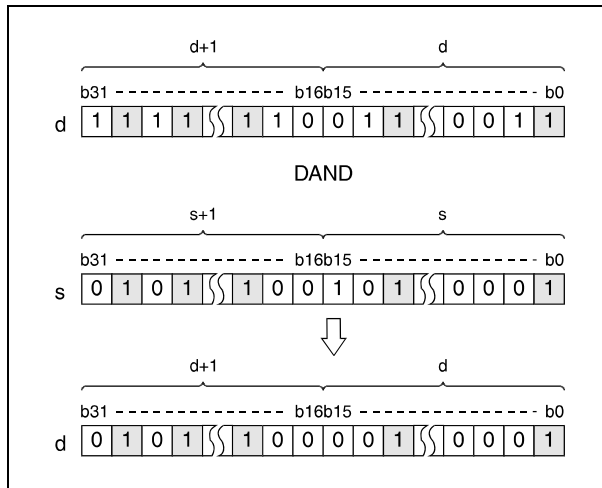


I bit che superano l'indicazione di cifra vengono impostati a 0. Ad esempio, se la cifra è indicata da K2, gli 8 bit più pesanti (da b8 a b15) sono considerati a 0.

DAND Dati 32-bit

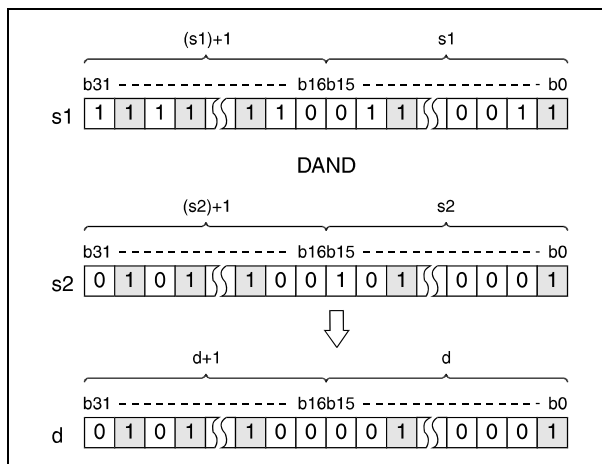
● Variante 1:

I dati a 32 bit indicati da s e d formano il prodotto logico calcolato bit a bit. Il risultato viene inviato all'operando indicato da d.



● Variante 2 (serie Q e System Q):

I dati a 32 bit indicati da s1 e 2 formano il prodotto logico calcolato bit a bit. Il risultato viene inviato all'operando indicato da d.

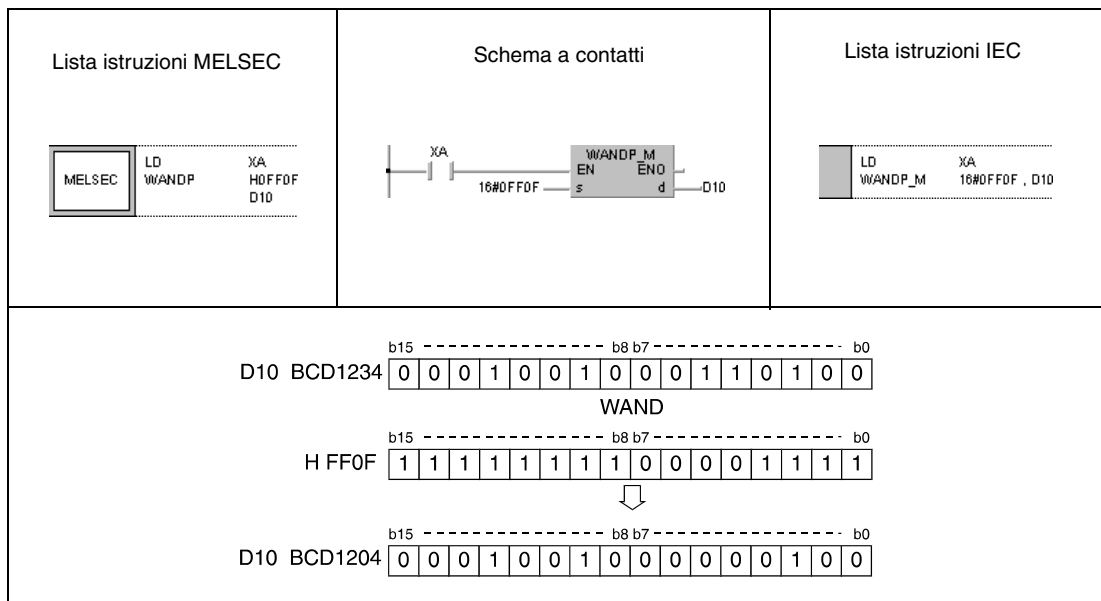


Dopo aver eseguito l'operazione, tutti i bit oltre la cifra indicata sono impostati a 0.

Programma di esempio 1

WANDP (s, d)

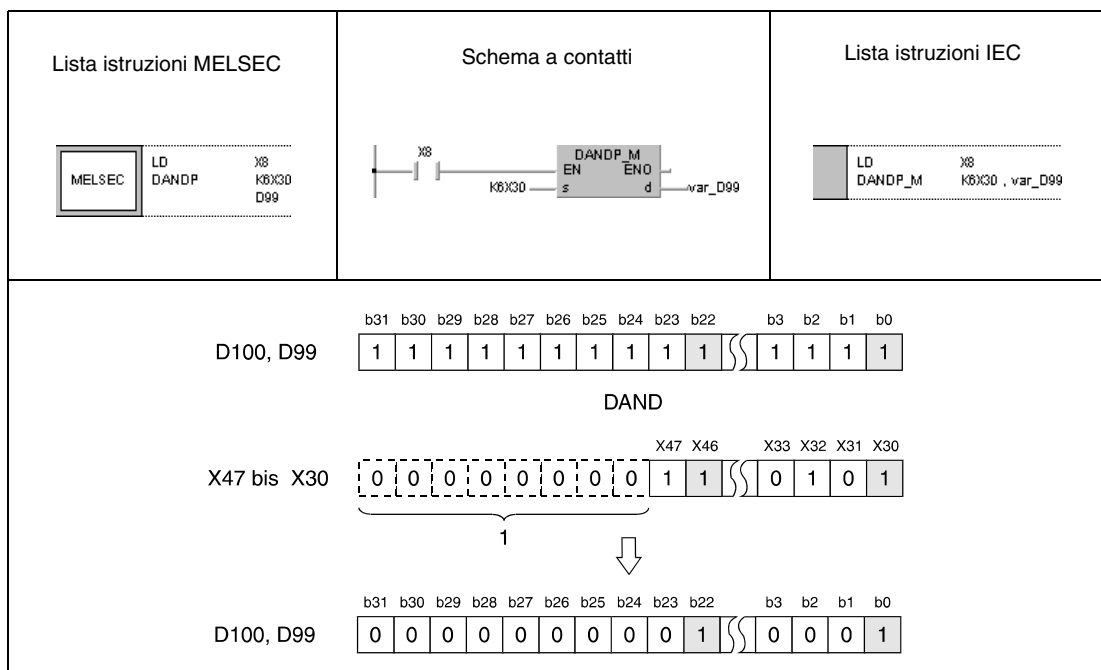
Con il fronte positivo di XA, il programma che segue imposta a 0 la cifra delle decine (b4-b7) del valore BCD a 4-cifre contenuto in D10. Il risultato viene memorizzato ancora in D10.



Programma di esempio 2

DANDP (s, d)

Con il fronte positivo di X8, il programma che segue forma il prodotto logico del dato a 32 bit in D99 e D100 con il dato a 24 bit da X30 a X47. Il risultato viene memorizzato nuovamente in D99 e D100.

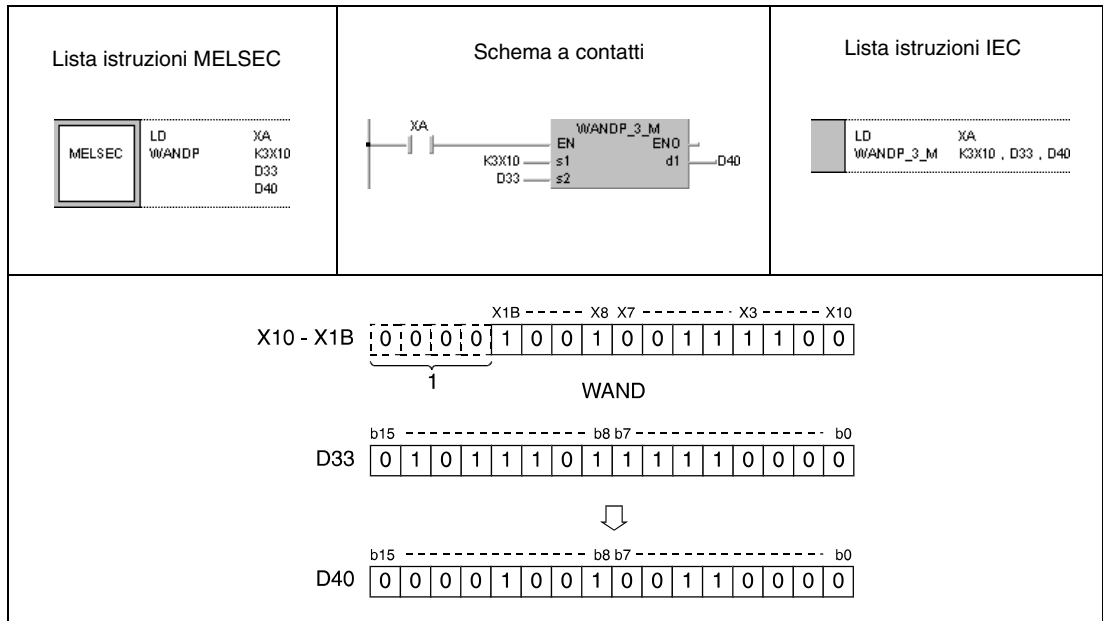


¹ Questi bit vengono impostati a 0.

Programma di esempio 3

WANDP (s1, s2, d1)

Con il fronte positivo di XA, il programma che segue forma il prodotto logico del dato contenuto da X10 a X1B e del dato in D33. Il risultato è memorizzato in D40.

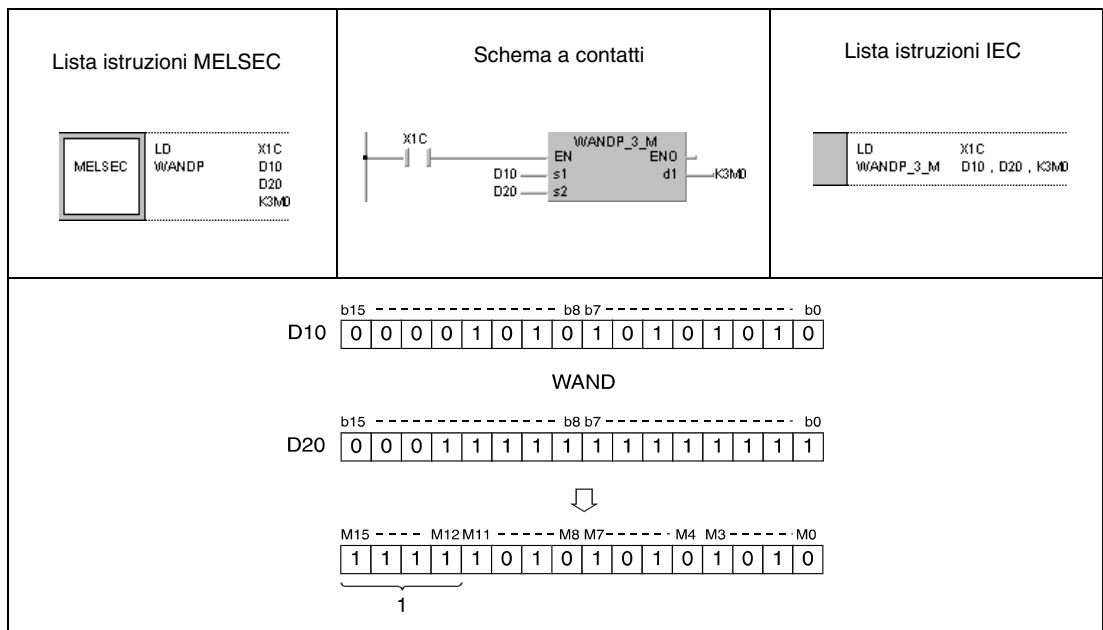


¹ Questi bit vengono impostati a 0.

Programma di esempio 4

WANDP (s1, s2, d1)

Con il fronte positivo di X1C, il programma che segue forma il prodotto logico dei dati in D10 e D20. Il risultato viene memorizzato da M0 a M11.



¹ Questi bit non vengono modificati.

7.1.2 BKAND, BKANDP

CPU

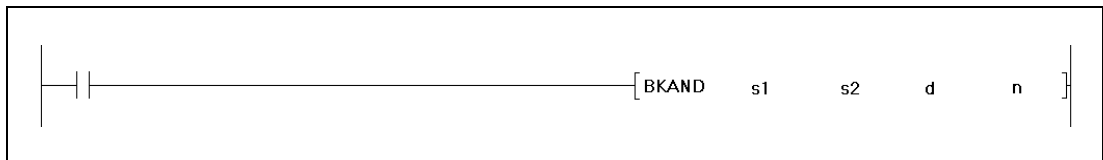
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi
MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	SM0	5	
s2	—	●	●	—	—	—	—	●			
d	—	●	●	—	—	—	—	—			
n	●	●	●	●	●	●	●	●			

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC										
<div style="border: 1px solid black; padding: 5px; width: fit-content;"> MELSEC </div> <table style="margin-left: 20px;"> <tr><td>BKAND</td><td>s1</td></tr> <tr><td></td><td>s2</td></tr> <tr><td></td><td>d</td></tr> <tr><td></td><td>n</td></tr> </table>	BKAND	s1		s2		d		n		<table style="margin-left: 20px;"> <tr><td>BKAND_M</td><td>s1, s2, n, d</td></tr> </table>	BKAND_M	s1, s2, n, d
BKAND	s1											
	s2											
	d											
	n											
BKAND_M	s1, s2, n, d											

GX
Developer

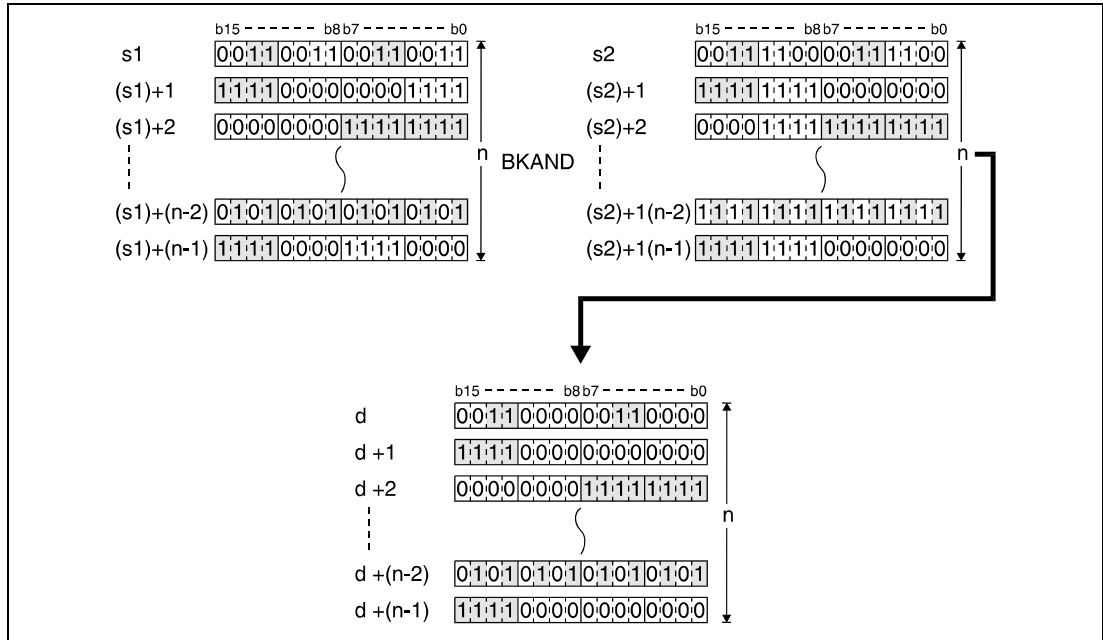
Variabili

Dati impostati	Significato	Tipo dati
s1	Primo indirizzo di operando che contiene il dato per il prodotto logico.	BIN 16-bit
s2	Primo indirizzo del dato o primo indirizzo dell'operando che contiene il dato per l'operazione logica.	
d	Primo indirizzo di operando che contiene il risultato dell'operazione logica	
n	Numero dei blocchi dati che formano il prodotto logico.	

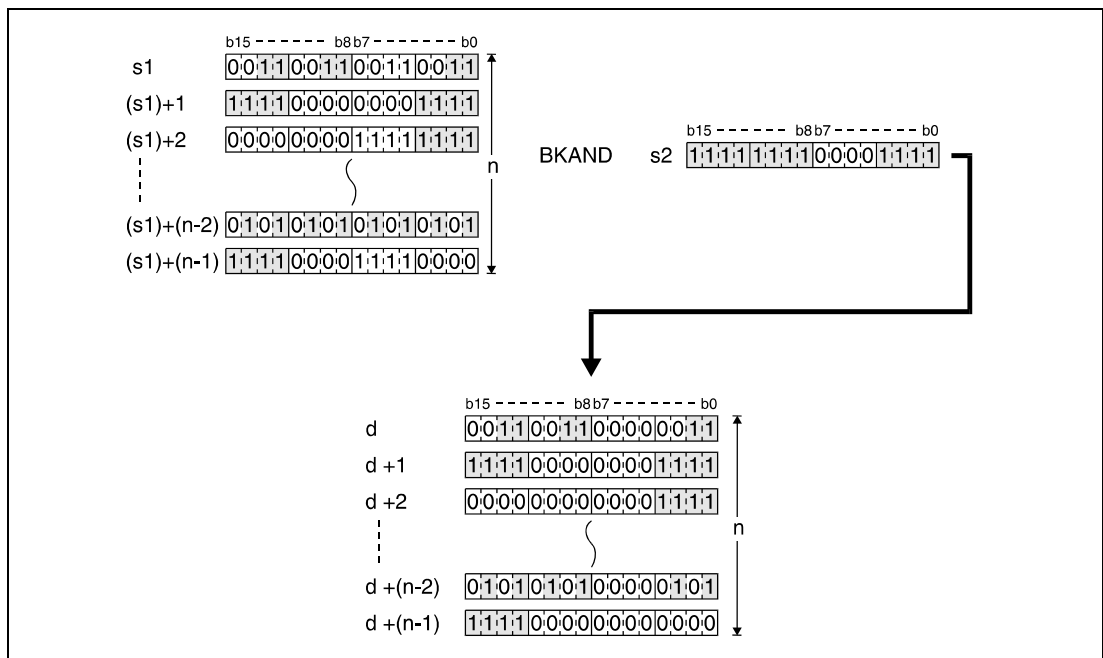
Funzioni **Formazione di prodotto logico con blocchi dati da 16 bit**

BKAND **Formazione prodotto logico con blocchi dati**

L'istruzione BKAND forma il prodotto logico della serie di blocchi iniziando dai blocchi dati a 16 bit indicati rispettivamente da s1 e s2. I vari blocchi a 16 bit del risultato sono memorizzati dall'operando d in avanti. Il numero di dati da elaborare è indicato in n.



La costante s2 può variare nel campo fra -32768 e 32767.



7.1.3 WOR, WOPR, DOR, DORP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

Operandi utilizzabili																	Definizione cifra	Numero di passi	Indice	Riporto	Errore						
Operandi a bit					Operandi a word (16-bit)						Costanti	Punta-tore	Livello														
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K						H (16#)	P	I	N		
WOR																											
s	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●					K1 ↓ K4	5	1			
d		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●							7	1			
s1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●						9	1	●		●
s2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●										
d1		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●											
DOR																											
s	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●					K1 ↓ K8	9	1	●		●
d		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●											

¹ Consultare la sezione "Programmazione con CPU AnA, AnAS e AnU" del manuale di programmazione per il corrispondente numero di passi.

Operandi MELSEC Q

Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word				U			
WOR											
s	●	●	●	●	●	●	●	●	—	—	3
d	●	●	●	●	●	●	●	—	—	—	
s1	●	●	●	●	●	●	●	●	—	—	4
s2	●	●	●	●	●	●	●	●	—	—	
d1	●	●	●	●	●	●	●	—	—	—	
DOR											
s	●	●	●	●	●	●	●	●	—	—	4 ¹⁾
d	●	●	●	●	●	●	●	—	—	—	
s1	●	●	●	●	●	●	●	●	—	—	4 ²⁾
s2	●	●	●	●	●	●	●	●	—	—	
d	●	●	●	●	●	●	●	—	—	—	

¹ Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.

Se viene usata una CPU QnA:	4
Se viene usata una CPU a singolo processore del System Q	3
Se viene usata una CPU multiprocessore del System Q con operandi interni a word (tranne registro file ZR) o costanti:	6
Se viene usata una CPU multiprocessore del System Q con operandi a bit il cui numero operando sia multiplo di 16, con indicazione di cifra K4, e che non usa indicizzazione:	6
Se viene usata una CPU multiprocessore del System Q con operandi diversi da quelli menzionati:	4

- ² Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.
- Se viene usata una CPU QnA: 4
 - Se viene usata una CPU multiprocessore del System Q con operandi interni a word (tranne file registri ZR) o costanti: 6
 - Se viene usata una CPU multiprocessore del System Q con operandi a bit il cui numero operando sia multiplo di 16, la cui designazione di cifra è K4, e che non usa indicizzazione: 6
 - Se viene usata una CPU del System Q con operandi diversi da quelli menzionati: 4

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC								
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;">MELSEC</td> <td>WOR</td> <td>s d</td> </tr> <tr> <td style="text-align: center;">MELSEC</td> <td>WOR</td> <td>s1 s2 d1</td> </tr> </table>	MELSEC	WOR	s d	MELSEC	WOR	s1 s2 d1		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;">WOR_3_M</td> <td>s1, s2, d1</td> </tr> </table>	WOR_3_M	s1, s2, d1
MELSEC	WOR	s d								
MELSEC	WOR	s1 s2 d1								
WOR_3_M	s1, s2, d1									

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Dato per la somma logica, o primo indirizzo dell'operando che contiene questo dato.	BIN 16/32-bit
d		
s1	Dato per la somma logica, o primo indirizzo dell'operando che contiene questo dato.	
s2		
d1 (per DOR d)	Primo indirizzo di operando che contiene il risultato dell'operazione logica	

Funzioni

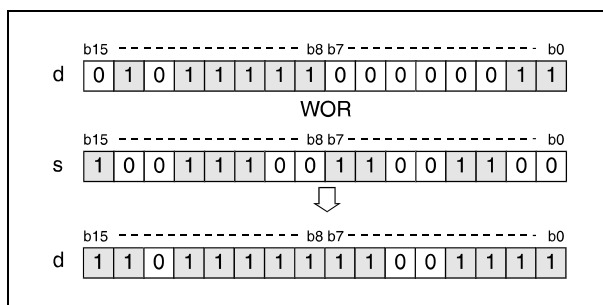
OR logico

WOR Dati 16-bit

L'operazione di OR logico fornisce la somma logica di due variabili di ingresso.

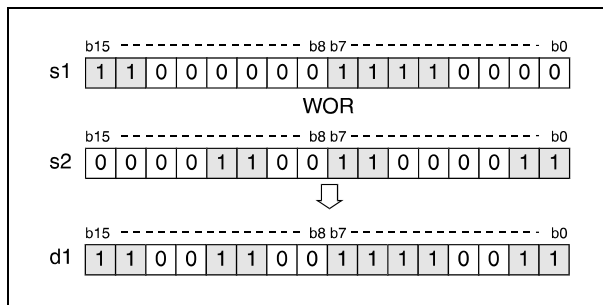
● Variante 1:

I dati a 16 bit indicati da s e d vengono sommati bit a bit. Il risultato viene inviato all'operando indicato da d.



● Variante 2:

I dati a 16 bit indicati da s1 e s2 vengono sommati bit a bit. Il risultato viene inviato all'operando indicato da d1.

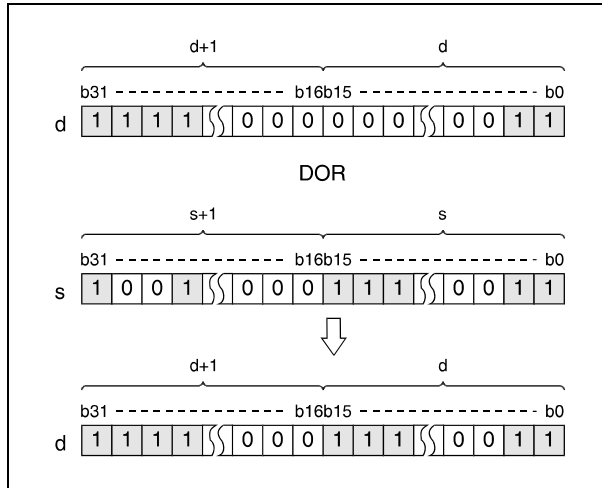


I bit che superano l'indicazione di cifra vengono impostati a 0. Ad esempio, se la cifra è indicata da K2, gli 8 bit più pesanti (da b8 a b15) sono considerati a 0.

DOR Dati 32-bit

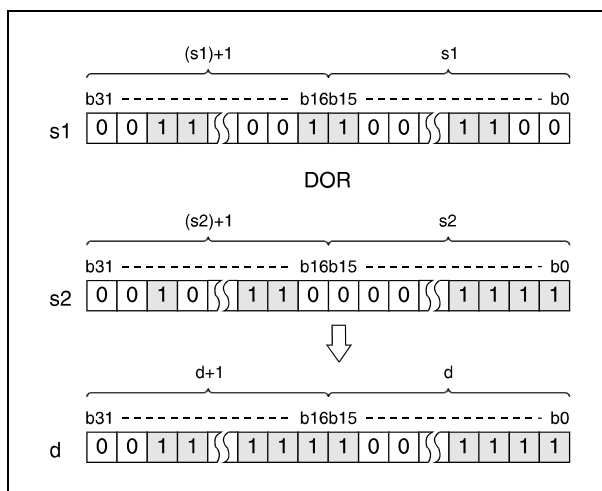
● Variante 1:

I dati a 32 bit indicati da s e d vengono sommati bit a bit. Il risultato viene inviato all'operando indicato da d.



● Variante 2 (serie Q e System Q):

I dati a 32 bit indicati da s1 e s2 vengono sommati bit a bit. Il risultato viene inviato all'operando indicato da d.

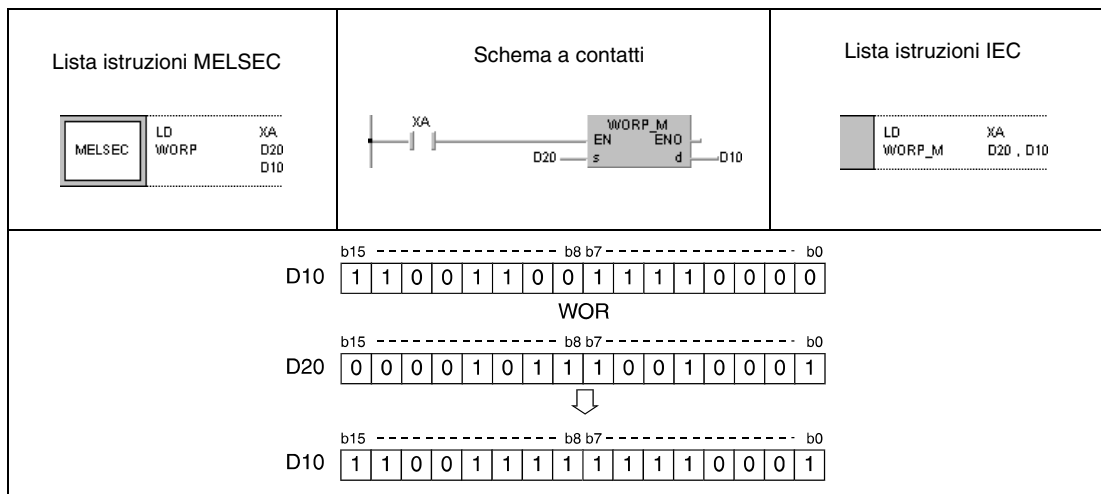


Dopo aver eseguito l'operazione, tutti i bit oltre la cifra indicata sono impostati a 0.

Programma di esempio 1

WORP (s, d)

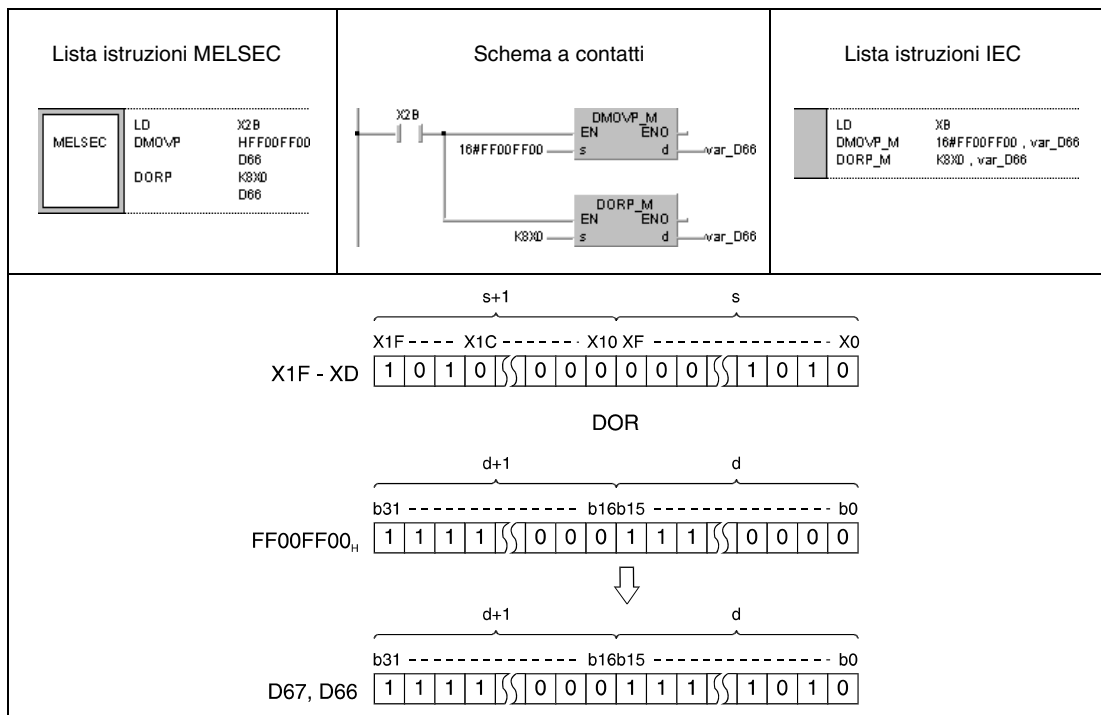
Il programma che segue somma il dato in D10 con quello in D20 sul fronte di salita di XA. Il risultato è memorizzato in D10.



Programma di esempio 2

DORP (s, d)

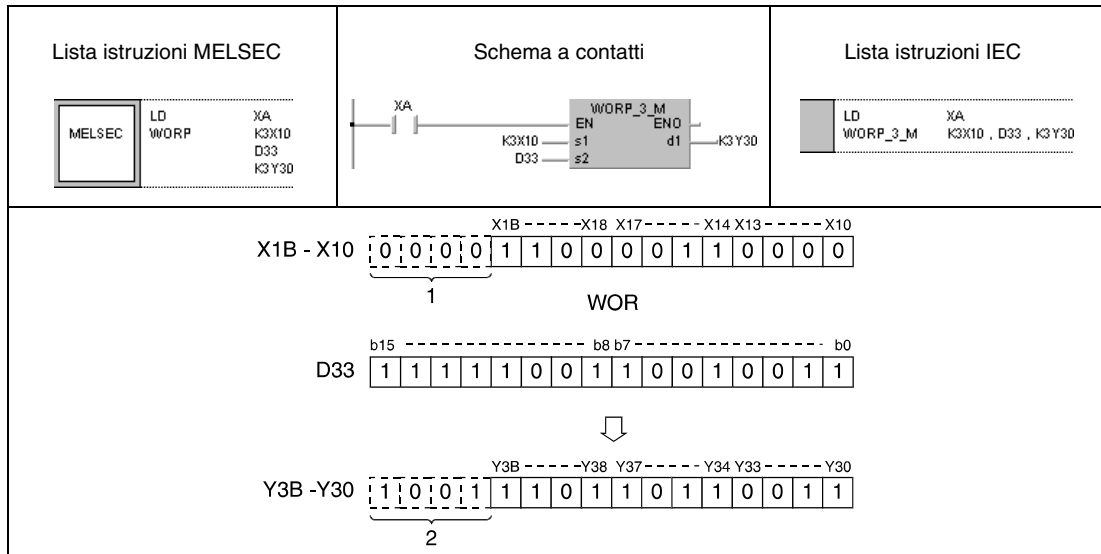
Con il fronte di salita di X2B, il programma che segue somma i dati sugli ingressi da X0 a X1F con il valore esadecimale FF00FF00. Il risultato viene memorizzato in D66 e D67.



Programma di esempio 3

WORP (s1, s2, d1)

Con il fronte positivo di XA, il programma che segue somma i dati sugli ingressi da X10 a X1B con il dato in D33. Il risultato viene inviato sulle uscite da Y30 a Y3B.

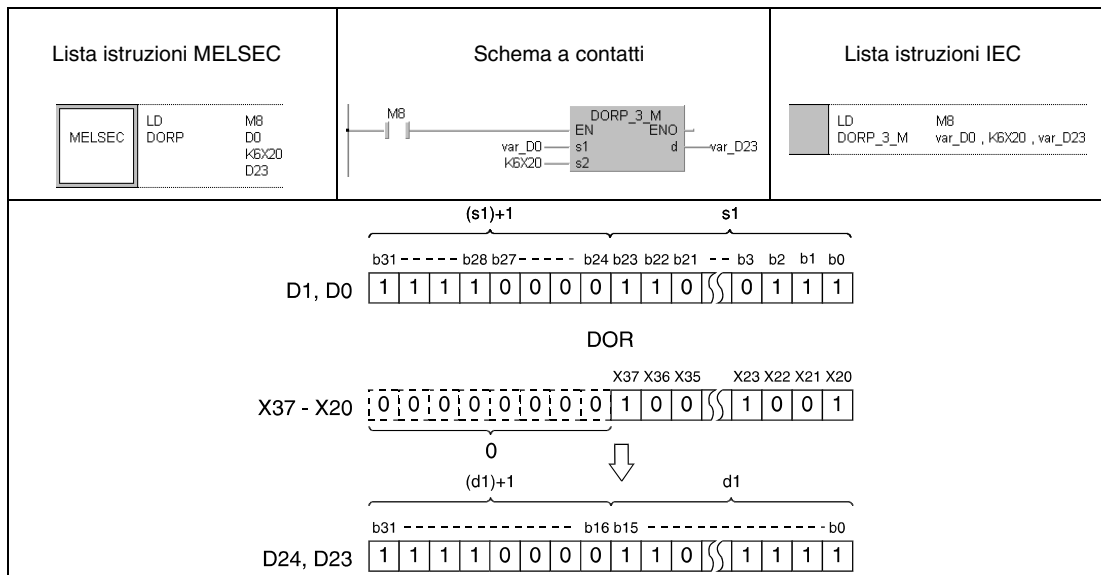


- ¹ Questi bit vengono impostati a 0
- ² Questi bit non vengono modificati

Programma di esempio 4

DORP (s1, s2, d)

Con il fronte positivo di M8, il programma che segue somma il dato a 32 bit in D0 e D1 con il dato a 24 bit sugli ingressi da X20 a X37. Il risultato viene memorizzato in D23 e D24.



NOTA

I programmi di esempio 2 e 4 non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provocano messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.1.4 BKOR, BKORP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

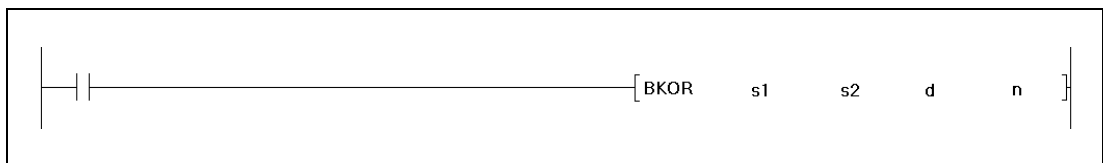
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	SM0	5	
s2	—	●	●	—	—	—	—	●			
d	—	●	●	—	—	—	—	—			
n	●	●	●	●	●	●	●	●			

GX IEC Developer

<p>Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p>MELSEC</p> </div> <p>BKOR s1 s2 d n</p>	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p>BKOR_M</p> </div> <p>s1, s2, n, d</p>
--	--------------------------	---

GX Developer



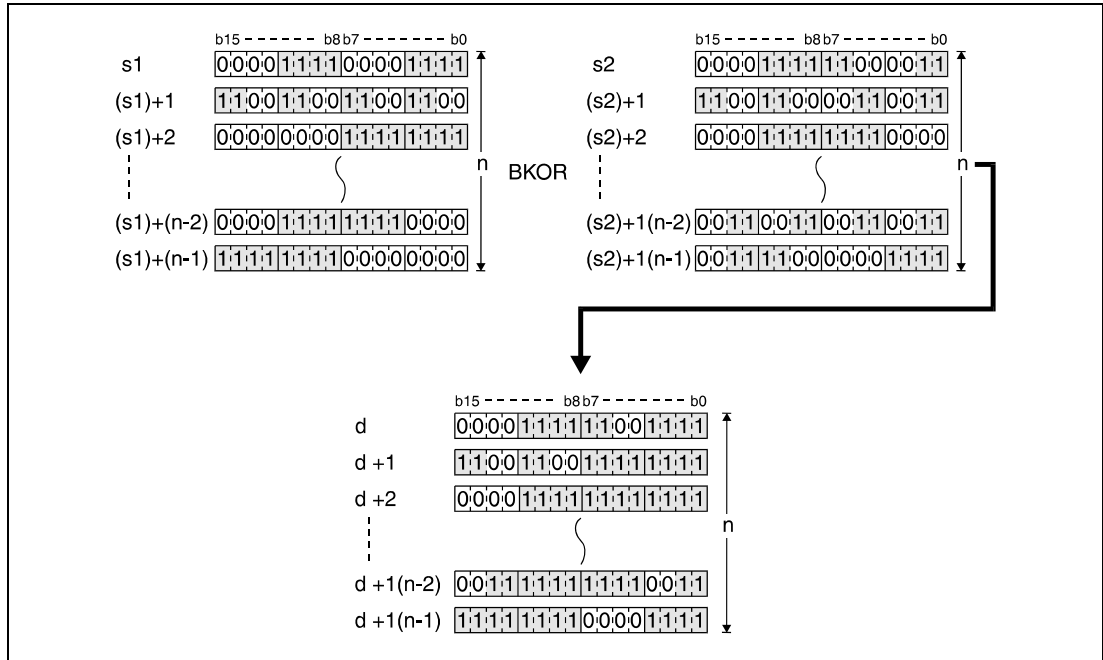
Variabili

Dati impostati	Significato	Tipo dati
s1	Primo indirizzo di operando che contiene il dato per la somma logica.	BIN 16-bit
s2	Primo indirizzo del dato o primo indirizzo dell'operando che contiene il dato per la somma logica.	
d	Primo indirizzo di operando che contiene il risultato dell'operazione logica	
n	Numero dei blocchi dati che formano la somma logica.	

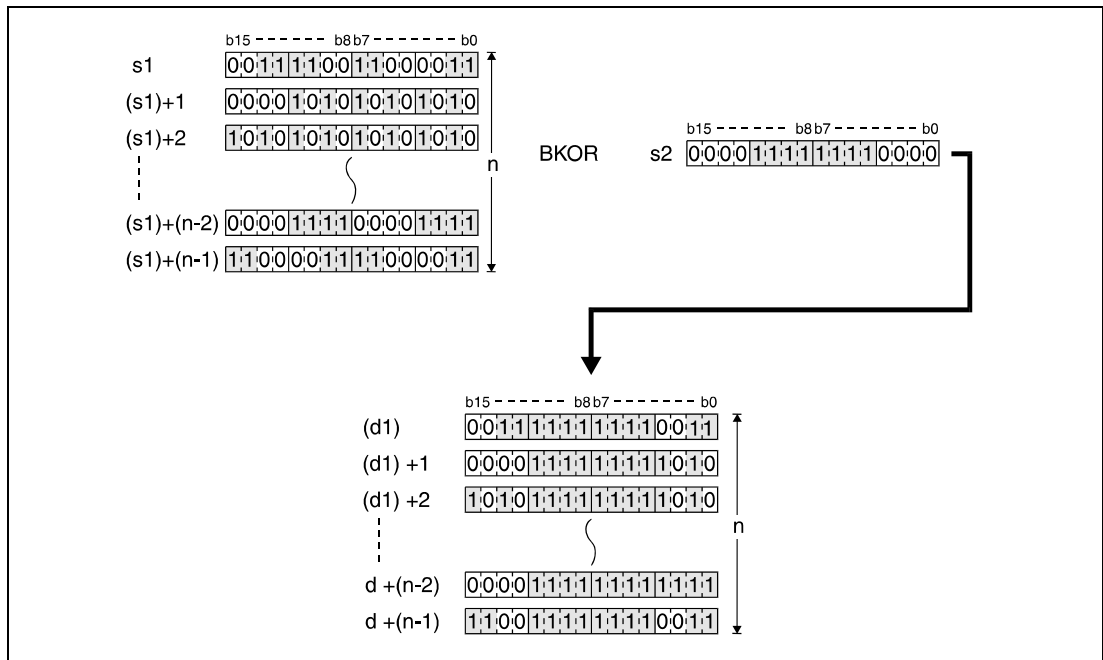
Funzioni Formazione di somma logica con blocchi dati da 16 bit

BKOR Formazione somma logica con blocchi dati

L'istruzione BKOR forma la somma logica della serie di blocchi iniziando dai blocchi dati a 16 bit indicati rispettivamente da s1 e s2. I vari blocchi a 16 bit del risultato sono memorizzati dall'operando d in avanti. Il numero di dati da elaborare è indicato in n.



La costante s2 può variare nel campo fra -32768 e 32767.



Errori di esecuzione

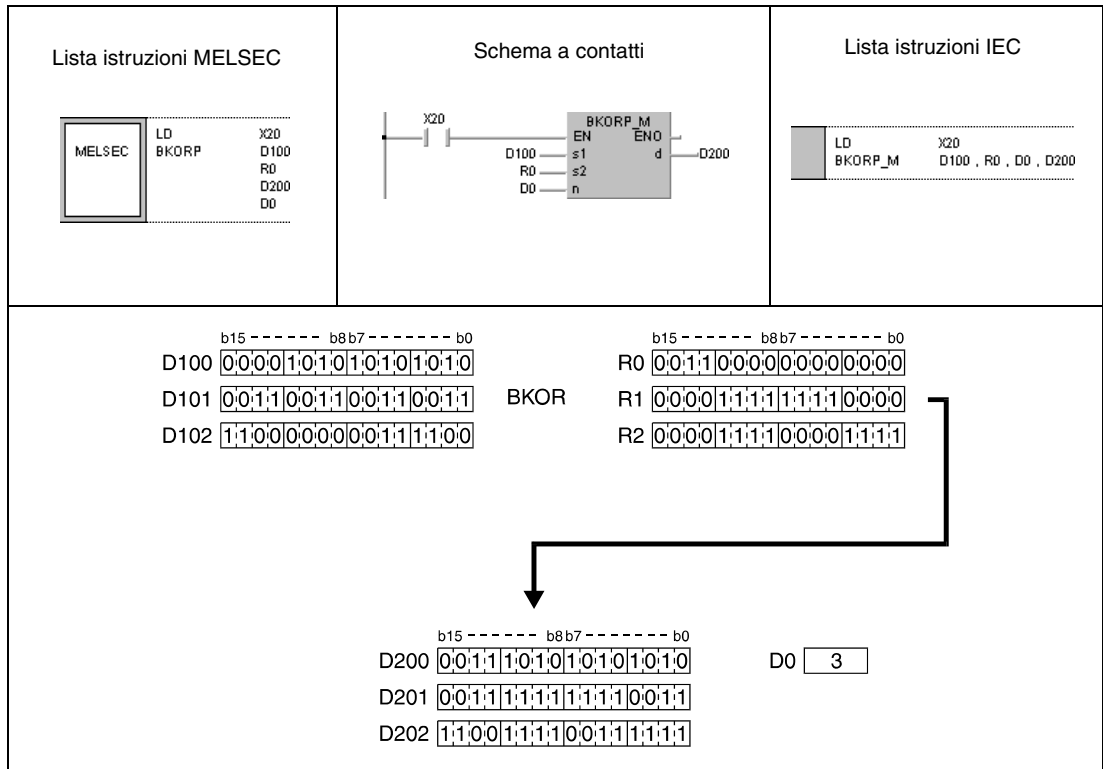
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il numero di blocchi dati indicato da n supera i numeri dispositivo indicati da s1 e s2 (codice di errore: 4101).
- I numeri degli operandi indicati da s1, s2 o d sono sovrapposti (codice di errore: 4101).

Esempio di programma

BKORP

Con il fronte positivo di X20, il programma che segue forma la somma logica del dato contenuto nei registri da D100 a D102 e del dato nei registri da R0 a R2. Il risultato viene memorizzato nei registri da D200 a D202. Il numero di blocchi da elaborare (3) è contenuto in D0.



7.1.5 WXOR, WXORP, DXOR, DXORP

CPU

AnS	AnN	AnA(S)	AnU	QnAS, Q4AR	System Q
●	●	●	●	●	●

Operandi
MELSEC A

		Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Rapporto	Errore			
		Operandi a bit						Operandi a word (16-bit)						Costanti		Punta- tore	Livello								
		X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V						K	H (16#)	P
WXOR																									
s	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●				K1 ↓ K4	5 ↓ 7	●	●
d		●	●	●	●	●	●	●	●	●	●	●	●	●	●										
s1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●					K1 ↓ K8	9 ↓ 1	●	●
s2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●									
d1		●	●	●	●	●	●	●	●	●	●	●	●	●	●										
DXOR																									
s	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●				K1 ↓ K8	9 ↓ 1	●	●
d		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●									

¹ Consultare la sezione "Programmazione con CPU AnA, AnAS e AnU" del manuale di programmazione per il corrispondente numero di passi.

Operandi
MELSEC Q

		Operandi utilizzabili								Segnale di errore	Numero di passi	
		Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□G□	Registro indice Zn	Costanti K, H (16#)			Altro
		Bit	Word		Bit	Word						
WXOR												
s	●	●	●	●	●	●	●	●	●	—	—	3
d	●	●	●	●	●	●	●	—	—	—	—	
s1	●	●	●	●	●	●	●	●	●	—	—	4
s2	●	●	●	●	●	●	●	●	●	—	—	
d1	●	●	●	●	●	●	●	—	—	—	—	
DXOR												
	●	●	●	●	●	●	●	●	●	—	—	4 ¹⁾
	●	●	●	●	●	●	●	—	—	—	—	
s1	●	●	●	●	●	●	●	●	●	—	—	4 ²⁾
s2	●	●	●	●	●	●	●	●	●	—	—	
d	●	●	●	●	●	●	●	—	—	—	—	

¹ Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.

Se viene usata una CPU QnA:

4

Se viene usata una CPU a singolo processore del System Q

3

Se viene usata una CPU multiprocessore del System Q con operandi interni a word (tranne registro file ZR) o costanti:

6

Se viene usata una CPU multiprocessore del System Q con operandi a bit il cui numero operando sia multiplo di 16, con indicazione di cifra K4, e che non usa indicizzazione:

6

Se viene usata una CPU multiprocessore del System Q con operandi diversi da quelli menzionati:

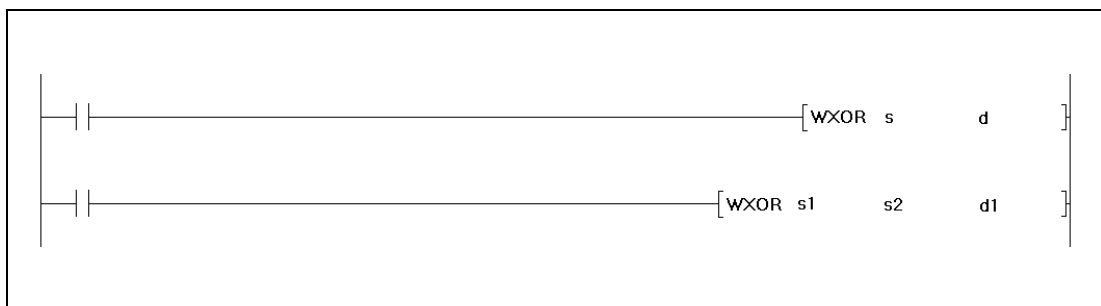
4

- ² Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.
- Se viene usata una CPU QnA: 4
 - Se viene usata una CPU multiprocessore del System Q con operandi interni a word (tranne file registri ZR) o costanti: 6
 - Se viene usata una CPU multiprocessore del System Q con operandi a bit il cui numero operando sia multiplo di 16, la cui designazione di cifra è K4, e che non usa indicizzazione: 6
 - Se viene usata una CPU del System Q con operandi diversi da quelli menzionati: 4

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC											
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center; width: 10%;">MELSEC</td> <td style="width: 10%;">WXOR</td> <td style="width: 10%; text-align: right;">s</td> <td style="width: 10%; text-align: right;">d</td> </tr> <tr> <td style="text-align: center;">MELSEC</td> <td style="width: 10%;">WXOR</td> <td style="width: 10%; text-align: right;">s1</td> <td style="width: 10%; text-align: right;">s2 d1</td> </tr> </table>	MELSEC	WXOR	s	d	MELSEC	WXOR	s1	s2 d1		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10%;"></td> <td style="width: 10%;">WXOR_3_M</td> <td style="width: 10%; text-align: right;">s1 , s2 , d1</td> </tr> </table>		WXOR_3_M	s1 , s2 , d1
MELSEC	WXOR	s	d										
MELSEC	WXOR	s1	s2 d1										
	WXOR_3_M	s1 , s2 , d1											

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Dato per l'operazione di OR esclusivo, o primo indirizzo dell'operando che contiene questo dato.	BIN 16/32-bit
d		
s1	Dato per l'operazione di OR esclusivo, o primo indirizzo dell'operando che contiene questo dato.	
s2		
d1 (per DXOR d)	Primo indirizzo di operando che contiene il risultato dell'operazione logica	

Funzioni

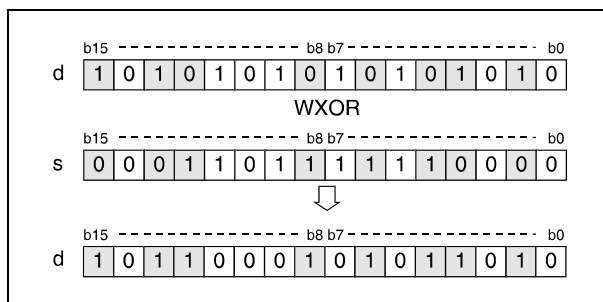
OR esclusivo logico

WXOR Dati 16-bit

L'operazione di OR esclusivo forma la seguente somma di prodotti logici di due variabili d'ingresso ($Y = (\bar{A} \times B) + (A \times \bar{B})$).

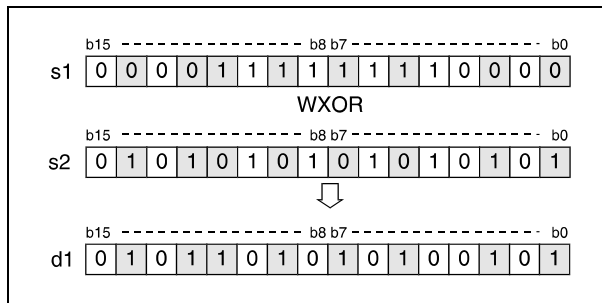
● Variante 1:

I dati a 16 bit indicati da s e da d producono l'OR esclusivo risultante dall'operazione. Il risultato viene inviato all'operando indicato da d.



● Variante 2:

I dati a 16 bit indicati da s1 e da s2 producono l'OR esclusivo risultante dall'operazione. Il risultato viene inviato all'operando indicato da d.

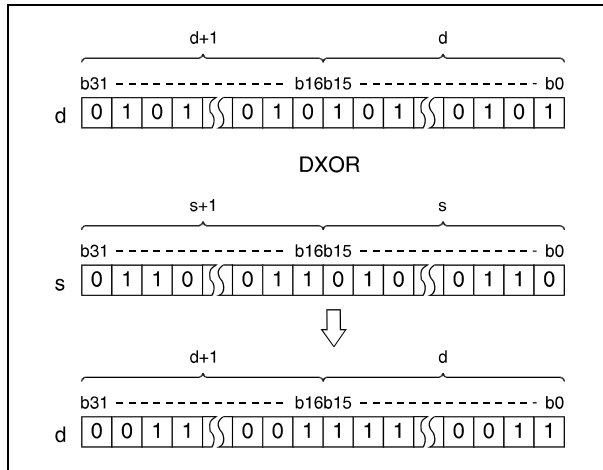


I bit che superano l'indicazione di cifra vengono impostati a 0. Ad esempio, se la cifra è indicata da K2, gli 8 bit più pesanti (da b8 a b15) sono considerati a 0.

DXOR Dati 32-bit

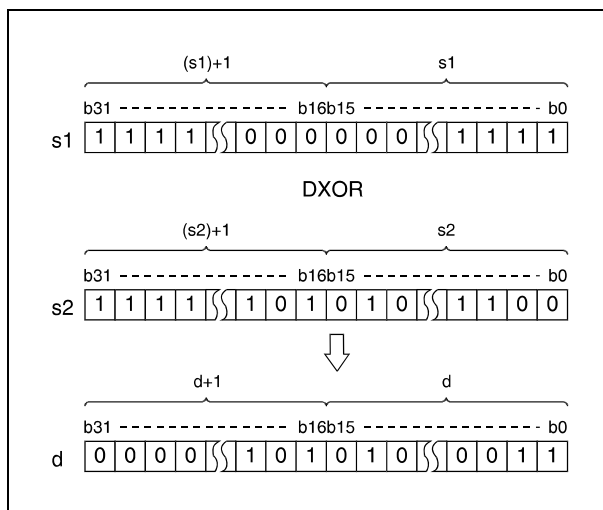
● Variante 1:

I dati a 32 bit indicati da s e da d producono l'OR esclusivo risultante dall'operazione. Il risultato viene inviato all'operando indicato da d.



● Variante 2 (serie Q e System Q):

I dati a 32 bit indicati da s1 e da s2 producono l'OR esclusivo risultante dall'operazione. Il risultato viene inviato all'operando indicato da d.



Dopo aver eseguito l'operazione, tutti i bit oltre la cifra indicata sono impostati a 0.

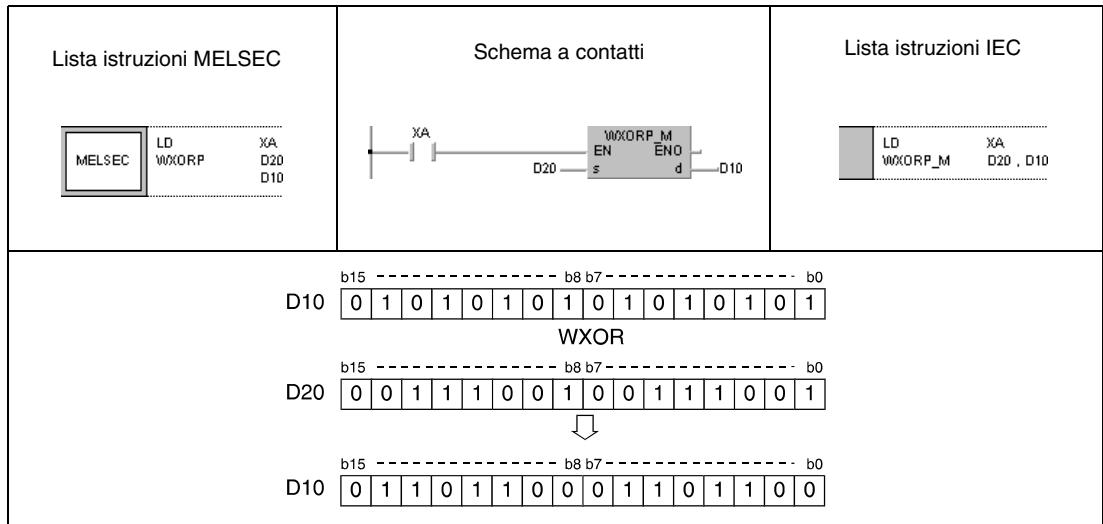
NOTA

Nella variante 1 (s, d) nessun errore di esecuzione è associato alle istruzioni WXOR, WXORP, DXOR, e DXORP, se non viene usata l'indicizzazione.

Programma di esempio 1

WXORP (s, d)

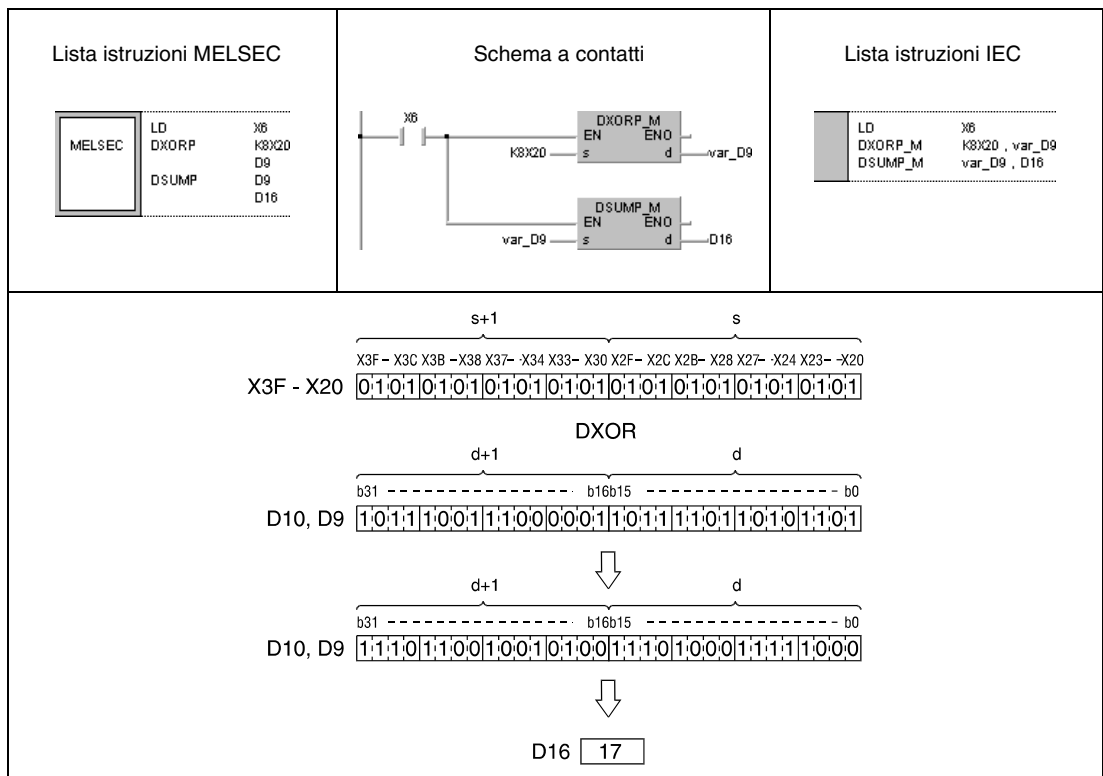
Il programma che segue collega il dato in D10 con quello in D20 sul fronte di salita di XA. Il risultato è memorizzato nuovamente in D10.



Programma di esempio 2

DXORP (s, d)

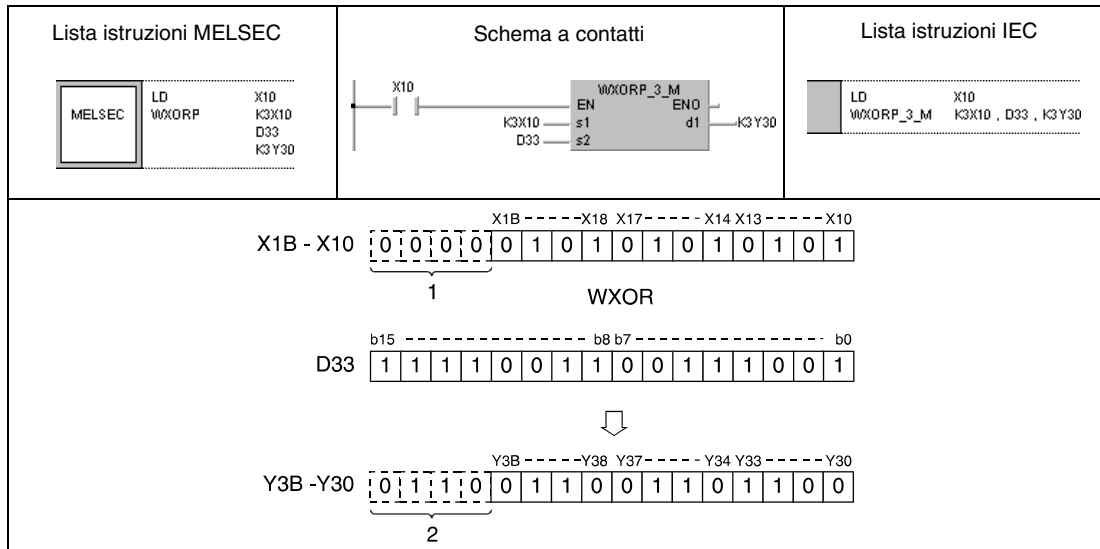
Con il fronte positivo di X6, il programma che segue confronta i dati a 32 bit sugli ingressi da X20 a X3F con la maschera di bit nei registri dati D9 e D10. Il risultato viene memorizzato nuovamente in D9 e D10. Il numero di bit a 1 in D9 e D10 viene memorizzato in D16.



Programma di esempio 3

WXORP (s1, s2, d1)

Con il fronte di salita di X10, il programma che segue forma l'OR esclusivo dei dati sugli ingressi da X10 a X1B con il dato in D33. Il risultato viene memorizzato in D33 ed inviato sulle uscite da Y30 a Y3B.

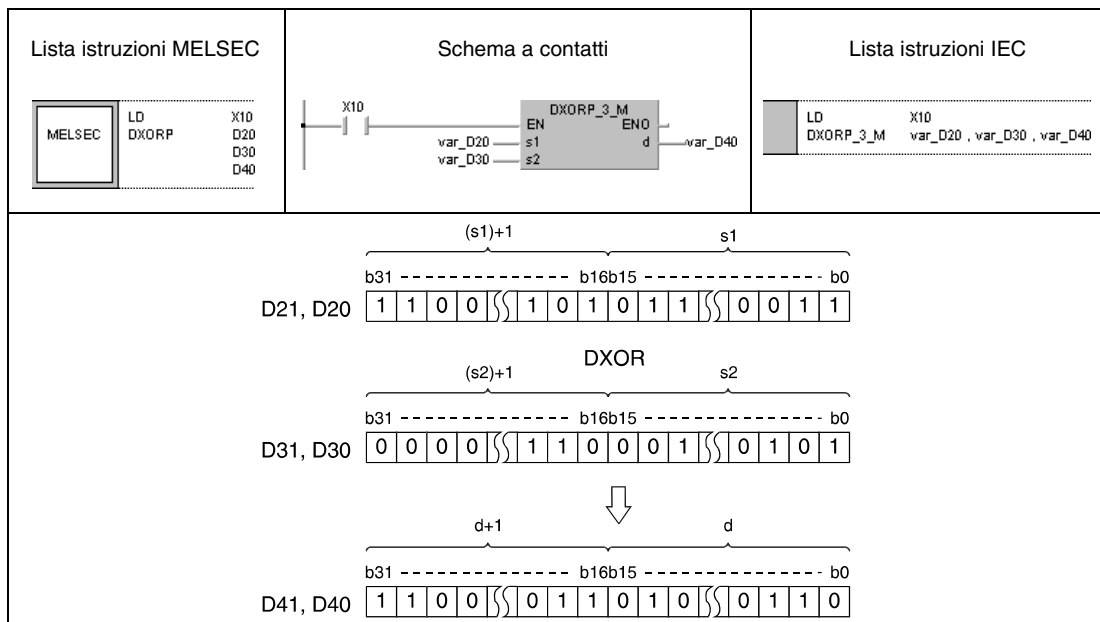


- ¹ Questi bit vengono impostati a 0
- ² Questi bit non vengono modificati

Programma di esempio 4

DXORP (s1, s2, d)

Con il fronte di salita di X10, il programma che segue forma l'OR esclusivo del dato in D20 e D21 con il dato in D30 e D31. Il risultato viene memorizzato in D40 e D41.



NOTA

I programmi di esempio 2 e 4 non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provocano messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.


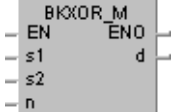
7.1.6 BKXOR, BKXORP**CPU**

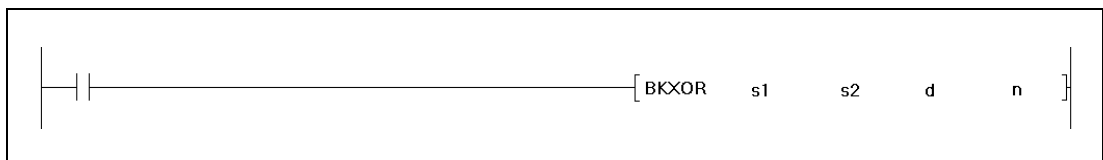
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	SM0	5	
s2	—	●	●	—	—	—	●	—			
d	—	●	●	—	—	—	—	—			
n	●	●	●	●	●	●	●	—			

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC										
 <table style="display: inline-table; border: none;"> <tr> <td style="padding: 0 10px;">BKXOR</td> <td style="padding: 0 10px;">s1</td> </tr> <tr> <td></td> <td style="padding: 0 10px;">s2</td> </tr> <tr> <td></td> <td style="padding: 0 10px;">d</td> </tr> <tr> <td></td> <td style="padding: 0 10px;">n</td> </tr> </table>	BKXOR	s1		s2		d		n		<table style="display: inline-table; border: none;"> <tr> <td style="padding: 0 10px;">BKXOR_M</td> <td style="padding: 0 10px;">s1, s2, n, d</td> </tr> </table>	BKXOR_M	s1, s2, n, d
BKXOR	s1											
	s2											
	d											
	n											
BKXOR_M	s1, s2, n, d											

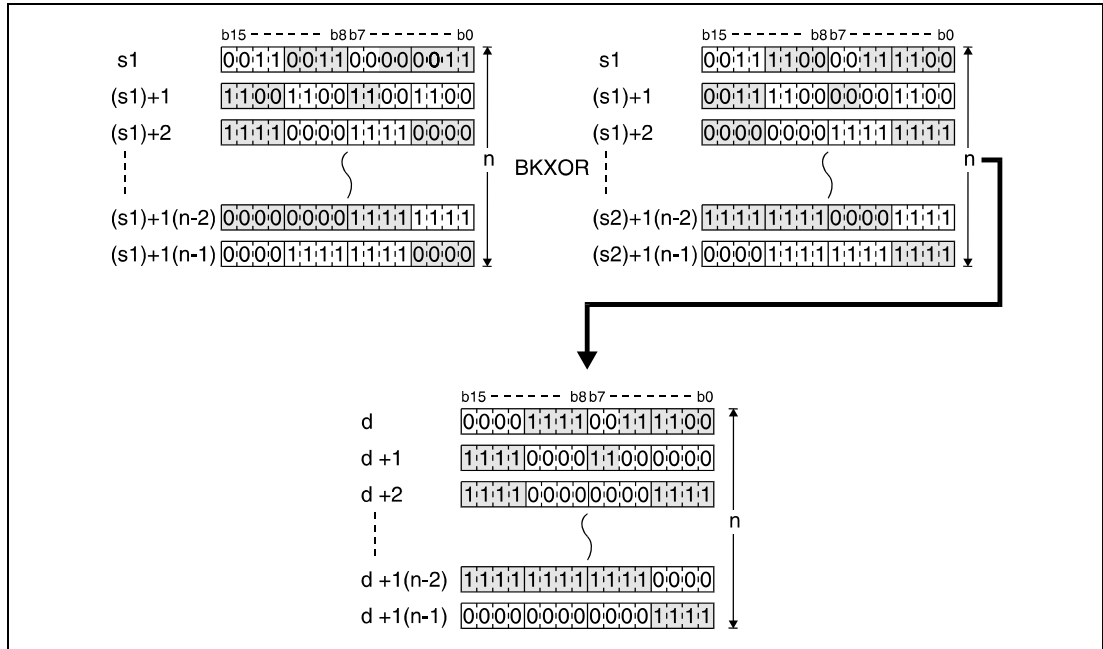
GX Developer**Variabili**

Dati impostati	Significato	Tipo dati
s1	Primo indirizzo di operando che contiene il dato per l'operazione logica.	BIN 16-bit
s2	Primo indirizzo del dato o primo indirizzo dell'operando che contiene il dato per l'operazione logica.	
d	Primo numero di operando che contiene il risultato dell'operazione	
n	Numero dei blocchi dati che formano l'OR esclusivo.	

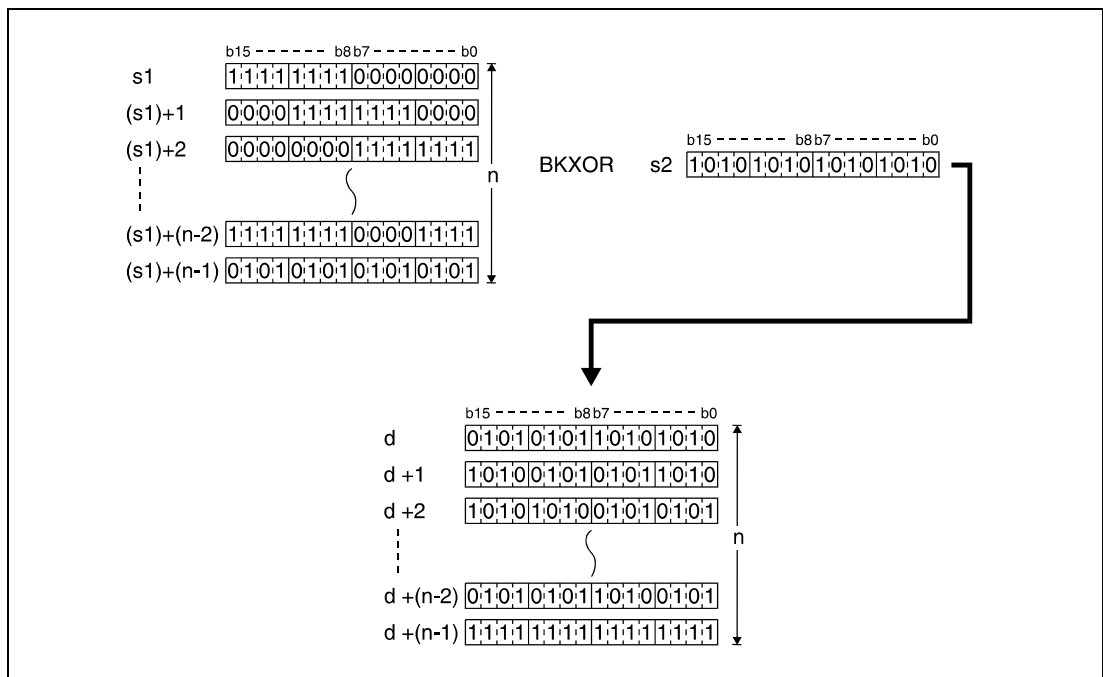
Funzioni Istruzioni di OR esclusivo con blocchi dati da 16 bit

BKXOR Operazioni OR esclusivo con blocchi dati

L'istruzione BKXOR forma l'OR esclusivo della serie di blocchi iniziando dai blocchi dati a 16 bit indicati rispettivamente da s1 e s2. I vari blocchi a 16 bit del risultato sono memorizzati dall'operando d in avanti. Il numero di dati da elaborare è indicato in n.



La costante s2 può variare nel campo fra -32768 e 32767.



Errori di esecuzione

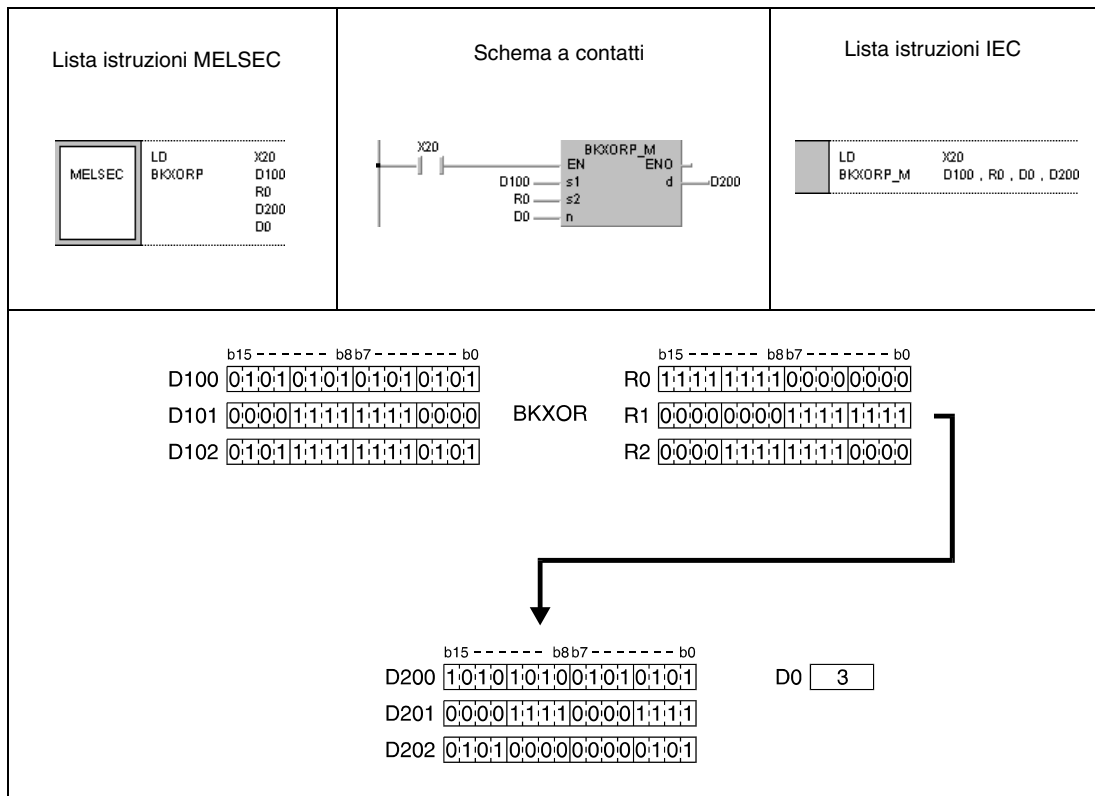
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il numero di blocchi dati indicato da n supera i numeri dispositivo indicati da s1 e s2 (codice di errore: 4101).
- I numeri degli operandi indicati da s1, s2 o d sono sovrapposti (codice di errore: 4101).

Esempio di programma

BKXORP

Con il fronte positivo di X20, il programma che segue forma l'OR esclusivo dei dati contenuti nei registri da D100 a D102 e dei dati nei registri da R0 a R2. Il risultato viene memorizzato nei registri da D200 a D202. Il numero di blocchi da elaborare (3) è contenuto in D0.



7.1.7 WXNR, WXNRP, DXNR, DXNRP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Rapporto	Errore								
Operandi a bit						Operandi a word (16-bit)						Costanti		Punta-tore	Livello													
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V						K	H (16#)	P	I	N	M9012	M9010 M9011	
WXNR																												
s	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●							K1 ↓ K4	5 ↓ 7	●	●
d		●	●	●	●	●	●	●	●	●	●	●	●	●	●													
s1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●											
s2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●											
d1		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●												
DXNR																												
s	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●							K1 ↓ K8	9 ↓ 1	●	●
d		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●											

¹ Consultare la sezione "Programmazione con CPU AnA, AnAS e AnU" del manuale di programmazione per il corrispondente numero di passi.

Operandi MELSEC Q

Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word							
WXNR, WXNRP											
s	●	●	●	●	●	●	●	●	—	—	3
d	●	●	●	●	●	●	●	—	—	—	
s1	●	●	●	●	●	●	●	●	—	—	4
s2	●	●	●	●	●	●	●	●	—	—	
d	●	●	●	●	●	●	●	—	—	—	
DXNR, DXNRP											
s	●	●	●	●	●	●	●	●	—	—	4 ¹⁾
d	●	●	●	●	●	●	●	—	—	—	
s1	●	●	●	●	●	●	●	●	—	—	4 ²⁾
s2	●	●	●	●	●	●	●	●	—	—	
d1	●	●	●	●	●	●	●	—	—	—	

¹ Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.

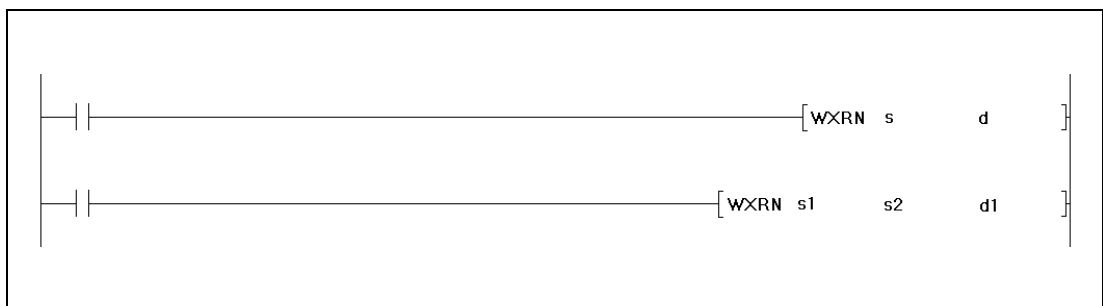
Se viene usata una CPU QnA:	4
Se viene usata una CPU a singolo processore del System Q	3
Se viene usata una CPU multiprocessore del System Q con operandi interni a word (tranne registro file ZR) o costanti:	6
Se viene usata una CPU multiprocessore del System Q con operandi a bit il cui numero operando sia multiplo di 16, con indicazione di cifra K4, e che non usa indicizzazione:	6
Se viene usata una CPU multiprocessore del System Q con operandi diversi da quelli menzionati:	4

- ² Il numero di passi di programma dipende dagli operandi e dal tipo di CPU usati.
- Se viene usata una CPU QnA: 4
 - Se viene usata una CPU multiprocessore del System Q con operandi interni a word (tranne file registri ZR) o costanti: 6
 - Se viene usata una CPU multiprocessore del System Q con operandi a bit il cui numero operando sia multiplo di 16, la cui designazione di cifra è K4, e che non usa indicizzazione: 6
 - Se viene usata una CPU del System Q con operandi diversi da quelli menzionati: 4

GX IEC Developer

<p>Lista istruzioni MELSEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;">MELSEC</td> <td style="padding: 2px;">W×NR</td> <td style="padding: 2px;">s</td> <td style="padding: 2px;">d</td> </tr> <tr> <td style="text-align: center;">MELSEC</td> <td style="padding: 2px;">W×NR</td> <td style="padding: 2px;">s1</td> <td style="padding: 2px;">s2 d</td> </tr> </table>	MELSEC	W×NR	s	d	MELSEC	W×NR	s1	s2 d	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">W×NR_3_M</td> <td style="padding: 2px;">s1, s2, d</td> </tr> </table>	W×NR_3_M	s1, s2, d
MELSEC	W×NR	s	d									
MELSEC	W×NR	s1	s2 d									
W×NR_3_M	s1, s2, d											

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Dato per l'operazione di NOR esclusivo, o primo indirizzo dell'operando che contiene questo dato.	BIN 16/32-bit
d		
s1	Dato per l'operazione di NOR esclusivo, o primo indirizzo dell'operando che contiene questo dato.	
s2		
d (d1 per WXNRP)	Primo indirizzo di operando che contiene il risultato dell'operazione logica	

Funzioni

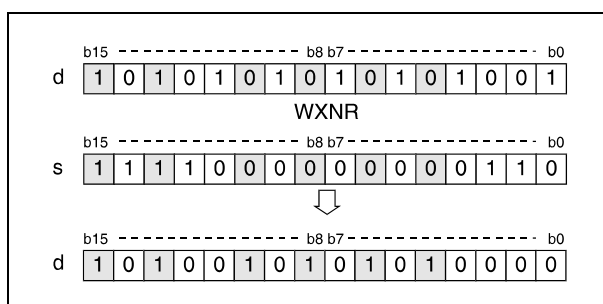
NOR esclusivo logico

WXNR Dati 16-bit

L'operazione di NOR esclusivo forma il seguente prodotto di somme logiche di due variabili d'ingresso ($Y = (\bar{A}+B)x(A+\bar{B})$).

● Variante 1:

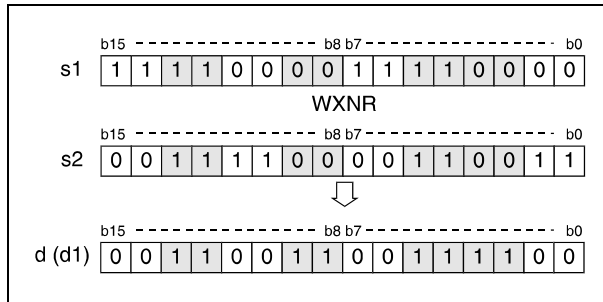
I dati a 16 bit indicati da s e da d producono il NOR esclusivo risultante dall'operazione. Il risultato viene inviato all'operando indicato da d.



● Variante 2:

I dati a 16 bit indicati da s1 e da s2 producono il NOR esclusivo risultante dall'operazione. Il risultato viene inviato all'operando indicato da d.

L'istruzione WXNRP invia il risultato all'operando indicato da d1.

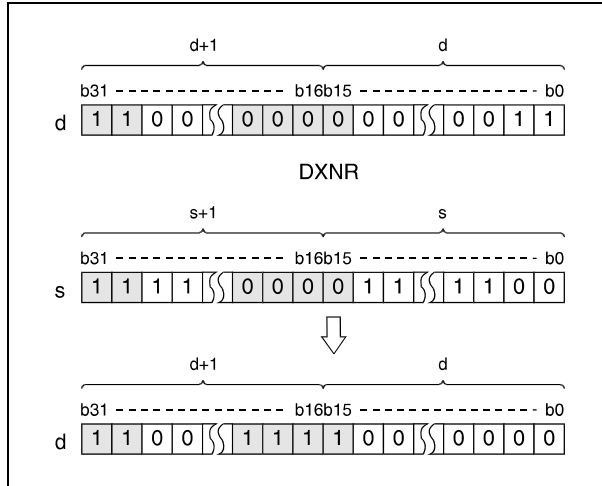


I bit che superano l'indicazione di cifra vengono impostati a 0. Ad esempio, se la cifra è indicata da K2, gli 8 bit più pesanti (da b8 a b15) sono considerati a 0.

DXNR Dati 32-bit

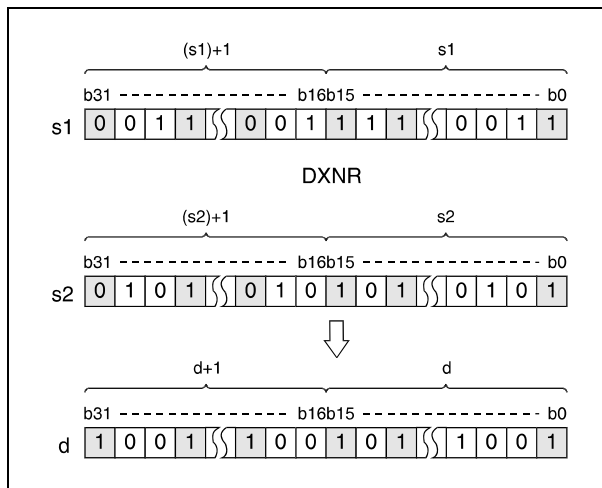
● Variante 1:

I dati a 32 bit indicati da s e da d producono il NOR esclusivo risultante dall'operazione. Il risultato viene inviato all'operando indicato da d.



● Variante 2 (serie Q e System Q):

I dati a 16 bit indicati da s1 e da s2 producono il NOR esclusivo risultante dall'operazione. Il risultato viene inviato all'operando indicato da d.

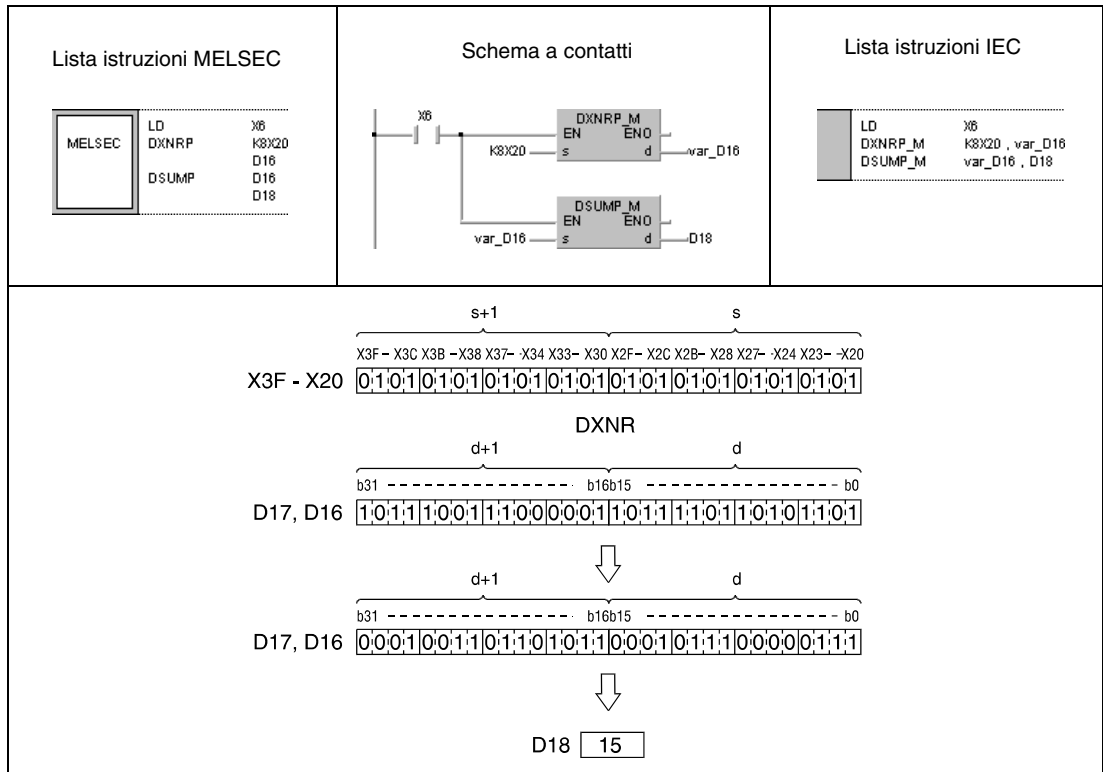


Dopo aver eseguito l'operazione, tutti i bit oltre la cifra indicata sono impostati a 0.

Programma di esempio 2

DXNRP (s, d)

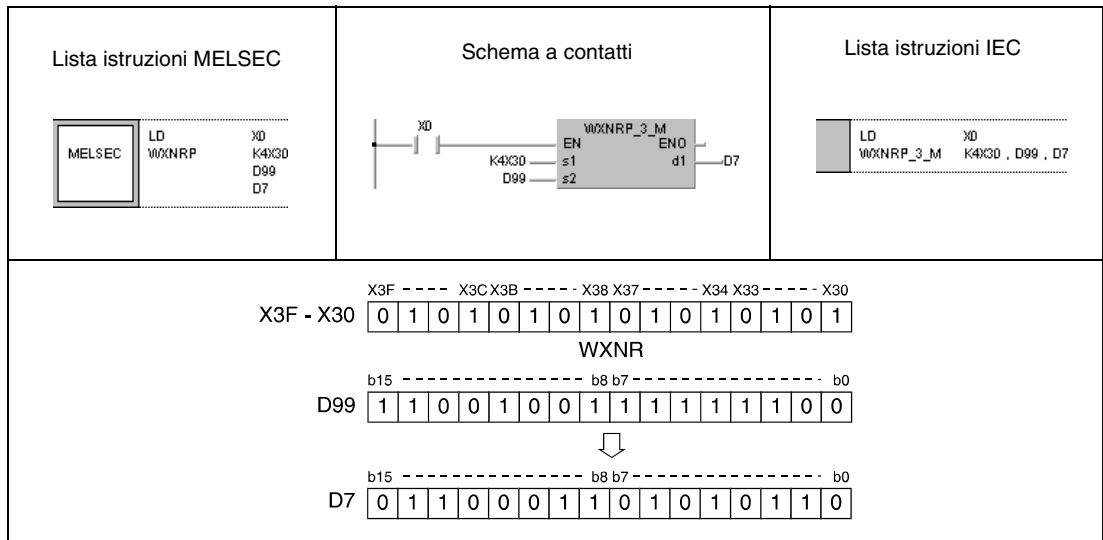
Con il fronte positivo di X6, il programma seguente confronta la maschera di bit del valore a 32 bit sugli ingressi da X20 a X3F con il valore del dato in D16 e D17. Il risultato della somma è memorizzato nuovamente in D16 e D17. Il numero di bit a 1 viene memorizzato in D18.



Programma di esempio 3

WXNRP (s1, s2, d1)

Con il fronte positivo di X0, il programma seguente esegue una operazione di NOR esclusivo fra il dato a 16 bit sugli ingressi da X30 a X3F e il dato in D99. Il risultato della somma è memorizzato in D7.



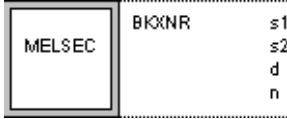
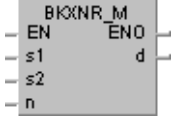
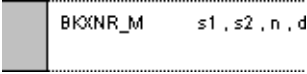
7.1.8 BKXNR, BKXNRP**CPU**

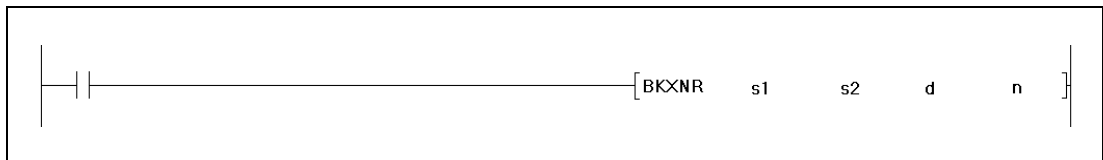
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	SM0	5	
s2	—	●	●	—	—	—	●	—			
d	—	●	●	—	—	—	—	—			
n	●	●	●	●	●	●	●	—			

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
		

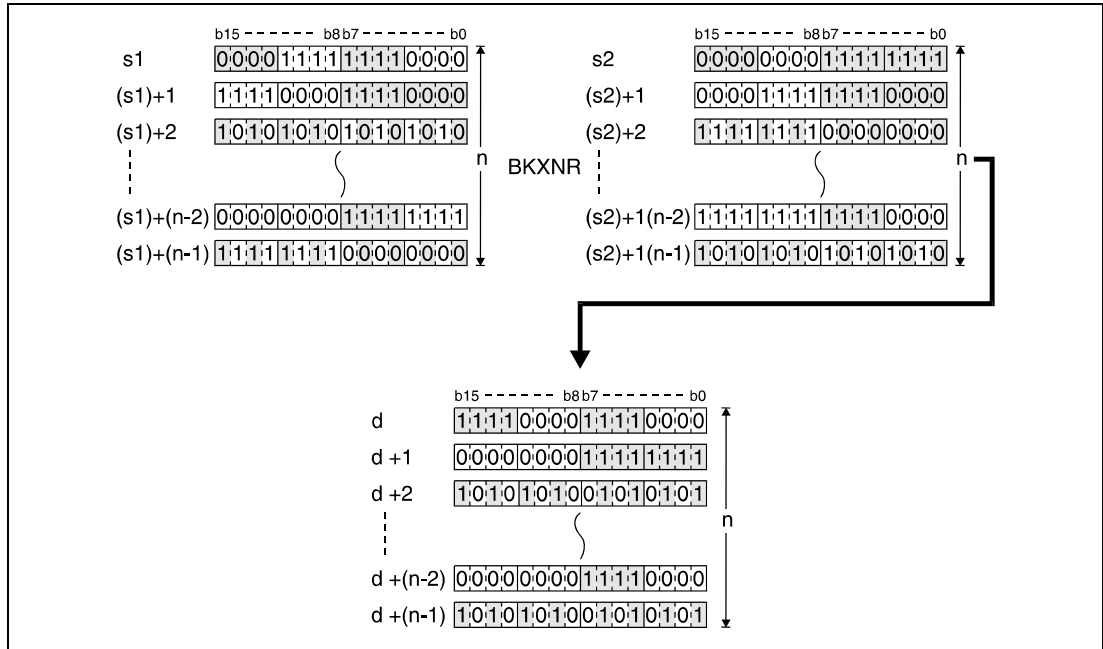
GX Developer**Variabili**

Dati impostati	Significato	Tipo dati
s1	Primo indirizzo di operando che contiene il dato per l'operazione logica.	BIN 16-bit
s2	Primo indirizzo del dato o primo indirizzo dell'operando che contiene il dato per l'operazione logica.	
d	Primo indirizzo di operando che contiene il risultato dell'operazione logica	
n	Numero di blocchi dati da elaborare.	

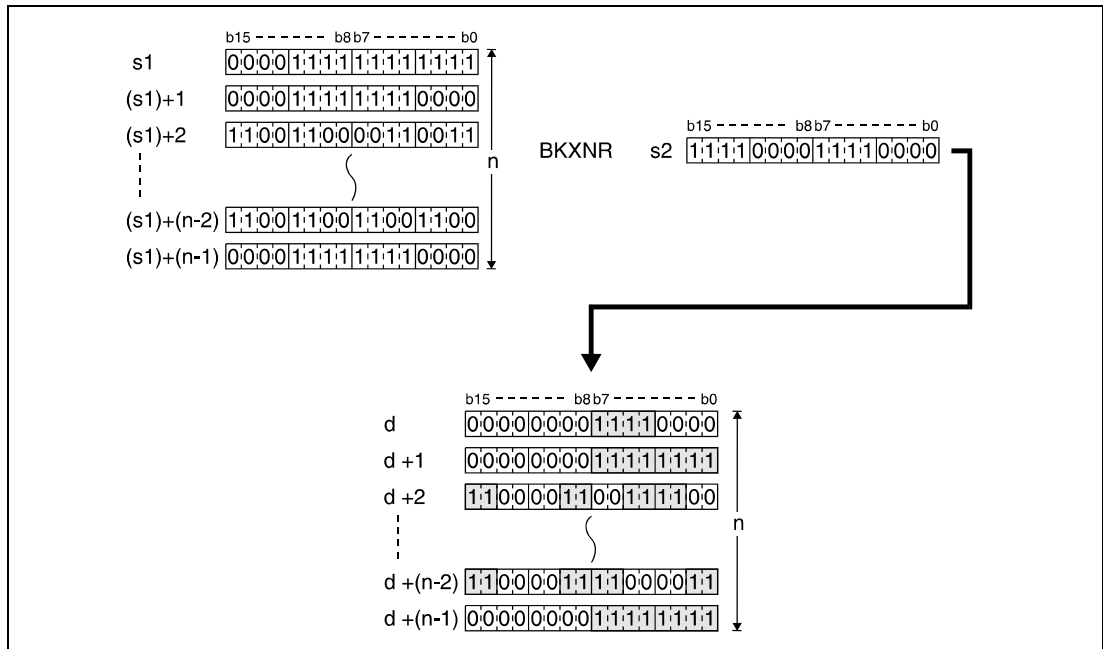
Funzioni Istruzioni di NOR esclusivo con blocchi dati da 16 bit

BKXNR Operazioni NOR esclusivo con blocchi dati

L'istruzione BKXNR forma il NOR esclusivo della serie di blocchi iniziando dai blocchi dati a 16 bit indicati rispettivamente da s1 e s2. I vari blocchi a 16 bit del risultato sono memorizzati dall'operando d in avanti. Il numero di dati da elaborare è indicato in n.



La costante s2 può variare nel campo fra -32768 e 32767.



Errori di esecuzione

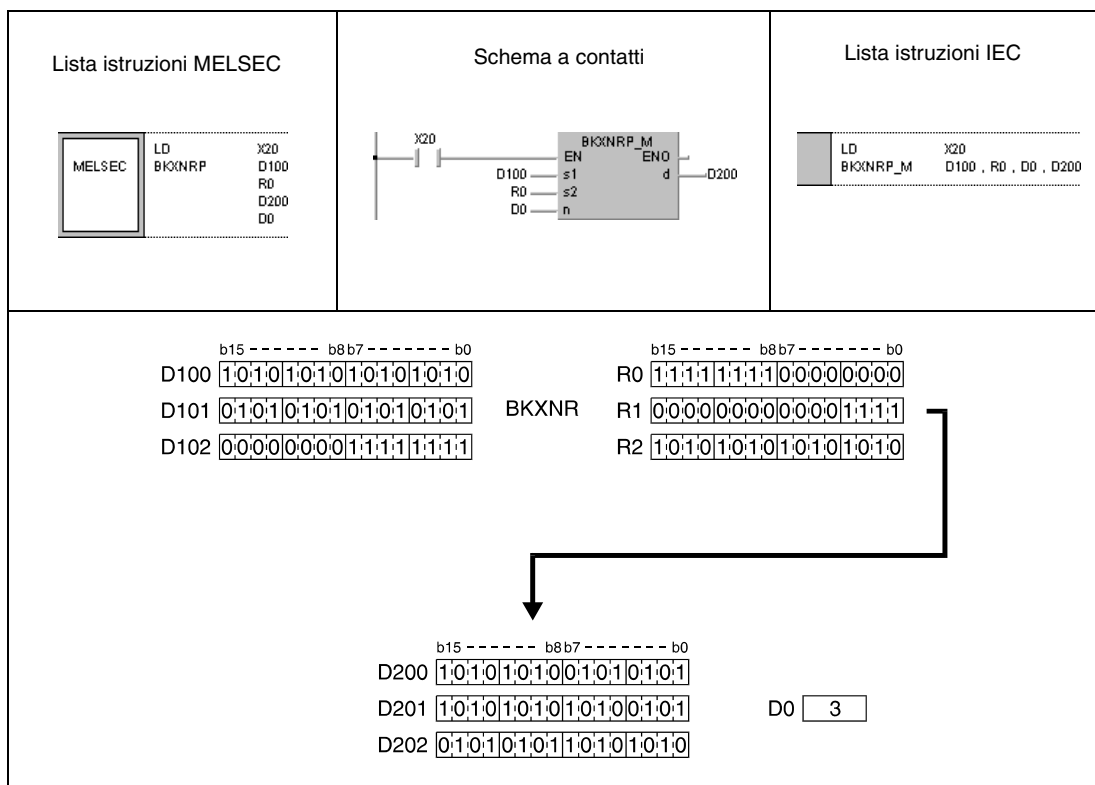
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il numero di blocchi dati indicato da n supera i numeri dispositivo indicati da s1 e s2 (codice di errore: 4101).
- I numeri degli operandi indicati da s1, s2 o d sono sovrapposti (codice di errore: 4101).

Esempio di programma

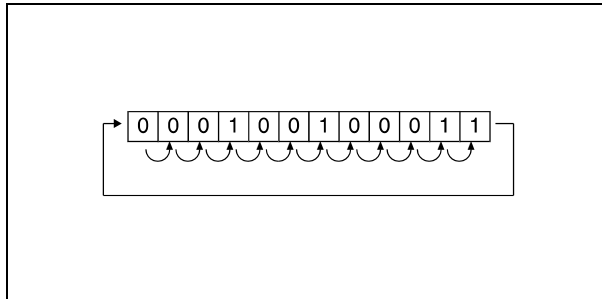
BKXNRP

Con il fronte positivo di X20, il programma che segue forma il NOR esclusivo dei dati contenuti nei registri da D100 a D102 e dei dati nei registri da R0 a R2. Il risultato dell'operazione viene memorizzato nei registri da D200 a D202. Il numero di blocchi da 16 bit da elaborare (3) è contenuto in D0.



7.2 Istruzioni per rotazione dati

Le istruzioni seguenti ruotano bit a bit i dati contenuti in accumulatori e registri. I dati possono essere ruotati verso destra e verso sinistra.



Le istruzioni di rotazione possono o meno comprendere il segnale di riporto. Le istruzioni di rotazione si applicano a dati a 16 e 32 bit. Vengono fornite in totale 16 diverse istruzioni di rotazione:

Funzione	Istruzione MELSEC in editor MELSEC	Istruzione IEC in editor IEC
Rotazione dati verso destra (16-bit)	ROR	ROR_M
	RORP	RORP_M
	RCR	RCR_M
	RCRP	RCRP_M
Rotazione dati verso sinistra (16-bit)	ROL	ROL_M
	ROLP	ROLP_M
	RCL	RCL_M
	RCLP	RCLP_M
Rotazione dati verso destra (32-bit)	DROR	DROR_M
	DRORP	DRORP_M
	DRCR	DRCR_M
	DRCRP	DRCRP_M
Rotazione dati verso sinistra (32-bit)	DROL	DROL_M
	DROLP	DROLP_M
	DRCL	DRCL_M
	DRCLP	DRCLP_M

NOTA Utilizzare istruzioni IEC con gli editor IEC

7.2.1 ROR, RORP, RCR, RCRP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A


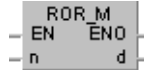

Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Riporto	Errore			
Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello								
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H (16#)	P	I	N	M9012	M9010 M9011
n																●	●				●	●

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

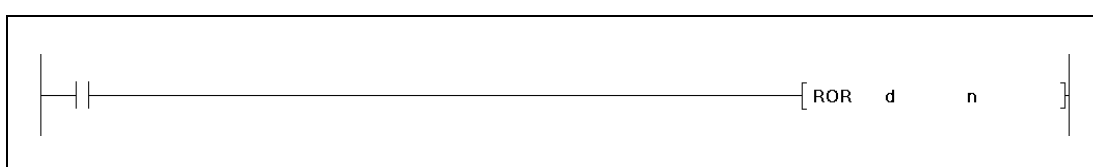
Operandi MELSEC Q

Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word							
d	●	●	●	●	●	●	—	—	—	3	
n	●	●	●	●	●	●	●	—	—		

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
 MELSEC ROR d n	 ROR_M EN ENO n d	 ROR_M n, d

GX Developer



NOTA

Le CPU della serie A ruotano sempre i dati nel registro A0. Per questa ragione, nessun operando d viene utilizzato quando si usano queste istruzioni con la serie A.

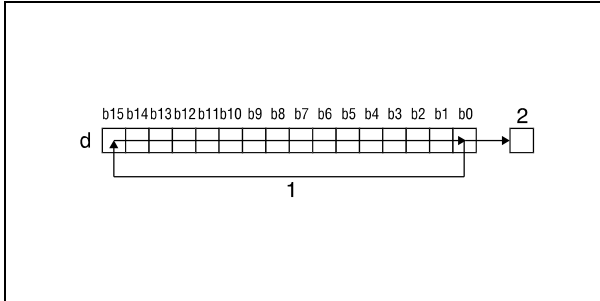
Variabili

Dati impostati	Significato	Tipo dati
d	Primo indirizzo dell'operando su cui si esegue la rotazione dei dati. serie A: solo operando A0.	BIN 16-bit
n	Numero di rotazioni (da 0 a 15).	

Funzioni Rotazione dati verso destra (16 bit)

ROR Rotazione senza segnale di riporto

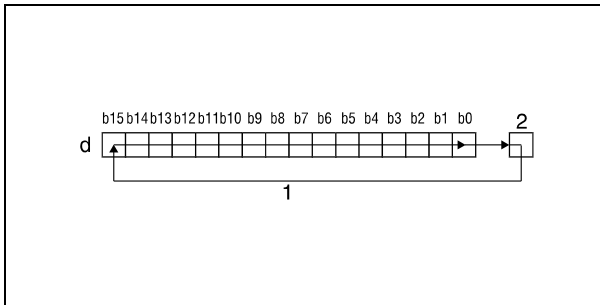
L'istruzione ROR ruota i bit dati dell'operando indicato da d (A0) di n bit verso destra. Il segnale di riporto (serie A = M9012, serie Q e System Q =SM700) non viene incluso. Il segnale di riporto assume lo stato dell'ultimo bit ruotato da b0 in b15.



- ¹ Rotazione di n bit
- ² Segnale di riporto

RCR Rotazione con segnale di riporto

L'istruzione RCR ruota i bit dati dell'operando indicato da d (A0) di n bit verso destra, includendo il segnale di riporto. Il segnale di riporto (serie A = M9012, serie Q e System Q = SM700) assume la condizione del bit ruotato di n bit. Lo stato del segnale di riporto (0 o 1) prima della rotazione viene spostato verso destra in d (A0) di n bit, iniziando da B15.



- ¹ Rotazione di n bit
- ² Segnale di riporto

NOTA

Solo serie Q e System Q:

Se d indica un operando a bit, l'operazione di rotazione viene eseguita su un operando che contiene il numero di bit specificato. Il numero di bit usati per la rotazione viene determinato dal resto del quoziente seguente:

Numero di rotazioni n / numero di bit

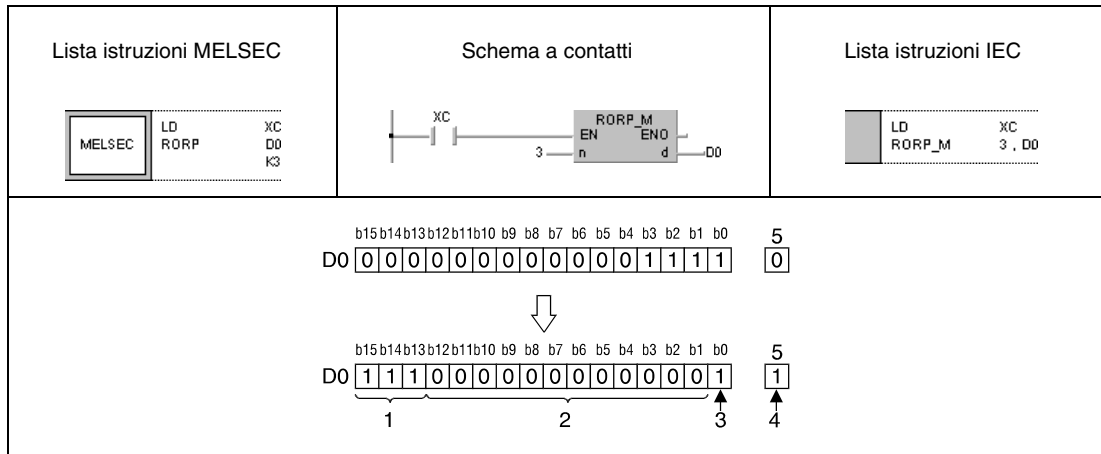
Ad esempio 16 rotazioni di 12 bit corrispondono ad una rotazione di 4 bit, dato che il resto del quoziente 16/12 è 4. Questo è dovuto al fatto che in una rotazione su 12 bit, dopo la dodicesima rotazione, si raggiunge nuovamente la stessa posizione iniziale.

Per questa ragione, indicare con n un numero nel campo da 0 a 15.

Programma di esempio 1

RORP (serie Q e System Q)

Con il fronte positivo di XC, il programma che segue ruota il contenuto di D0 di 3 bit verso destra.

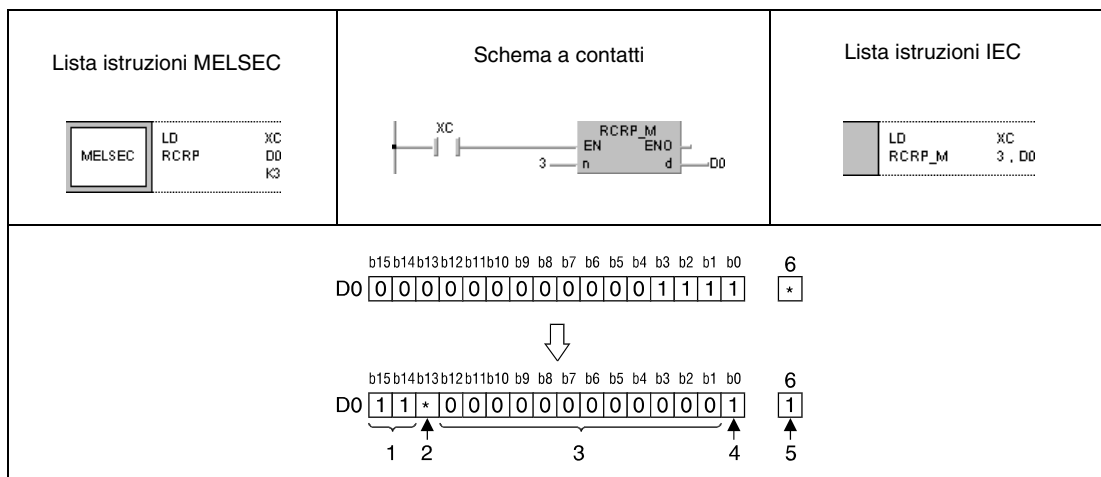


- ¹ Contenuto dei bit b0-b2 prima della rotazione
- ² Contenuto dei bit b4-b15 prima della rotazione
- ³ Contenuto del bit b3 prima della rotazione
- ⁴ Contenuto del bit b2 prima della rotazione
- ⁵ Segnale di riporto

Programma di esempio 2

RCRP (serie Q e System Q)

Con il fronte positivo di XC, il programma che segue ruota il contenuto di D0 di 3 bit verso destra, includendo il segnale di riporto SM700. Lo stato di SM700 (0/1) prima della rotazione viene spostato di 3 bit sulla destra.



- ¹ Contenuto dei bit b1 e b0 prima della rotazione
- ² Contenuto del segnale di riporto prima della rotazione
- ³ Contenuto dei bit b4-b15 prima della rotazione
- ⁴ Contenuto del bit b3 prima della rotazione
- ⁵ Contenuto del bit b2 prima della rotazione
- ⁶ Segnale di riporto

7.2.2 ROL, ROLP, RCL, RCLP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

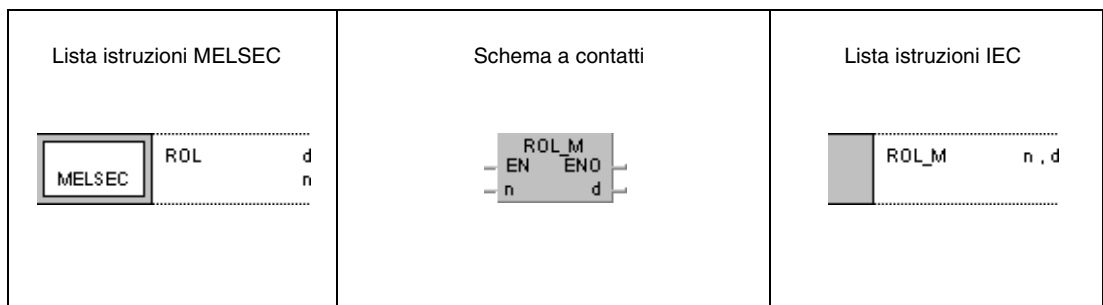
Operandi utilizzabili														Definizione cifra	Numero di passi	Indice	Riporto	Errore									
Operandi a bit						Operandi a word (16-bit)						Costanti							Puntatore		Livello						
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1						Z	V	K	H (16#)	P	I	N		
n																●	●					3	1	●	●	●	●

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

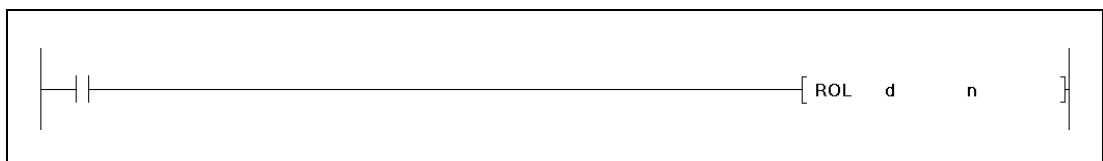
Operandi MELSEC Q

Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word							
d	●	●	●	●	●	●	—	—	—	3	
n	●	●	●	●	●	●	●	—	—		

GX IEC Developer



GX Developer



NOTA

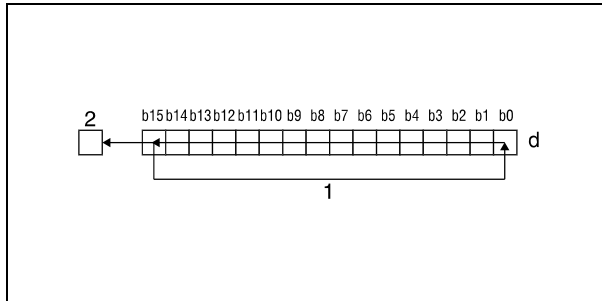
Le CPU della serie A ruotano sempre i dati nel registro A0. Per questa ragione, nessun operando d viene utilizzato quando si usano queste istruzioni con la serie A.

Variabili

Dati impostati	Significato	Tipo dati
d	Primo indirizzo dell'operando su cui si esegue la rotazione dei dati. serie A: solo operando A0.	BIN 16-bit
n	Numero di rotazioni (da 0 a 15).	

Funzioni **Rotazione dati verso sinistra (16 bit)****ROL** **Rotazione senza segnale di riporto**

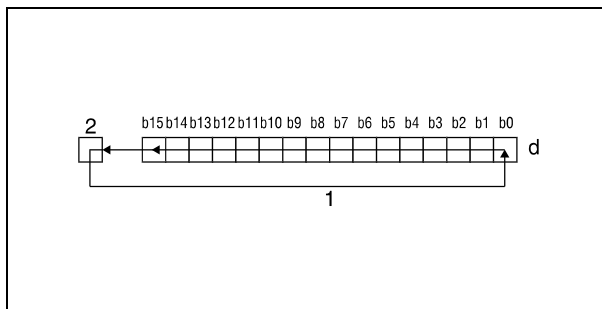
L'istruzione ROL ruota i bit dati dell'operando indicato da d (A0) di n bit verso sinistra. Il segnale di riporto (serie A = M9012, serie Q e System Q = SM700) non viene incluso. Il segnale di riporto assume lo stato dell'ultimo bit ruotato da b15 in b0.



- ¹ Rotazione di n bit
² Segnale di riporto

RCL **Rotazione con segnale di riporto**

L'istruzione RCL ruota i bit dati dell'operando indicato da d (A0) di n bit verso sinistra, includendo il segnale di riporto. Il segnale di riporto (serie A = M9012, serie Q e System Q = SM700) assume la condizione del bit ruotato di n bit. Lo stato del segnale di riporto (0 o 1) prima della rotazione viene spostato verso sinistra in d (A0) di n bit, iniziando da b15.



- ¹ Rotazione di n bit
² Segnale di riporto

NOTA

Solo serie Q e System Q:

Se d indica un operando a bit, l'operazione di rotazione viene eseguita su un operando che contiene il numero di bit specificato. Il numero di bit usati per la rotazione viene determinato dal resto del quoziente seguente:

Numero di rotazioni n / numero di bit

Ad esempio 16 rotazioni di 12 bit corrispondono ad una rotazione di 4 bit, dato che il resto del quoziente 16/12 è 4. Questo è dovuto al fatto che in una rotazione su 12 bit, dopo la dodicesima rotazione, si raggiunge nuovamente la stessa posizione iniziale.

Per questa ragione, indicare con n un numero nel campo da 0 a 15.

7.2.3 DROR, DRORP, DRCR, DRCRP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A


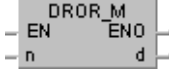

Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Riporto	Errore							
Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello												
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1							Z	V	K	H (16#)	P	I	N
n																●	●				3	1	●	●	●	●

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

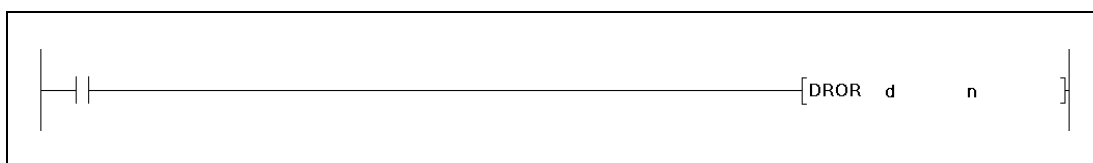
Operandi MELSEC Q

Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word							
d	●	●	●	●	●	●	—	—	—	3	
n	●	●	●	●	●	●	●	—	—		

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
 DROR d n	 DROR_M EN ENO - n d -	 DROR_M n, d

GX Developer



NOTA

Le CPU della serie A ruotano sempre i dati nei registri A0 e A1. Per questa ragione, nessun operando d viene utilizzato quando si usano queste istruzioni con la serie A.

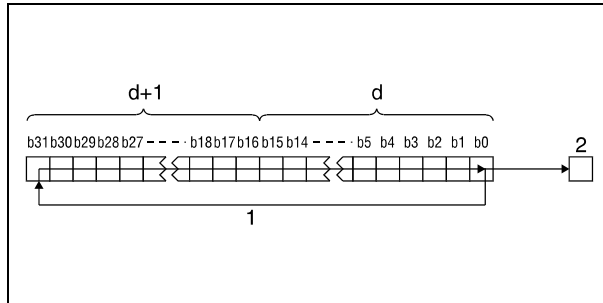
Variabili

Dati impostati	Significato	Tipo dati
d	Primo indirizzo dell'operando su cui si esegue la rotazione dei dati. serie A: solo operandi A0 e A1.	BIN 32-bit
n	Numero di rotazioni (da 0 a 31).	BIN 16-bit

Funzioni Rotazione dati verso destra (32 bit)

DROR Rotazione senza segnale di riporto

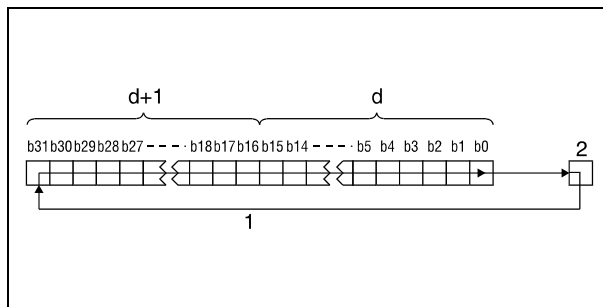
L'istruzione DROR ruota i bit dati dell'operando indicato da d (A0, A1) di n bit verso destra. Il segnale di riporto (serie A = M9012, serie Q e System Q =SM700) non viene incluso. Il segnale di riporto assume lo stato dell'ultimo bit ruotato da b0 in b31.



- ¹ Rotazione di n bit
- ² Segnale di riporto

DRCR Rotazione con segnale di riporto

L'istruzione DRCR ruota i bit dati dell'operando indicato da d (A0, A1) di n bit verso destra, includendo il segnale di riporto. Il segnale di riporto (serie A = M9012, serie Q e System Q = SM700) assume la condizione del bit ruotato di n bit. Lo stato del segnale di riporto (0 o 1) prima della rotazione viene spostato verso destra in d (A0, A1) di n bit, iniziando da b31.



- ¹ Rotazione di n bit
- ² Segnale di riporto

NOTA

Solo serie Q e System Q:

Se d indica un operando a bit, l'operazione di rotazione viene eseguita su un operando che contiene il numero di bit specificato. Il numero di bit usati per la rotazione viene determinato dal resto del quoziente seguente:

Numero di rotazioni n / numero di bit

Ad esempio 31 rotazioni di 24 bit corrispondono ad una rotazione di 7 bit, dato che il resto del quoziente 31/24 è 7. Questo è dovuto al fatto che in una rotazione su 24 bit, dopo la ventiquattresima rotazione, si raggiunge nuovamente la stessa posizione iniziale.

Per questa ragione, indicare con n un numero nel campo da 0 a 31.

7.2.4 DROL, DROLP, DRCL, DRCLP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

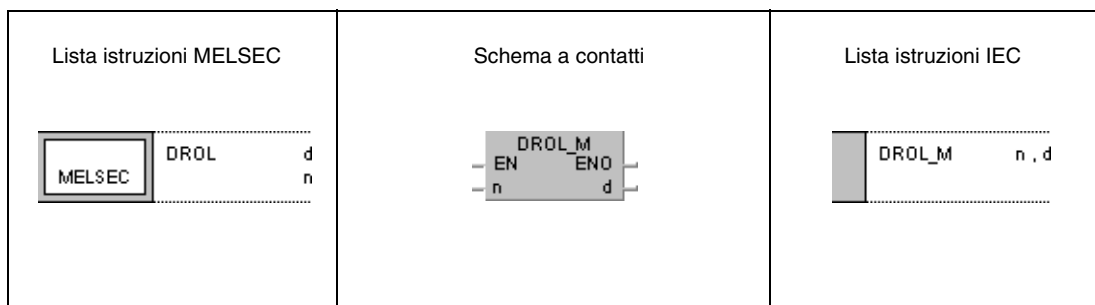
Operandi utilizzabili																Definizione cifra Numero di passi	Indice	Riporto M9012	Errore M9010 M9011					
Operandi a bit						Operandi a word (16-bit)						Costanti		Punta- tore	Livello									
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V					K	H (16#)	P	I	N
n																	●	●				3 1	●	●

¹ Consultare la sezione "Programmazione con CPU AnA, AnAS e AnU" del manuale di programmazione per il corrispondente numero di passi.

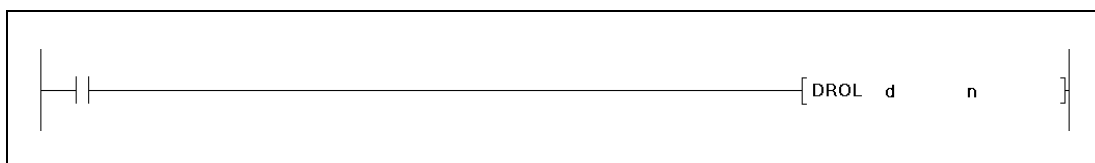
Operandi MELSEC Q

Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti E	Altro			
Bit	Word		Bit	Word							
d	●	●	●	●	●	●	—	—	—	3	
n	●	●	●	●	●	●	●	—	—		

GX IEC Developer



GX Developer



NOTA

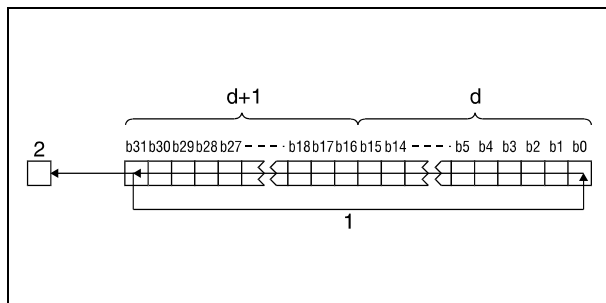
Le CPU della serie A ruotano sempre i dati nei registri A0 e A1. Per questa ragione, nessun operando d viene utilizzato quando si usano queste istruzioni con la serie A.

Variabili

Dati impostati	Significato	Tipo dati
d	Primo indirizzo dell'operando su cui si esegue la rotazione dei dati. serie A: solo operandi A0 e A1.	BIN 32-bit
n	Numero di rotazioni (da 0 a 31).	BIN 16-bit

Funzioni Rotazione dati verso sinistra (32 bit)**DROL Rotazione senza segnale di riporto**

L'istruzione DROL ruota i bit dati dell'operando indicato da d (A0, A1) di n bit verso sinistra. Il segnale di riporto (serie A = M9012, serie Q e System Q = SM700) non viene incluso. Il segnale di riporto assume lo stato dell'ultimo bit ruotato da b31 in b0.

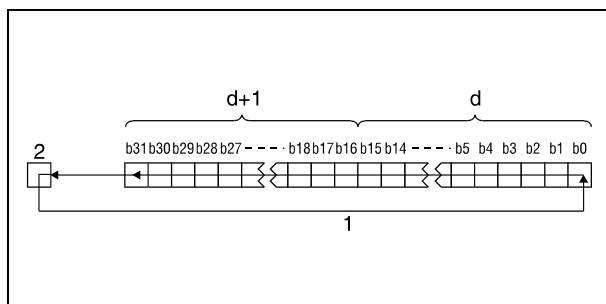


¹ Rotazione di n bit

² Segnale di riporto

DRCL Rotazione con segnale di riporto

L'istruzione DRCL ruota i bit dati dell'operando indicato da d (A0, A1) di n bit verso sinistra, includendo il segnale di riporto. Il segnale di riporto (serie A = M9012, serie Q e System Q = SM700) assume la condizione del bit ruotato di n bit. Lo stato del segnale di riporto (0 o 1) prima della rotazione viene spostato verso sinistra in d (A0, A1) di n bit, iniziando da b31.



¹ Rotazione di n bit

² Segnale di riporto

NOTA

Solo serie Q e System Q:

Se d indica un operando a bit, l'operazione di rotazione viene eseguita su un operando che contiene il numero di bit specificato. Il numero di bit usati per la rotazione viene determinato dal resto del quoziente seguente:

Numero di rotazioni n / numero di bit

Ad esempio 31 rotazioni di 24 bit corrispondono ad una rotazione di 7 bit, dato che il resto del quoziente 31/24 è 7. Questo è dovuto al fatto che in una rotazione su 24 bit, dopo la ventiquattresima rotazione, si raggiunge nuovamente la stessa posizione iniziale.

Per questa ragione, indicare con n un numero nel campo da 0 a 31.

7.3 Istruzioni di scorrimento

Le istruzioni di scorrimento spostano dati di bit o blocchi dati all'interno di una word. I dati possono essere fatti scorrere verso destra e verso sinistra.

Sono disponibili in totale 12 istruzioni di scorrimento:

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Scorrimento di una word dati da 16 bit per n bit	SFR	SFR_M
	SFRP	SFRP_M
	SFL	SFL_M
	SFLP	SFLP_M
Scorre n operandi a bit di 1 bit	BSFR	BSFR_M
	BSFRP	BSFRP_M
	BSFL	BSFL_M
	BSFLP	BSFLP_M
Scorre n operandi a word di un bit	DSFR	DSFR_M
	DSFRP	DSFRP_M
	DSFL	DSFL_M
	DSFLP	DSFLP_M

NOTA *Utilizzare istruzioni IEC con gli editor IEC*

7.3.1 SFR, SFRP, SFL, SFLP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

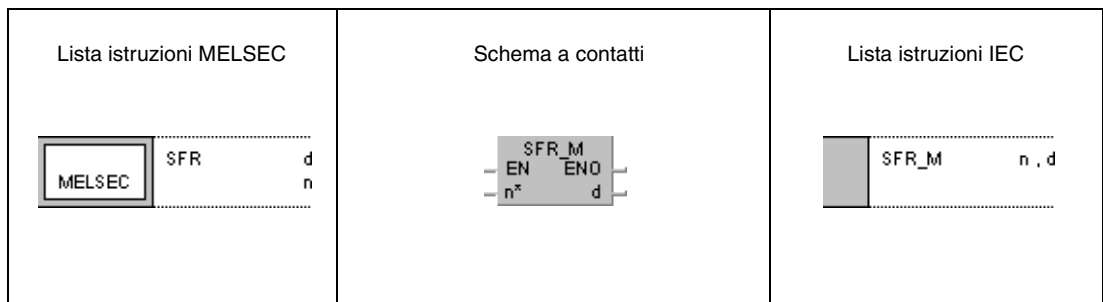
	Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Riporto	Errore							
	Operandi a bit					Operandi a word (16-bit)					Costanti		Punta-tore		Livello												
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z						V	K	H (16#)	P	I	N	
d	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●						K1 ↓ K4	3 ↓ 1	●	●	●	●	
n																●	●										

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

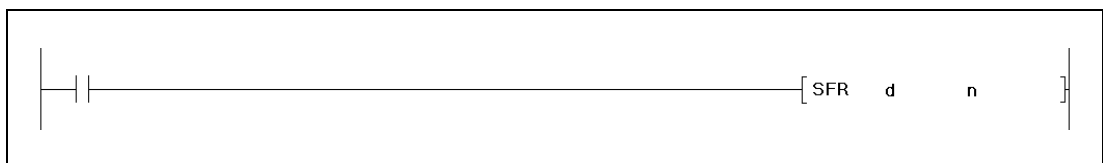
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
d	●	●	●	●	●	●	●	—	—	—	3
n	●	●	●	●	●	●	●	●	—	—	

GX IEC Developer



GX Developer

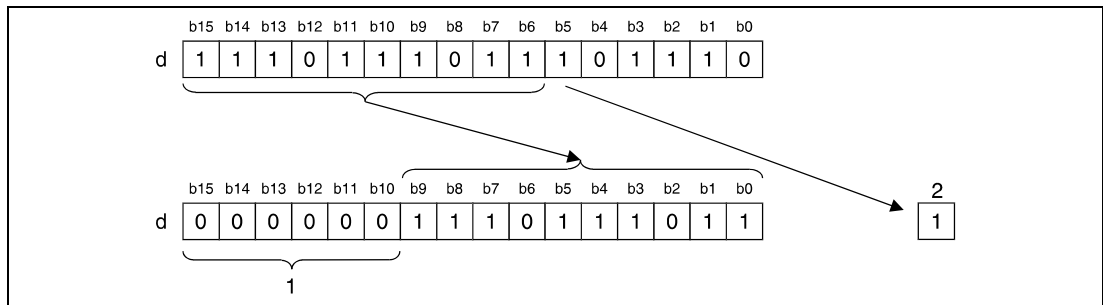


Variabili

Dati impostati	Significato	Tipo dati
d	Primo indirizzo dell'operando che contiene i dati da scorrere.	BIN 16-bit
n	Numero di scorrimenti (da 0 a 15).	

Funzioni **Scorrimento di una word dati da 16 bit per n bit****SFR** **Scorrimento sulla destra**

L'istruzione SFR scorre la word dati da 16 bit indicata da d di n bit verso destra.



¹ Questi bit vengono impostati a 0

² Segnale di riporto

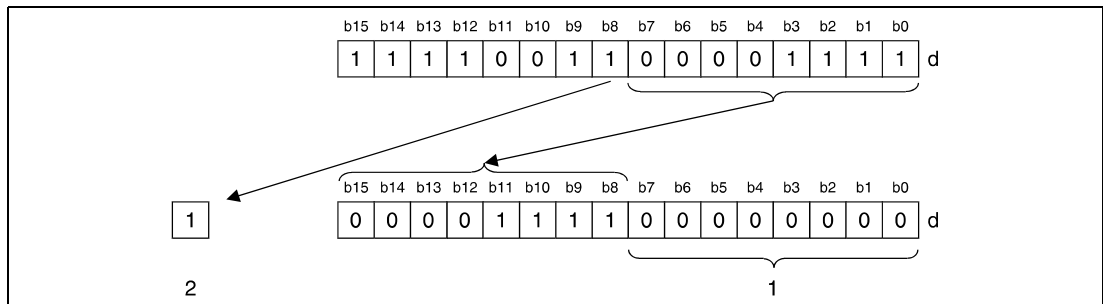
Gli n bit più significativi iniziando dal bit b15 vengono impostati a 0. L'ennesimo bit (b(n-1)) da far scorrere viene spostato nel segnale di riporto (serie A = M9012, serie Q e System Q = SM700).

Su timer e contatori solo il valore attuale (conteggio) può essere fatto scorrere. Il valore impostato non può essere fatto scorrere.

Sugli operandi a bit è possibile scorrere di un numero specificato di bit all'interno di un operando (vedi esempio di programma 1).

SFL **Scorrimento sulla sinistra**

L'istruzione SFL scorre la word dati da 16 bit indicata da d di n bit verso sinistra.



¹ Questi bit vengono impostati a 0

² Segnale di riporto

Gli n bit meno significativi iniziando dal bit b0 vengono impostati a 0. L'ennesimo bit (b(15-n)) da far scorrere viene spostato nel segnale di riporto (serie A = M9012, serie Q e System Q = SM700).

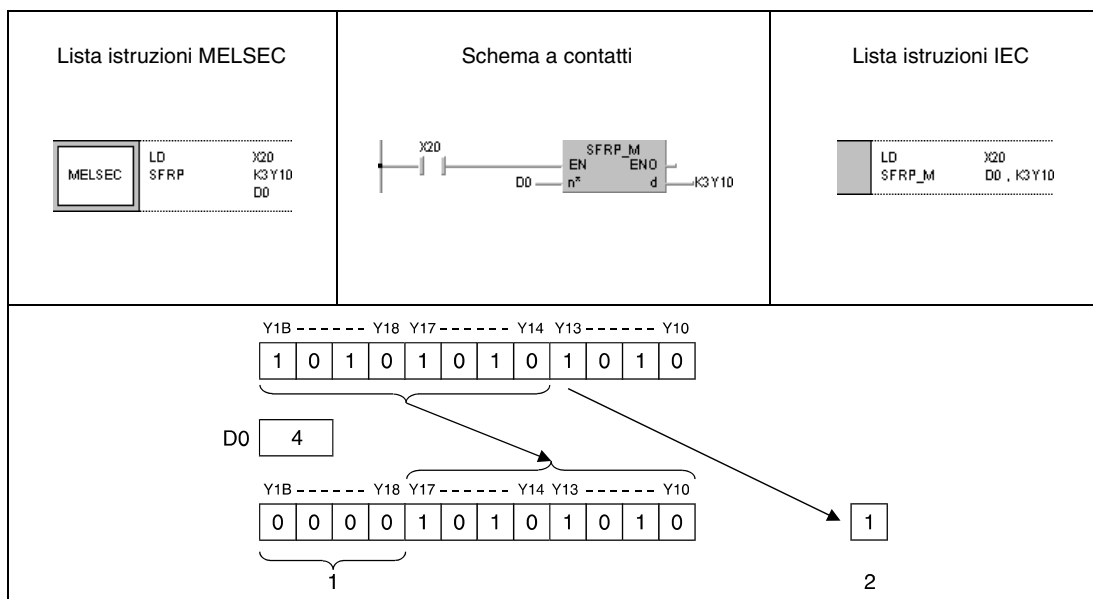
Su timer e contatori solo il valore attuale (conteggio) può essere fatto scorrere. Il valore impostato non può essere fatto scorrere.

Sugli operandi a bit è possibile scorrere di un numero specificato di bit all'interno di un operando (vedi esempio di programma 1).

Programma di esempio 1

SFRP

Con il fronte di salita di X20, il programma seguente scorre verso destra il contenuto da Y10 a Y1B del numero di bit specificato da D0. Lo stato del bit Y13 viene memorizzato nel segnale di riporto (serie A = M9012, serie Q e System Q = SM700).

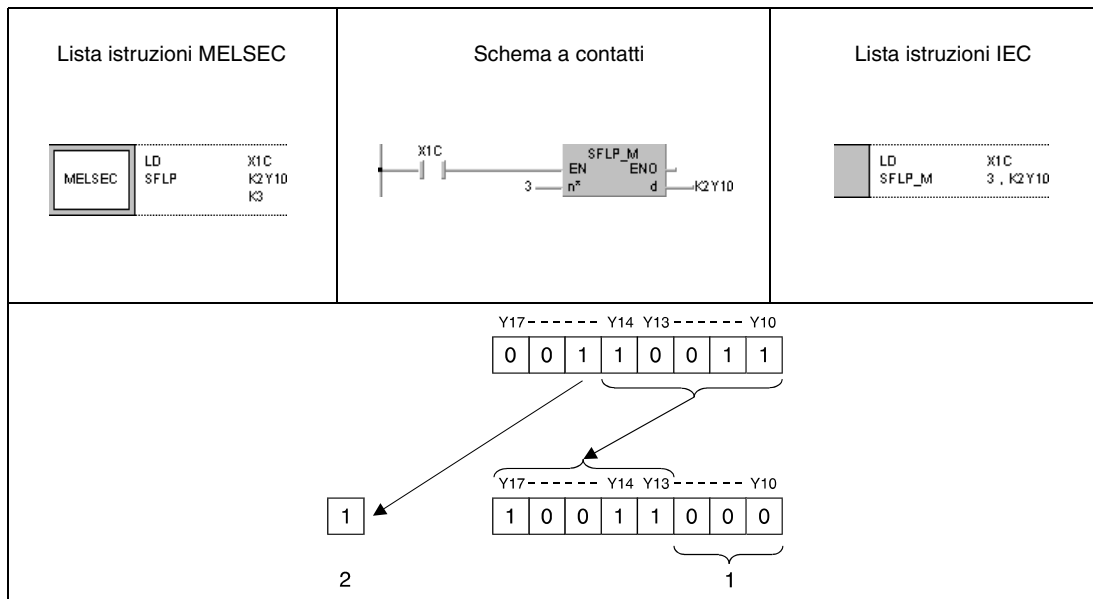


- ¹ Questi bit vengono impostati a 0
- ² Segnale di riporto

Programma di esempio 2

SFLP

Con il fronte positivo di XC, il programma che segue fa scorrere il contenuto da Y10 a Y18 di 3 bit verso sinistra. Lo stato del bit Y15 viene memorizzato nel segnale di riporto (serie A = M9012, serie Q e System Q = SM700).



- ¹ Questi bit vengono impostati a 0
- ² Segnale di riporto

7.3.2 BSFR, BSFRP, BSFL, BSFLP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A


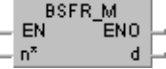

	Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Ripporto	Errore							
	Operandi a bit						Operandi a word (16-bit)						Costanti		Puntatore							Livello						
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V				K	H (16#)	P	I	N	M9012	M9010 M9011		
d	●	●	●	●	●	●																						
n																●	●								7 ₁	●	●	●

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

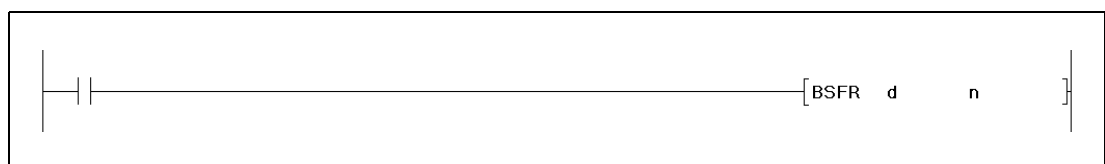
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti E			Altro U
	Bit	Word		Bit	Word						
d	●	—	—	—	—	—	—	—	—	SM0	3
n	—	●	●	●	●	●	●	●	—		

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
 BSFR d n		 BSFR_M n, d

GX Developer



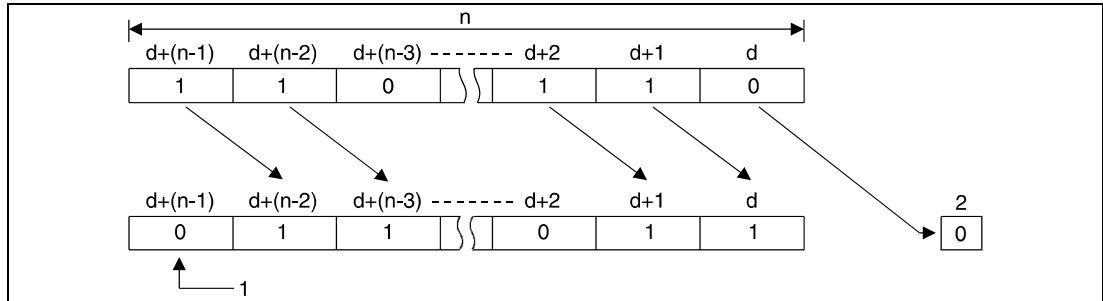
Variabili

Dati impostati	Significato	Tipo dati
d	Primo indirizzo dell'operando che deve scorrere.	Bit
n	Numero di operandi che devono scorrere.	BIN 16-bit

Funzioni **Scorrimento di 1 bit di n operandi a bit**

BSFR Scorrimento sulla destra

L'istruzione BSFR scorre il contenuto degli operandi a bit specificati di 1 bit verso destra. L'operazione di scorrimento inizia dall'indirizzo dell'operando indicato da d e prosegue sugli n indirizzi successivi.

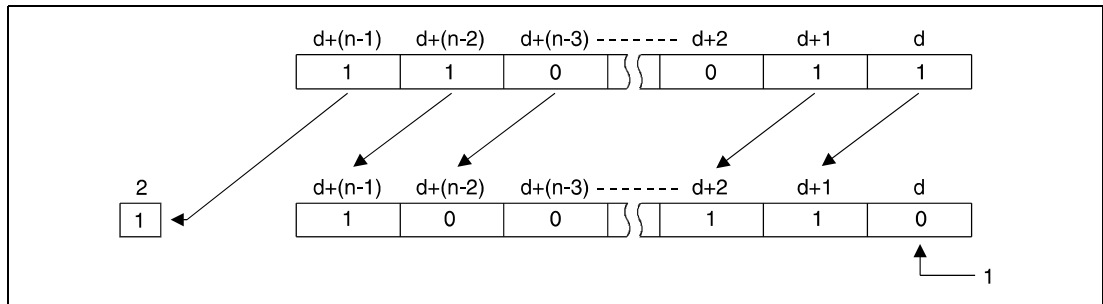


¹ Questo bit viene impostato a 0.

² Segnale di riporto

BSFL Scorrimento sulla sinistra

L'istruzione BSFL scorre il contenuto degli operandi a bit specificati di 1 bit verso sinistra. L'operazione di scorrimento inizia dall'indirizzo dell'operando indicato da d e prosegue sugli n indirizzi successivi.



¹ Questo bit viene impostato a 0.

² Segnale di riporto

Errori di esecuzione

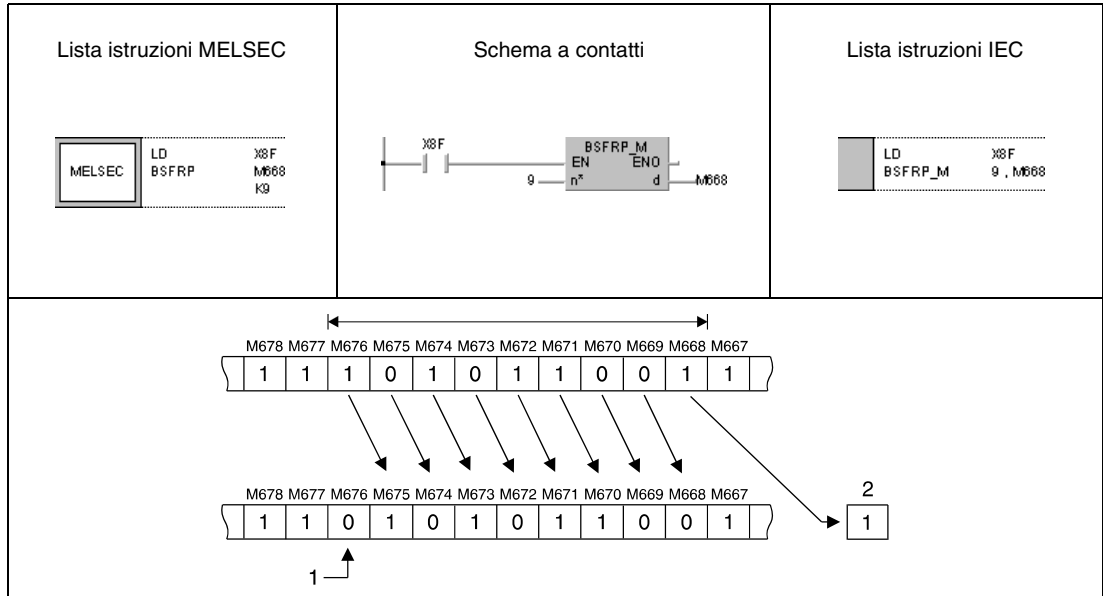
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il valore di n è negativo.
- Il valore di n supera il numero di bit disponibili per l'operando indicato da d. (serie Q e System Q = codice di errore 4101).

Programma di esempio 1

BSFRP

Con il fronte positivo di X8F, il programma che segue scorre i dati dei relé interni da M668 a M676 di un bit verso destra. M668 prende lo stato di M669, M669 quello di M670, ecc. Lo stato del primo operando (M668) viene inserito nel segnale di riporto (serie A = M9012, serie Q e System Q = SM700), mentre l'ultimo operando (M676) assume il valore 0.

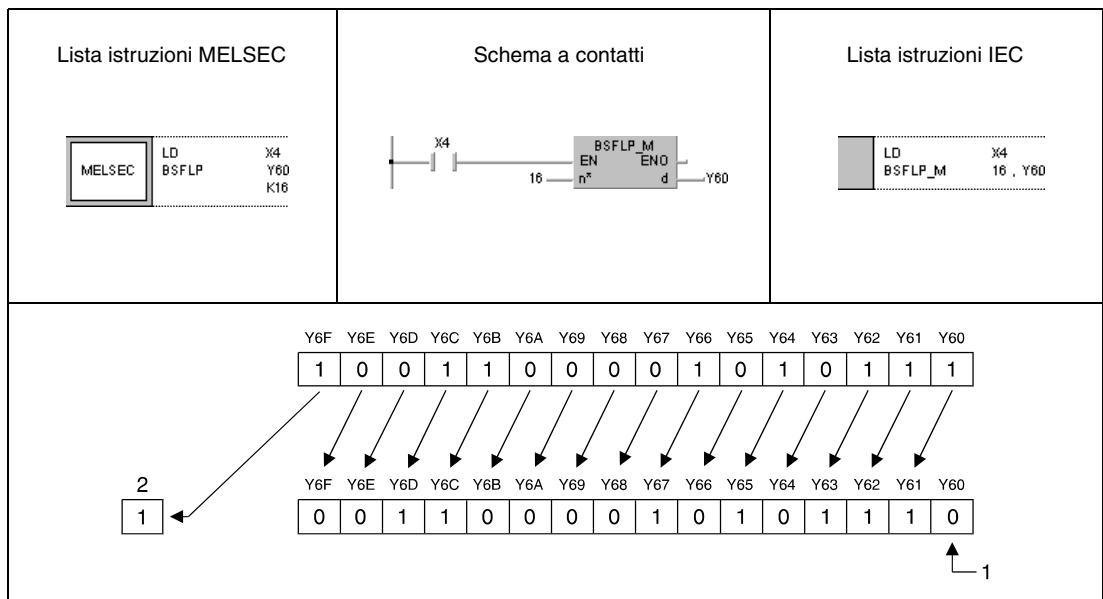


- 1 Questo bit viene impostato a 0.
- 2 Segnale di riporto

Programma di esempio 2

BSFLP

Con il fronte di salita di X4, il programma che segue scorre il contenuto delle uscite da Y60 a Y6F di un bit verso sinistra. Il contenuto dell'ultimo operando (Y6F) viene imposto sul segnale di riporto (serie A = M9012, serie Q e System Q = SM700), mentre la prima uscita (Y60) viene impostata a 0.



- 1 Questo bit viene impostato a 0.
- 2 Segnale di riporto

7.3.3 DSFR, DSFRP, DSFL, DSFLP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A


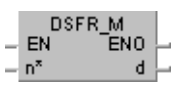
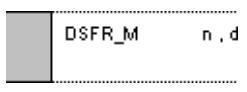
	Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Riporto	Errore			
	Operandi a bit							Operandi a word (16-bit)							Costanti						Puntatore		Livello
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z						V	K	H (16#)
d							●	●	●	●	●												
n																●	●						●

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

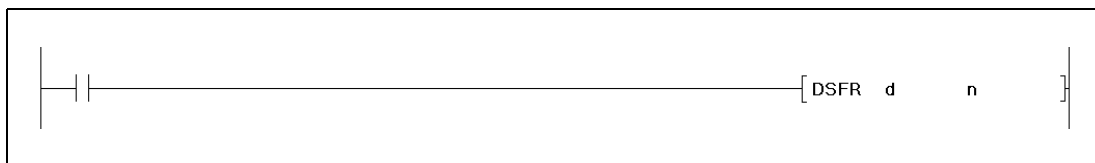
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti E			Altro
	Bit	Word		Bit	Word						
d	—	●	●	—	—	—	—	—	—	SM0	3
n	●	●	●	●	●	●	●	●	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
--	--	---

GX Developer



Variabili

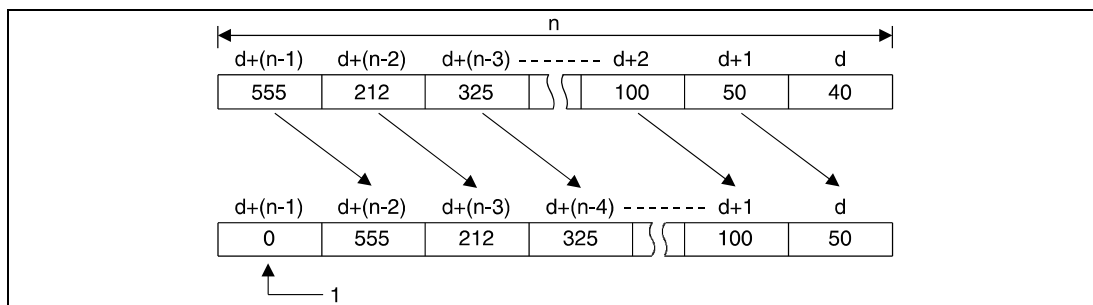
Dati impostati	Significato	Tipo dati
d	Primo indirizzo dell'operando che deve scorrere.	BIN 16-bit
n	Numero di operandi che devono scorrere.	BIN 16-bit

Funzioni Scorrimento di 1 indirizzo di n operandi a word**DSFR Scorrimento sulla destra**

L'istruzione DSFR scorre il contenuto degli operandi a word specificati di 1 indirizzo verso destra. L'operazione di scorrimento inizia dall'indirizzo dell'operando indicato da d e prosegue sugli n indirizzi successivi.

Il contenuto dell'operando più significativo viene azzerato dopo lo scorrimento.

Su timer e contatori solo il valore attuale (conteggio) può essere fatto scorrere. Il valore impostato non può essere fatto scorrere.



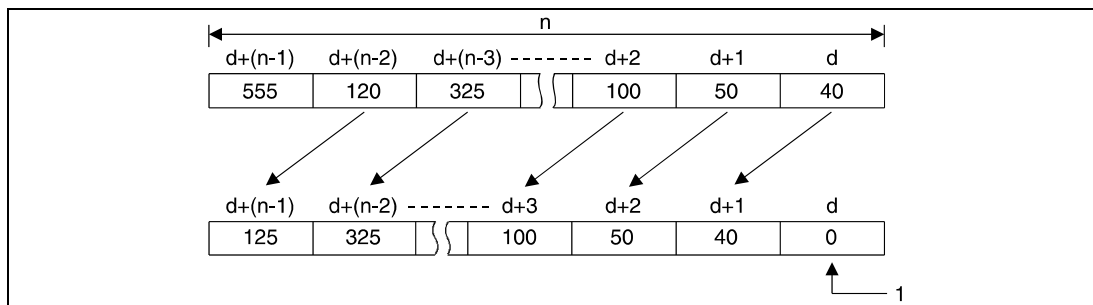
¹ Questo bit è impostato a 0.

DSFL Scorrimento sulla sinistra

L'istruzione DSFL scorre il contenuto degli operandi a word specificati di 1 indirizzo verso sinistra. L'operazione di scorrimento inizia dall'indirizzo dell'operando indicato da d e prosegue sugli n indirizzi successivi.

Il contenuto dell'operando meno significativo viene azzerato dopo lo scorrimento.

Su timer e contatori solo il valore attuale (conteggio) può essere fatto scorrere. Il valore impostato non può essere fatto scorrere.



¹ Questa word viene impostata a 0.

Errori di esecuzione

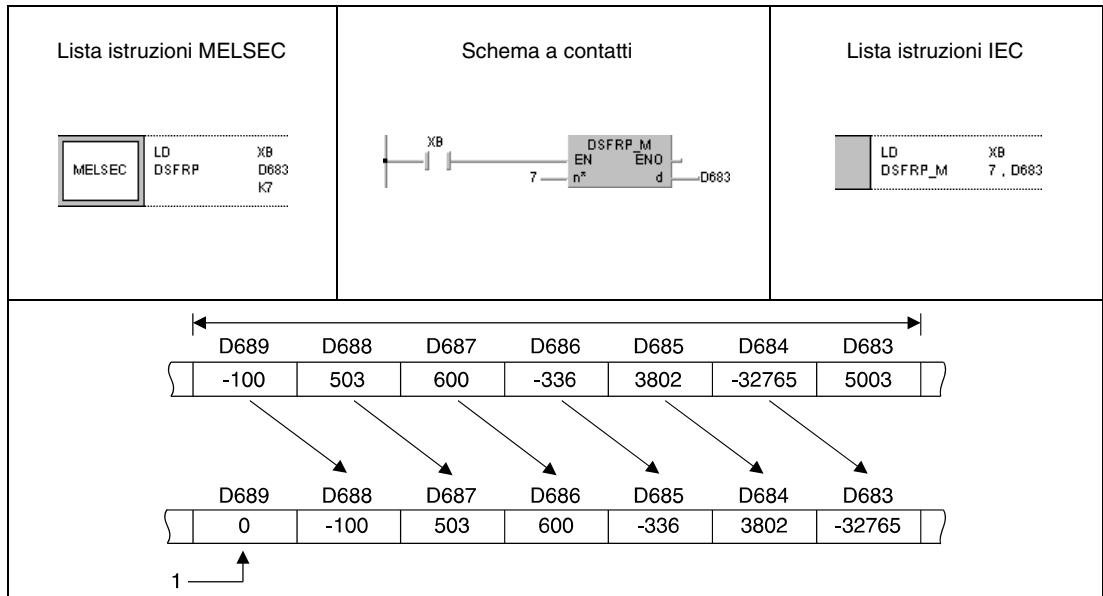
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il valore di n è negativo.
- Il valore di n supera il numero di bit disponibili per l'operando indicato da d. (serie Q e System Q = codice di errore 4101).

Programma di esempio 1

DSFRP

Con il fronte positivo di XB, il programma che segue scorre i dati dei registri da D683 a D689 di un indirizzo verso destra. D683 assume il valore di D684, D684 quello di D685, ecc. Il contenuto dell'ultimo registro dati (D689) viene azzerato.

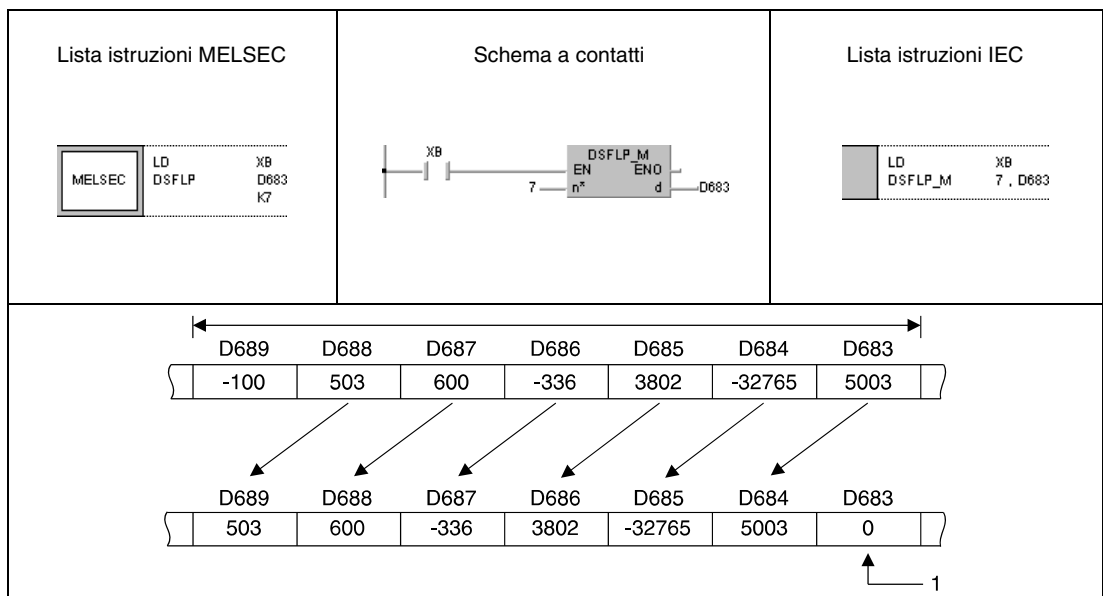


¹ Questo bit viene impostato a 0.

Programma di esempio 2

DSFLP

Con il fronte positivo di XB, il programma che segue scorre i dati dei registri da D683 a D689 di un indirizzo verso sinistra. D689 assume il valore di D688, D688 quello di D687, ecc. Il contenuto dell'ultimo registro dati (D683) viene azzerato.



¹ Questa word viene impostata a 0.

7.4 Istruzioni di elaborazione bit

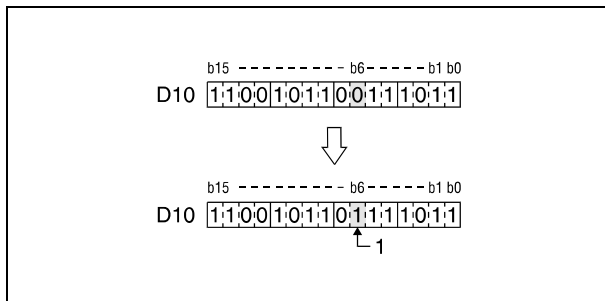
Le istruzioni di elaborazione bit modificano lo stato (set e reset) di singoli bit o di interi gruppi di bit. Con queste istruzioni è anche possibile eseguire test sullo stato dei bit in word di dati.

Sono disponibili in totale 10 istruzioni per l'elaborazione a bit:

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Set / reset di singoli bit	BSET	BSET_M
	BSETP	BSETP_M
	BRST	BRST_M
	BRSTP	BRSTP_M
Esegue il test sullo stato di bit singoli in word dati a 16/32 bit	TEST	TEST_MD
		TEST_K_MD
	TESTP	TEST_P_MD
		TEST_K_P_MD
	DTEST	DTEST_MD
		DTEST_K_MD
	DTESTP	DTEST_P_MD
		DTEST_K_P_MD
Azzera gruppi di bit	BKRST	BKRST_M
	BKRSTP	BKRSTP_M

Funzioni Attivazione / cancellazione di bit singoli**BSET Attivazione di singoli bit in un operando a word**

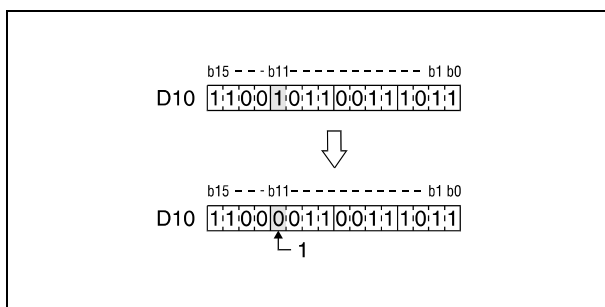
L'istruzione BSET attiva (1) l'ennesimo bit di un operando a word. L'operando n deve specificare un valore compreso fra 0 e 10 (da b0 a b15). L'operando a word è indicato da d. Se il valore di n è superiore a 15, l'istruzione BSET viene eseguita sui 4 bit più leggeri (da b0 a b3). Nella figura seguente, n è impostato a 6, per cui viene attivato il bit b6.



¹ Questo bit è attivato

BRST Azzeramento di singoli bit in un operando a word

L'istruzione BRST cancella (0) l'ennesimo bit di un operando a word. L'operando n deve specificare un valore compreso fra 0 e 10 (da b0 a b15). L'operando a word è indicato da d. Se il valore di n è superiore a 15, l'istruzione BRST viene eseguita sui 4 bit più leggeri (da b0 a b3). Nella figura seguente, n è impostato a 11, per cui viene azzerato il bit b11.



¹ Questo bit è azzerato

Esempio di programma

BRSTP/BSETP

Con il fronte di salita di XB, il programma che segue attiva (1) il bit (b3) di D8. Con il fronte di salita del contatto NC XB, il bit b8 viene azzerato.

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC																																																												
<table border="1" style="margin: auto;"> <tr> <td style="padding: 5px;">MELSEC</td> <td style="padding: 5px;">LDI XB</td> </tr> <tr> <td style="padding: 5px;"></td> <td style="padding: 5px;">BRSTP D8</td> </tr> <tr> <td style="padding: 5px;"></td> <td style="padding: 5px;">K8</td> </tr> <tr> <td style="padding: 5px;"></td> <td style="padding: 5px;">LD XB</td> </tr> <tr> <td style="padding: 5px;"></td> <td style="padding: 5px;">BSETP D8</td> </tr> <tr> <td style="padding: 5px;"></td> <td style="padding: 5px;">K3</td> </tr> </table>	MELSEC	LDI XB		BRSTP D8		K8		LD XB		BSETP D8		K3		<table border="1" style="margin: auto;"> <tr> <td style="padding: 5px;">LDN XB</td> </tr> <tr> <td style="padding: 5px;">BRSTP_M 8, D8</td> </tr> <tr> <td style="padding: 5px;">LD XB</td> </tr> <tr> <td style="padding: 5px;">BSETP_M 3, D8</td> </tr> </table>	LDN XB	BRSTP_M 8, D8	LD XB	BSETP_M 3, D8																																												
MELSEC	LDI XB																																																													
	BRSTP D8																																																													
	K8																																																													
	LD XB																																																													
	BSETP D8																																																													
	K3																																																													
LDN XB																																																														
BRSTP_M 8, D8																																																														
LD XB																																																														
BSETP_M 3, D8																																																														
<table style="margin: auto;"> <tr> <td style="padding: 5px;">D8</td> <td style="padding: 5px;"> <table style="border-collapse: collapse;"> <tr> <td style="padding: 0 5px;">b15</td><td style="padding: 0 5px;">-----</td><td style="padding: 0 5px;">b8</td><td style="padding: 0 5px;">-----</td><td style="padding: 0 5px;">b3</td><td style="padding: 0 5px;">--</td><td style="padding: 0 5px;">b0</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td> </tr> </table> </td> </tr> <tr> <td style="padding: 5px;">D8</td> <td style="padding: 5px;"> <table style="border-collapse: collapse;"> <tr> <td style="padding: 0 5px;">b15</td><td style="padding: 0 5px;">-----</td><td style="padding: 0 5px;">b8</td><td style="padding: 0 5px;">-----</td><td style="padding: 0 5px;">b3</td><td style="padding: 0 5px;">--</td><td style="padding: 0 5px;">b0</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td> </tr> </table> </td> </tr> </table>			D8	<table style="border-collapse: collapse;"> <tr> <td style="padding: 0 5px;">b15</td><td style="padding: 0 5px;">-----</td><td style="padding: 0 5px;">b8</td><td style="padding: 0 5px;">-----</td><td style="padding: 0 5px;">b3</td><td style="padding: 0 5px;">--</td><td style="padding: 0 5px;">b0</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td> </tr> </table>	b15	-----	b8	-----	b3	--	b0	0	0	1	1	0	1	0	1	1	1	1	1	1	0	0	0	0	1	0	0	1	D8	<table style="border-collapse: collapse;"> <tr> <td style="padding: 0 5px;">b15</td><td style="padding: 0 5px;">-----</td><td style="padding: 0 5px;">b8</td><td style="padding: 0 5px;">-----</td><td style="padding: 0 5px;">b3</td><td style="padding: 0 5px;">--</td><td style="padding: 0 5px;">b0</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td> </tr> </table>	b15	-----	b8	-----	b3	--	b0	0	0	1	1	0	0	1	1	1	1	1	1	1	0	0	0	1	0	0	1	1
D8	<table style="border-collapse: collapse;"> <tr> <td style="padding: 0 5px;">b15</td><td style="padding: 0 5px;">-----</td><td style="padding: 0 5px;">b8</td><td style="padding: 0 5px;">-----</td><td style="padding: 0 5px;">b3</td><td style="padding: 0 5px;">--</td><td style="padding: 0 5px;">b0</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td> </tr> </table>	b15	-----	b8	-----	b3	--	b0	0	0	1	1	0	1	0	1	1	1	1	1	1	0	0	0	0	1	0	0	1																																	
b15	-----	b8	-----	b3	--	b0																																																								
0	0	1	1	0	1	0																																																								
1	1	1	1	1	1	0																																																								
0	0	0	1	0	0	1																																																								
D8	<table style="border-collapse: collapse;"> <tr> <td style="padding: 0 5px;">b15</td><td style="padding: 0 5px;">-----</td><td style="padding: 0 5px;">b8</td><td style="padding: 0 5px;">-----</td><td style="padding: 0 5px;">b3</td><td style="padding: 0 5px;">--</td><td style="padding: 0 5px;">b0</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">1</td> </tr> </table>	b15	-----	b8	-----	b3	--	b0	0	0	1	1	0	0	1	1	1	1	1	1	1	0	0	0	1	0	0	1	1																																	
b15	-----	b8	-----	b3	--	b0																																																								
0	0	1	1	0	0	1																																																								
1	1	1	1	1	1	0																																																								
0	0	1	0	0	1	1																																																								

NOTA

Singoli bit all'interno di operandi a bit possono essere anche attivati o azzerati rispettivamente con le istruzioni SET o RST. In questo caso si devono specificare i bit dell'operando a word nell'indirizzamento dei registri. Ad esempio, il bit b8 della word dati D8 viene indirizzato come D8.8.

7.4.2 TEST, TESTP, DTEST, DTESTP


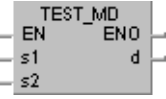
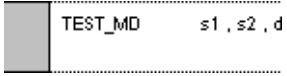
CPU

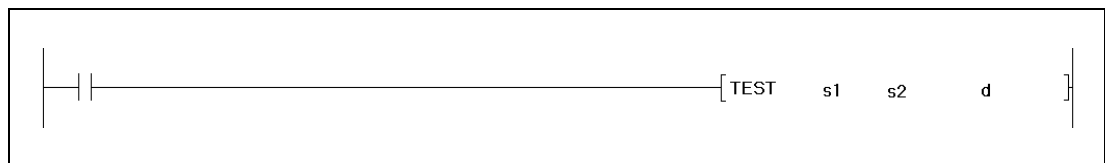
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi
MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	●	●	●	●	●	●	—	—	—	4	
s2	●	●	●	●	●	●	●	—	—		
d	●	—	—	●	—	—	—	—	—		

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
		

GX
Developer

Variabili

Dati impostati	Significato	Tipo dati
s1	Indirizzo dell'operando che contiene i bit da testare	Word
s2	Numero di bit da testare	Word
d	Indirizzo dell'operando a bit contenente lo stato del bit testato	Bit

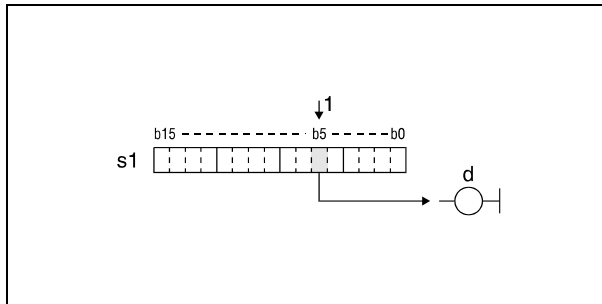
Funzioni Test dello stato di singoli bit in word dati da 16/32 bit

TEST Test di bit (16-bit)

L'istruzione TEST controlla lo stato del bit s2 in un operando a word s1. Il risultato del test viene memorizzato nell'operando a bit indicato da d.

L'operando indicato da d si attiva se il bit testato è a 1, mentre viene azzerato se il bit testato è a 0.

Il bit specificato da s2 può essere qualsiasi dei bit da b0 a b15 di una word dati. Nella figura seguente, s2 è impostato a 5, per cui viene testato lo stato del bit b5 di s1.



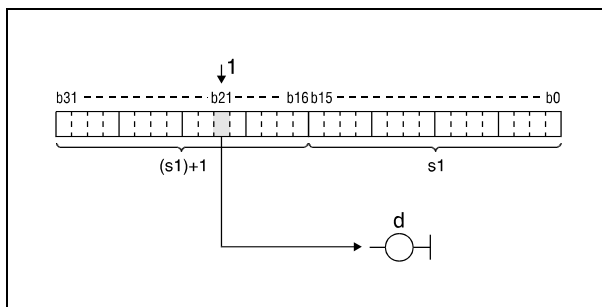
¹ Bit testato

DTEST Test di bit (32-bit)

L'istruzione DTEST controlla lo stato del bit s2 nell'operando a word s1 e (s1)+1. Il risultato del test viene memorizzato nell'operando a bit indicato da d.

L'operando indicato da d si attiva se il bit testato è a 1, mentre viene azzerato se il bit testato è a 0.

Il bit specificato da s2 può essere qualsiasi dei bit da b0 a b31 di una word dati a 32 bit. Nella figura seguente, s2 è impostato a 21, per cui viene testato lo stato del bit b21 di s1.

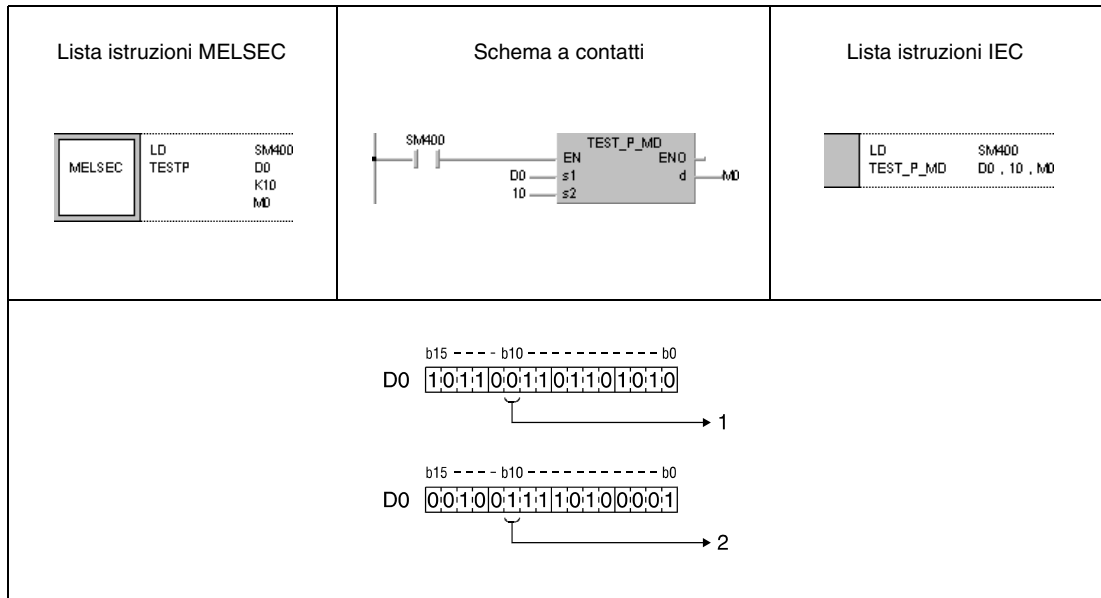


¹ Bit testato

Programma di esempio 1

TESTP

Con il fronte di salita di SM400 e in funzione del risultato del test del bit b10 della word dati a 16 bit contenuta in D0, il programma seguente azzerava o attiva il relé M0.

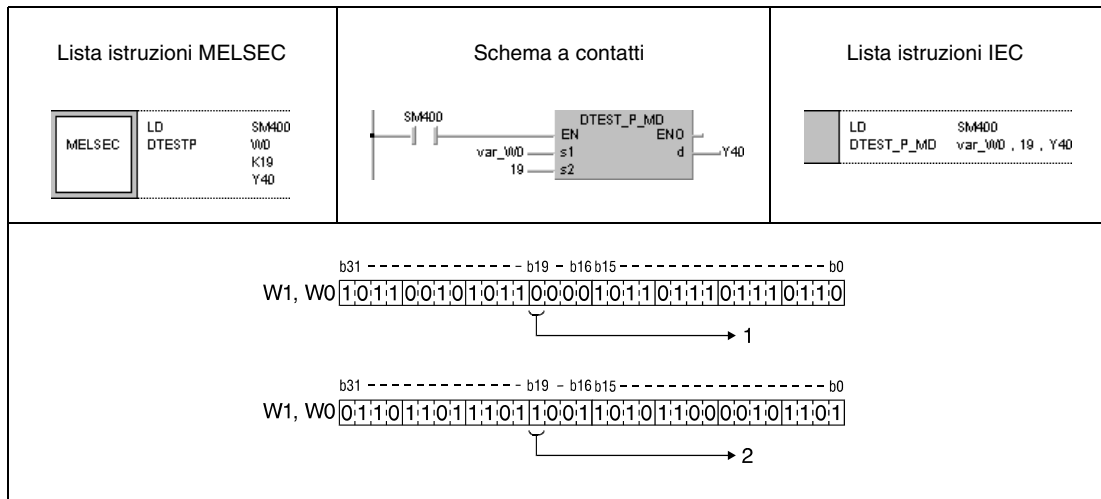


- 1 Reset
- 2 Attivazione

Programma di esempio 2

DTESTP

Con il fronte di salita di SM400 e in funzione del risultato del test del bit b19 della word dati a 32 bit contenuta in W0 e W1, il programma seguente azzerava o attiva l'uscita Y40.



- 1 Reset
- 2 Attivazione

NOTA

Il programma di esempio 2 non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Deve loper" di questo manuale.

Il bit da testare può anche essere specificato come contatto d'ingresso, invece di usare una istruzione TEST (vedi figura).




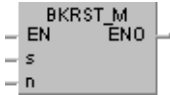

7.4.3 BKRST, BKRSTP**CPU**

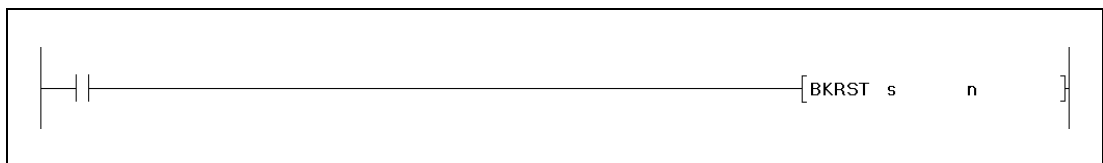
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	●	●	●	—	—	—	—	—	SM0	3	
n	●	●	●	●	●	●	●	—			

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
 BKRST s n	 BKRST_M EN ENO s n	 BKRST_M s, n

GX Developer**Variabili**

Dati impostati	Significato	Tipo dati
s	Primo indirizzo dell'operando da azzerare.	Bit
n	Numero degli operandi da disattivare.	BIN 16-bit

Funzioni **Reset di gruppi di bit**

BKRST Istruzione di disattivazione

L'istruzione BKRST disattiva (0) n bit nell'operando indicato da s.

Nel caso di annunciatori (F) il numero di annunciatori n contenuti in s viene cancellato e il contenuto dei registri da SD64 a SD79 viene azzerato in corrispondenza degli annunciatori azzerati. I dati rimanenti vengono fatti scorrere in avanti. Inoltre, il numero degli annunciatori attivi nei registri da SD64 a SD79 viene memorizzato nel registro SD63.

In caso di timer (T) e contatori (C), dopo l'esecuzione di questa istruzione, i valori impostati di n timer e contatori vengono azzerati ed i contatti aperti.

Per tutti gli altri operandi a bit, viene azzerato il numero n di bobine o contatti.

Se l'operando indicato è già a zero, il suo stato rimane invariato dopo l'esecuzione dell'istruzione.

Errori di esecuzione

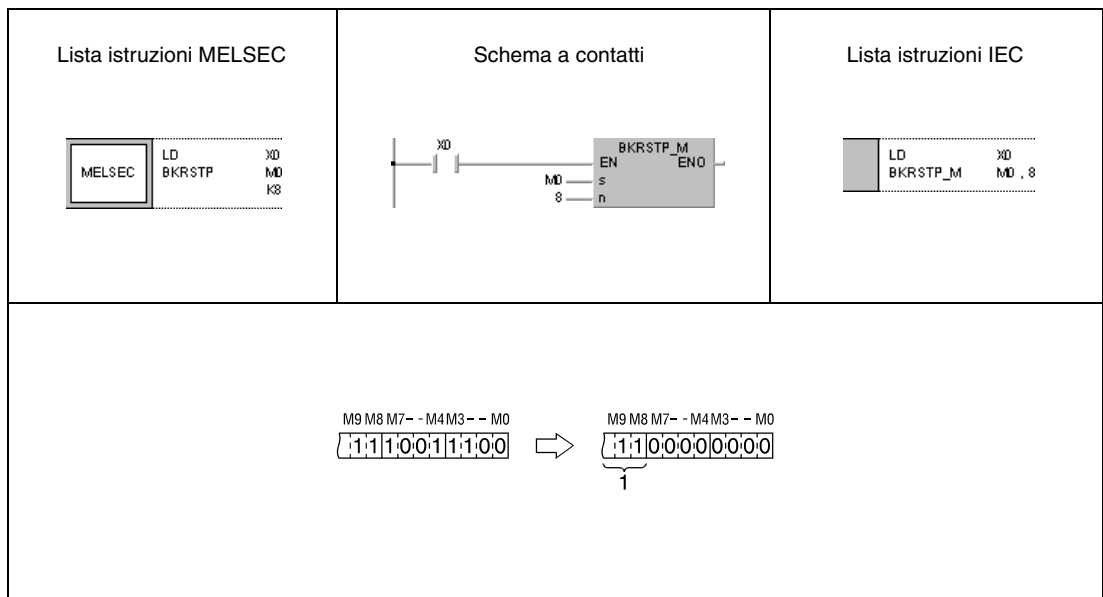
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il valore di n supera il numero di bit dell'operando indicato da s (codice di errore 4101).

Programma di esempio 1

BKRSTP

Con il fronte di salita di X0, il programma seguente azzerava i relé dal M0 a M7.



¹ Questi bit non vengono modificati

Programma di esempio 2

BKRSTP

Con il fronte di salita di X20, il programma che segue azzerava i bit dal bit b2 di D10 al bit b1 di D11.

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC												
<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> <table style="border-collapse: collapse; margin: 0 auto;"> <tr> <td style="border: 1px solid black; padding: 2px;">MELSEC</td> <td style="padding: 2px;">LD</td> <td style="padding: 2px;">X20</td> </tr> <tr> <td></td> <td style="padding: 2px;">BKRSTP</td> <td style="padding: 2px;">D10.2 K16</td> </tr> </table> </div>	MELSEC	LD	X20		BKRSTP	D10.2 K16		<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> <table style="border-collapse: collapse; margin: 0 auto;"> <tr> <td style="padding: 2px;">LD</td> <td style="padding: 2px;">X20</td> </tr> <tr> <td style="padding: 2px;">BKRSTP_M</td> <td style="padding: 2px;">D10.2, 16</td> </tr> </table> </div>	LD	X20	BKRSTP_M	D10.2, 16		
MELSEC	LD	X20												
	BKRSTP	D10.2 K16												
LD	X20													
BKRSTP_M	D10.2, 16													
<table style="margin: 0 auto;"> <tr> <td style="text-align: right; padding-right: 10px;">D10</td> <td style="text-align: center; padding: 0 10px;">b15 ----- b8b7 ----- b2b1b0</td> <td style="border: 1px solid black; padding: 2px;">1·1·1·0·0·0·0·1·1·1·1·1·1·1·0</td> <td style="font-size: 2em; vertical-align: middle;">➔</td> <td style="text-align: center; padding: 0 10px;">b15 ----- b8b7 ----- b2b1b0</td> <td style="border: 1px solid black; padding: 2px;">0·0·0·0·0·0·0·0·0·0·0·0·0·0·1·0</td> </tr> <tr> <td style="text-align: right; padding-right: 10px;">D11</td> <td style="text-align: center; padding: 0 10px;">b15 ----- b8b7 ----- b1b0</td> <td style="border: 1px solid black; padding: 2px;">0·0·0·1·0·1·0·0·1·1·1·1·1·1·1·1</td> <td></td> <td style="text-align: center; padding: 0 10px;">b15 ----- b8b7 ----- b1b0</td> <td style="border: 1px solid black; padding: 2px;">0·0·0·1·0·1·0·0·1·1·1·1·1·1·1·0</td> </tr> </table>			D10	b15 ----- b8b7 ----- b2b1b0	1·1·1·0·0·0·0·1·1·1·1·1·1·1·0	➔	b15 ----- b8b7 ----- b2b1b0	0·0·0·0·0·0·0·0·0·0·0·0·0·0·1·0	D11	b15 ----- b8b7 ----- b1b0	0·0·0·1·0·1·0·0·1·1·1·1·1·1·1·1		b15 ----- b8b7 ----- b1b0	0·0·0·1·0·1·0·0·1·1·1·1·1·1·1·0
D10	b15 ----- b8b7 ----- b2b1b0	1·1·1·0·0·0·0·1·1·1·1·1·1·1·0	➔	b15 ----- b8b7 ----- b2b1b0	0·0·0·0·0·0·0·0·0·0·0·0·0·0·1·0									
D11	b15 ----- b8b7 ----- b1b0	0·0·0·1·0·1·0·0·1·1·1·1·1·1·1·1		b15 ----- b8b7 ----- b1b0	0·0·0·1·0·1·0·0·1·1·1·1·1·1·1·0									

7.5 Istruzioni per elaborazione dati

Le istruzioni di elaborazione dati ricercano dati in operandi specifici, controllano il numero di bit a uno, codificano e decodificano dati (ad es. per display a 7 segmenti), disaggregano e aggregano dati, ricercano valori minimi e massimi, ordinano dati e calcolano i totali do blocchi dati a 16 e 32 bit.

Sono disponibili in totale 41 istruzioni per elaborazione dati:

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Ricerca dati a 16/32 bit	SER	SER_M
	SERP	SERP_M
	DSER	DSER_M
	DSERP	DSERP_M
Controllo bit dati (16-/32-bit)	SUM	SUM_M
	SUMP	SUMP_M
	DSUM	DSUM_M
	DSUMP	DSUMP_M
Codifica/decodifica dati	DECO	DECO_M
	DECOP	DECOP_M
	ENCO	ENCO_M
	ENCOP	ENCOP_M
Decodifica 7 segmenti	SEG	SEG_M
Separa/unisce word dati a 16 bit (unità da 4-bit)	DIS	DIS_M
	DISP	DISP_M
	UNI	UNI_M
	UNIP	UNIP_M
Separa/unisce dati a 16 bit (unità con bit variabili)	NDIS	NDIS_M
	NDISP	NDISP_M
	NUNI	NUNI_M
	NUNIP	NUNIP_M
Separa/unisce dati a 16 bit (unità byte)	WTOB	WTOB_MD
		WTOB_K_MD
	WTOBP	WTOB_P_MD
		WTOB_K_P_MD
	BTOW	BTOW_MD
		BTOW_K_MD
BTOWP	BTOW_P_MD	
	BTOW_K_P_MD	
Ricerca valore massimo in dati a 16-/32-bit	MAX	MAX_M
	MAXP	MAXP_M
	DMAX	DMAX_M
	DMAXP	DMAXP_M

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Ricerca valore minimo in dati a 16-/32-bit	MIN	MIN_M
	MINP	MINP_M
	DMIN	DMIN_M
	DMINP	DMINP_M
Ordina dati a 16-/32-bit	SORT	SORT_M
	SORTP	SORTP_M
	DSORT	DSORT_M
	DSORTP	DSORTP_M
Calcola il totale di blocchi dati BIN 16/32-bit	WSUM	WSUM_M
	WSUMP	WSUMP_M
	DWSUM	DWSUM_M
	DWSUMP	DWSUMP_M

7.5.1 SER, SERP, DSER , DSERP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

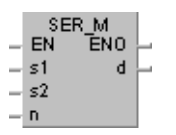
	Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Rapporto	Errore					
	Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello										
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z						V	K	H (16#)	P	I
s1							●	●	●	●	●	●	●	●	●	●	●					9 1	●		●
s2						●	●	●	●	●															
n																●	●								

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

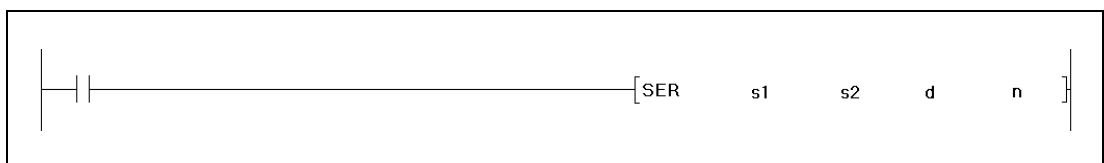
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J		Modulo funzione speciale U	Registro indice Zn	Costanti K, H (16#)			Altro U
	Bit	Word		Bit	Word						
s1	●	●	●	●	●	●	●	●	—	SM0	5
s2	—	●	●	—	—	—	—	—			
d	—	●	●	—	●	●	●	—			
n	●	●	●	●	●	●	●	●			

GX IEC Developer

<p>Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 10px auto;"> <p>MELSEC</p> </div> <p>SER s1 s2 d n</p>	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 10px auto;"> <p>SER_M</p> </div> <p>SER_M s1, s2, n, d</p>
---	--	--

GX Developer



NOTA

Le CPU della serie A memorizzano sempre i risultati della ricerca nei registri A0 e A1. Per questa ragione, nessun operando viene utilizzato quando si usano queste istruzioni con la serie A.

Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s1	Valore del dato da ricercare, o primo indirizzo dell'operando che contiene questo valore.	Word	ANY16
s2	Dati su cui deve essere eseguita la ricerca o primo numero dell'operando che contiene questi dati.		ANY16/ANY32
d	Primo numero di operando che contiene il risultato della ricerca. serie A: solo operandi A0 e A1.		Array [1..2] di ANY16/ANY32
n	Numero di operandi su cui deve essere eseguita la ricerca.		ANY16

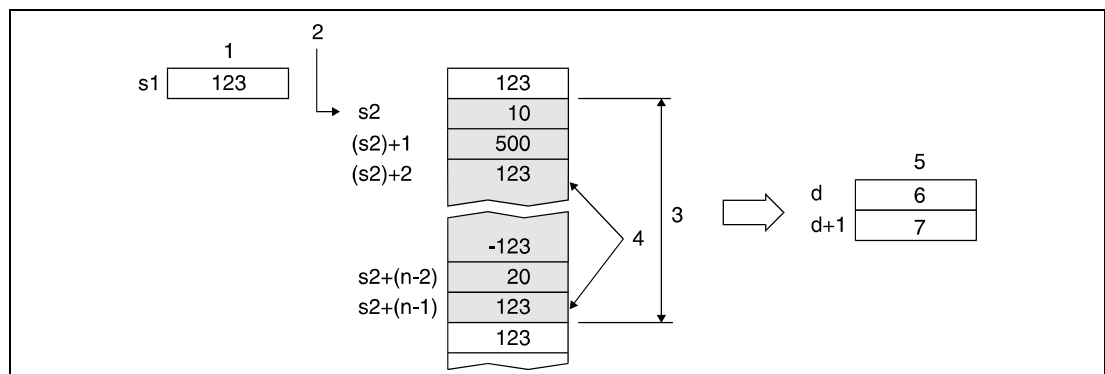
Funzioni**Ricerca dati****SER (serie A e Q/System Q) / SERP (serie Q e System Q) Ricerca dati a 16-bit**

L'istruzione SER consente la ricerca di un dato specifico in un campo di ricerca determinato. L'operazione di ricerca inizia dall'indirizzo del primo operando indicato da s2. Il codice da ricercare è specificato da s1. La dimensione del campo di ricerca, cioè il numero di operandi interessati, è indicato da n.

Le CPU della serie Q o del System Q memorizzano il risultato della ricerca in d e d+1, come array [1..2] di ANY16.

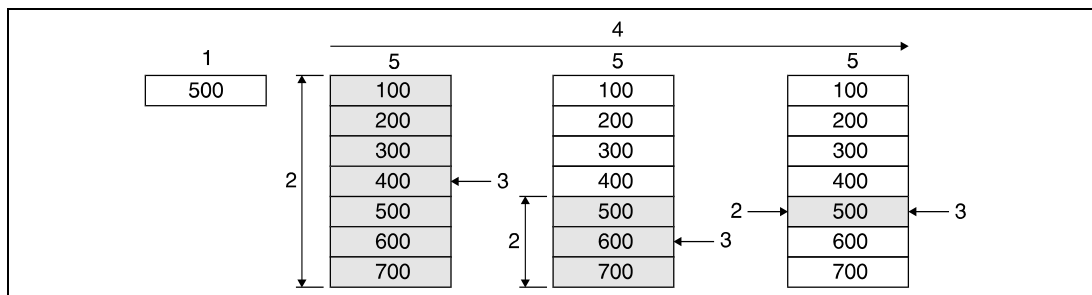
Al termine dell'operazione di ricerca, la posizione del primo operando che contiene il valore ricercato viene memorizzata in d in array[1], mentre array[2] in d+1 contiene il numero dei dati che contengono il codice ricercato.

Le CPU della serie A memorizzano la posizione del primo operando che contiene il dato ricercato nel registro A0. Il numero di occorrenze viene memorizzato in A1.



- 1 Codice da ricercare
- 2 Inizio ricerca
- 3 Campo di ricerca (n blocchi)
- 4 Dati corrispondenti
- 5 Risultato ricerca
- 6 Posizione della corrispondenza
- 7 Numero occorrenze

Se il valore di n è inferiore o uguale a 0, l'operazione di ricerca non viene eseguita. Se non vengono trovati dati corrispondenti, il contenuto di d e d+1 (A0 e A1 per la serie A) è 0.

NOTA*Serie Q e System Q**Se i dati da ricercare sono memorizzati in ordine crescente, il tempo di ricerca può essere abbreviato attivando il relé speciale SM702.***SM702 ON:***Il campo di ricerca viene diviso in due e il valore del codice da ricercare determina in quale metà il codice deve essere ricercato. Questa metà viene divisa ancora in due per la decisione successiva. Questa operazione viene ripetuta fino a trovare il valore desiderato.*

- ¹ Codice da ricercare
- ² Campo di ricerca
- ³ Confronto con codice da cercare
- ⁴ Sequenza di elaborazione
- ⁵ Ricerca dati

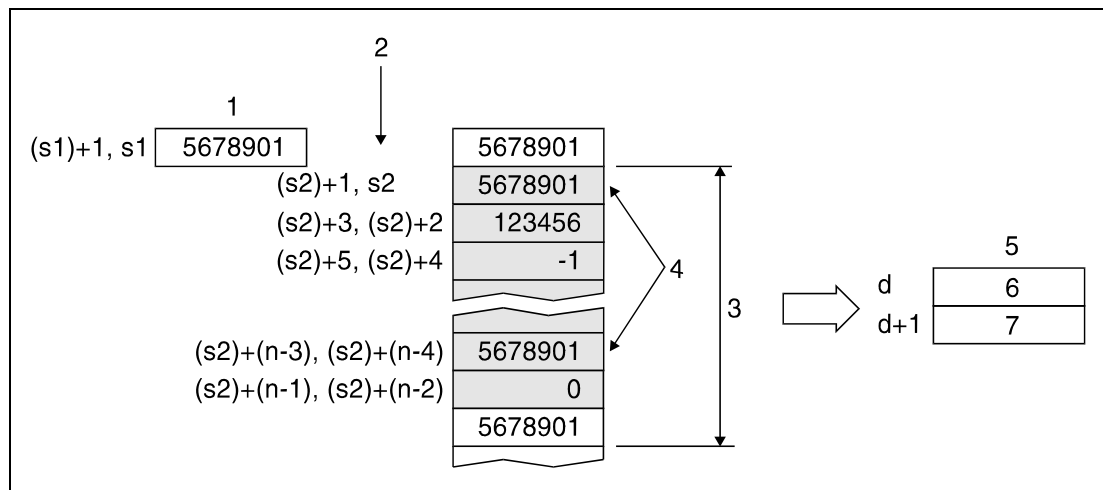
SM702 OFF:*Il confronto del dato da ricercare con ciascuno dei valori inizia dall'inizio del campo di ricerca.**Se il campo da ricercare non è ordinato in senso ascendente, il risultato con SM702 attivo non è garantito.*

DSER / DSERP (serie Q e System Q) Ricerca dati 32-bit

L'istruzione DSER consente la ricerca di un dato specifico in un campo di ricerca determinato. L'operazione di ricerca inizia dall'indirizzo del primo operando indicato da s2 (2 x n operandi). Il codice da ricercare è specificato da s1 e (s1)+1. La dimensione del campo di ricerca, cioè il numero di operandi interessati, è specificato da n.

Il risultato della ricerca è memorizzato in d e d+1 come array [1..2] di ANY16.

Al termine della ricerca, la posizione del primo operando che corrisponde al valore ricercato viene memorizzata nell'array meno significativo (d). L'array più significativo (d+1) contiene il numero di occorrenze del valore ricercato.



- 1 Codice da ricercare
- 2 Inizio ricerca
- 3 Campo di ricerca (2 x n)
- 4 Dati corrispondenti
- 5 Risultato ricerca
- 6 Posizione della corrispondenza
- 7 Numero occorrenze

Se il valore di n è inferiore o uguale a 0, l'operazione di ricerca non viene eseguita. Se non viene trovata nessuna corrispondenza, il contenuto di d e d+1 è 0.

Errori di esecuzione

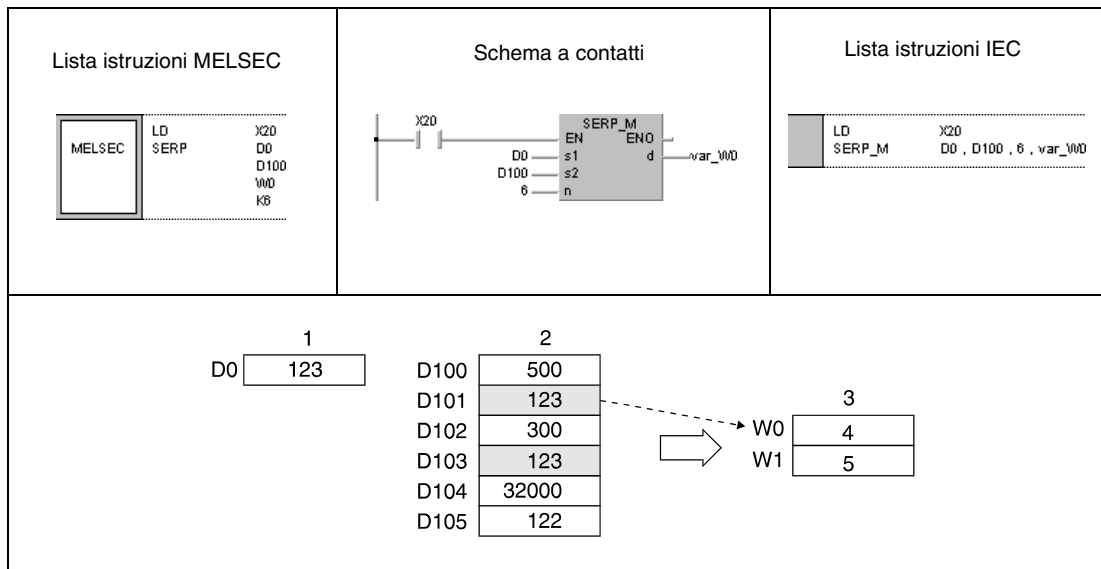
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il campo di ricerca indicato da n, iniziando da s2, supera il campo operandi ammesso (serie Q e System Q = codice di errore 4101).

Per dettagli sulla indicizzazione, consultare il capitolo 3.6.

Programma di esempio 1 SERP (serie Q e System Q)

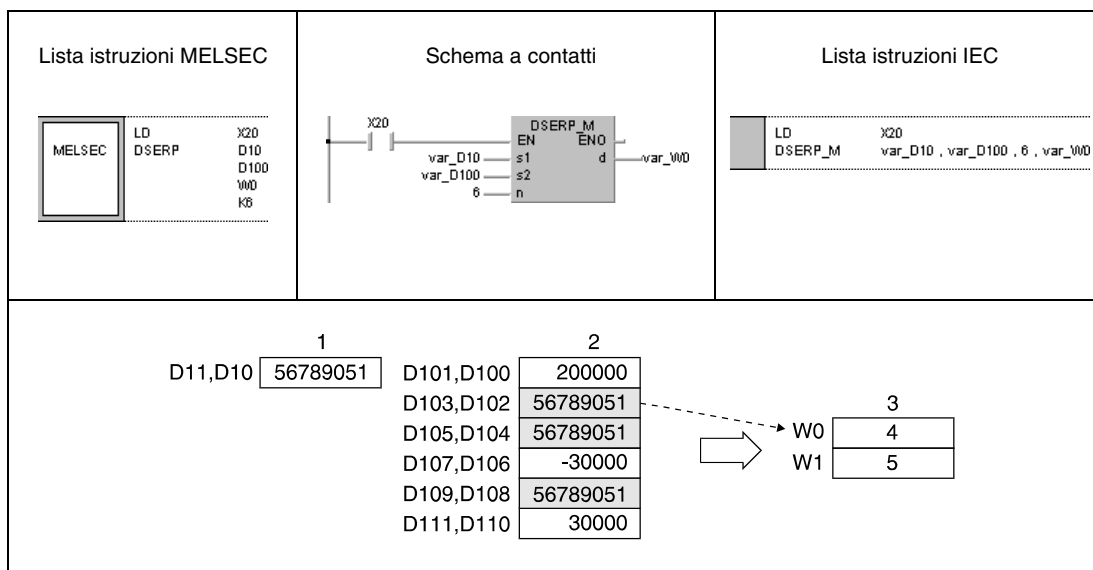
Sul fronte di salita di X20, il programma che segue confronta i dati da D100 a D105 con il valore del dato in D0. La prima posizione corrispondente viene memorizzata in W0. Il numero di occorrenze viene memorizzato in W1.



- ¹ Codice da ricercare
- ² Campo di ricerca
- ³ Risultato ricerca
- ⁴ Posizione della prima corrispondenza
- ⁵ Numero occorrenze

Programma di esempio 2 DSERP (serie Q e System Q)

Sul fronte di salita di X20, il programma che segue confronta i dati da D100 a D111 con il valore del dato in D11 e D10. La prima posizione corrispondente viene memorizzata in W0. Il numero di occorrenze viene memorizzato in W1.



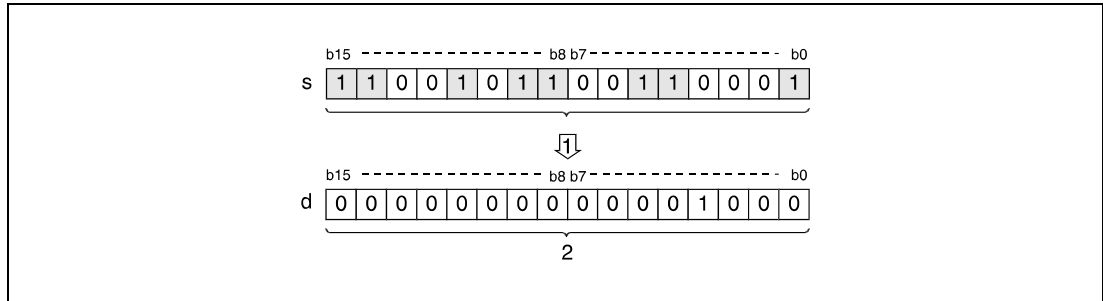
- ¹ Codice da ricercare
- ² Campo di ricerca
- ³ Risultato ricerca
- ⁴ Posizione della prima corrispondenza
- ⁵ Numero occorrenze

NOTA

Questi programmi non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provocano messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

Funzioni **Controllo bit dati****SUM (16-bit)**

L'istruzione SUM determina il numero di bit a 1 in una word dati a 16 bit. Il campo di operandi da controllare è indicato da s. Il numero di bit a 1 viene memorizzato in d (A0).

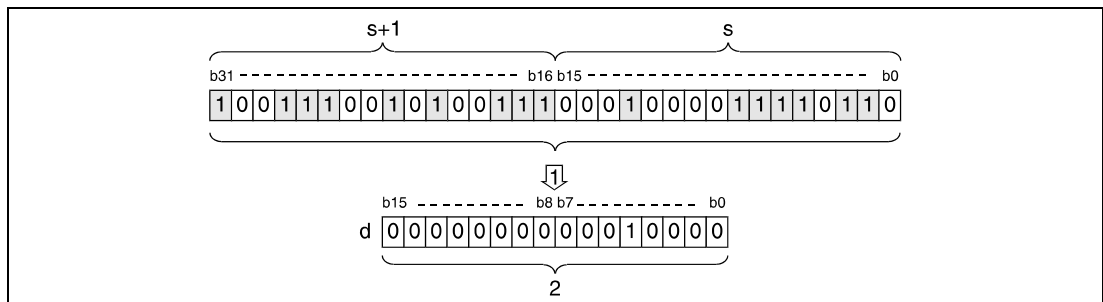


¹ Conteggio bit a 1

² Codice binario del numero di bit a 1

DSUM (32-bit)

L'istruzione DSUM determina il numero di bit a 1 in una word dati a 32 bit. Il campo di operandi da controllare è indicato da s. Il numero di bit a 1 viene memorizzato in d (A0).



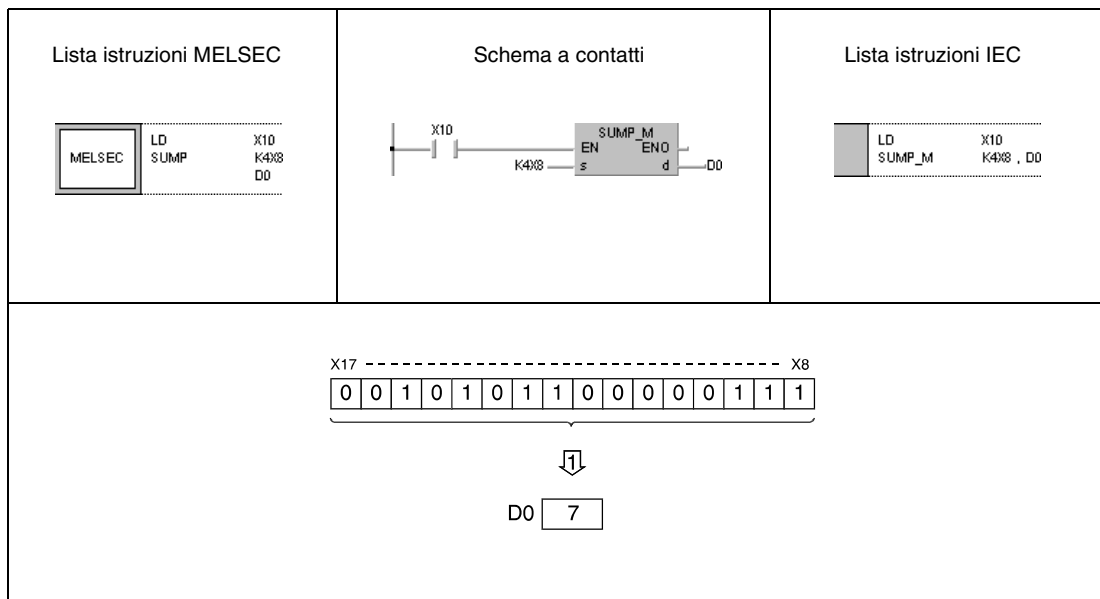
¹ Conteggio bit a 1

² Codice binario del numero di bit a 1

Programma di esempio 1

SUMP (serie Q e System Q)

Con il fronte positivo di X10, il programma seguente determina il numero di ingressi attivi da X8 a X10. Il risultato è memorizzato in D0.

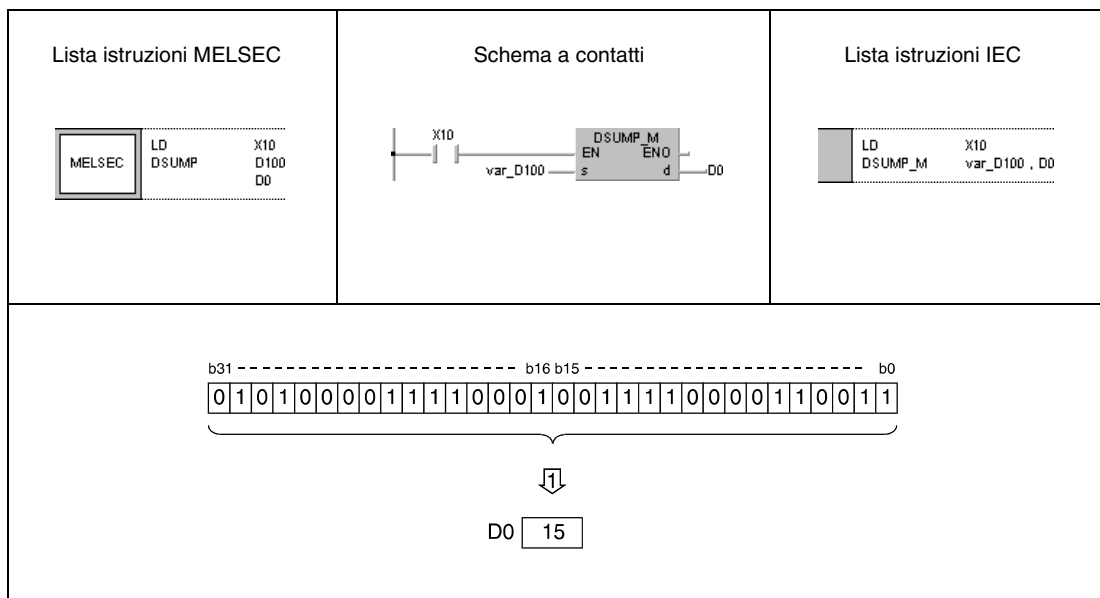


¹ Memorizzazione numero bit a 1 in D0

Programma di esempio 2

DSUMP (serie Q e System Q)

Con il fronte positivo di X10, il programma seguente determina il numero di bit a 1 da D100 a D101. Il risultato è memorizzato in D0.



¹ Memorizzazione numero bit a 1 in D0

NOTA

Il programma di esempio 2 non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

Funzioni **Decodifica da 8 a 256 bit**

DECO Decodifica dati

L'istruzione DECO decodifica i dati contenuti nell'operando indicato da s. Il dato binario codificato viene decodificato come numero decimale. Questo numero decimale (≤ 256) indica il bit x (bx), corrispondente al peso 2^x da attivare nell'operando specificato da d. Il numero degli operandi dell'indirizzo s che contengono il dato codificato è indicato da n.

La variabile n deve contenere un valore nel campo da 0 a 8.

Se n = 0 l'istruzione non viene eseguita e gli operandi contenuti negli indirizzi specificati non vengono modificati.

Un operando a bit viene come bit singolo, mentre un operando a word, come valore a 16 bit.

Errori di esecuzione

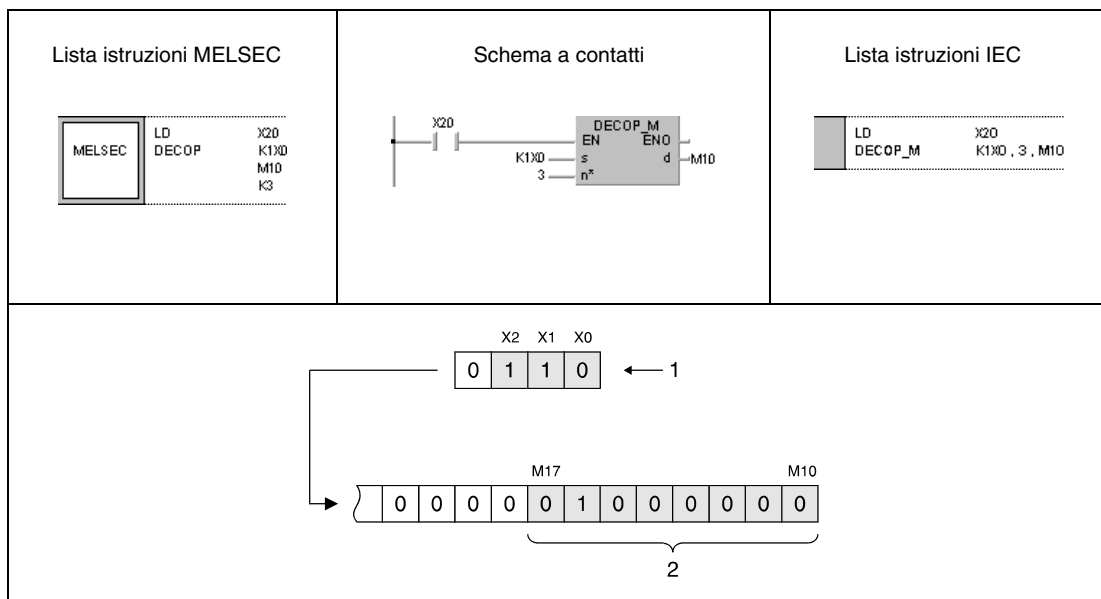
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- La variabile n non contiene un valore fra 0 e 8 (serie Q e System Q = codice di errore 4100).
- Il bit x supera il campo di indirizzamento ammesso (serie Q e System Q = codice di errore 4100).

Esempio di programma

DECOP

Con il fronte positivo di X20, il programma che segue decodifica i dati da X0 a X2. Il risultato viene memorizzato da M10 a M14. Il numero binario 6 è contenuto da X0 a X2, per cui il bit b6 (M16) del campo da M10 a M17 viene attivato.



¹ Valore binario 6

² Se il valore binario è specificato su 4 bit, vengono occupati 8 bit per la rappresentazione

NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.5.4 ENCO, ENCO^P

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

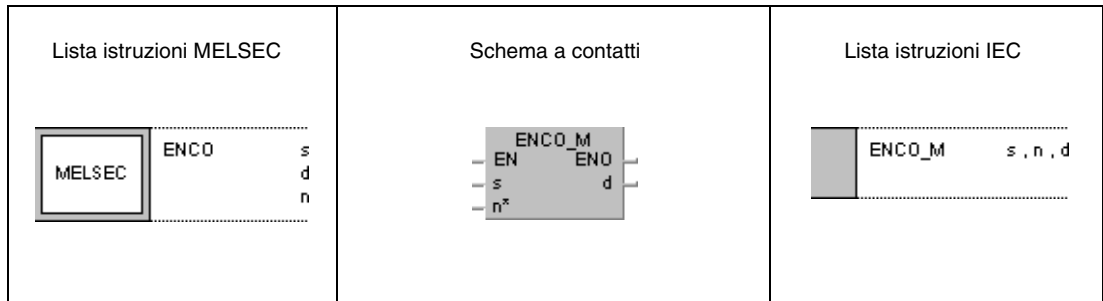
	Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Rapporto	Errore				
	Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore	Livello										
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z						V	K	H (16#)	P
s	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●								
d							●	●	●	●	●	●	●	●	●	●							●	
n																	●	●						

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

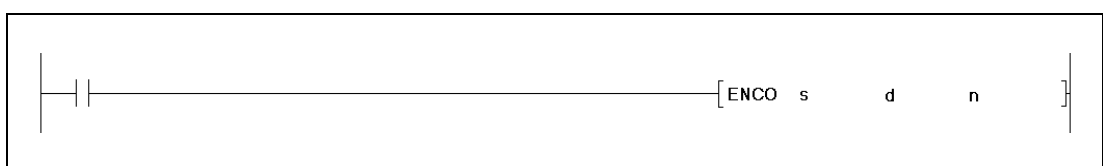
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	●	●	●	—	—	—	—	●	—	SM0	4
d	●	●	●	●	●	●	●	—	—		
n	●	●	●	●	●	●	●	●	—		

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Dato decodificato o operando che contiene il dato.	BIN 16-bit
d	Primo numero dell'operando che memorizza il dato codificato.	
n	Numero di bit che contengono il valore codificato	

Funzioni **Codifica da 256 a 8 bit**

ENCO Codifica dati

L'istruzione ENCO codifica i dati di una serie fino a 256 bit in una sequenza dati binari da 8 bit. L'indirizzo del primo operando che contiene i dati da codificare è specificato da s. Il bit x specificato da s indica il valore decimale che verrà memorizzato in d dopo averlo codificato in binario. Il numero di bit di d che contengono il dato codificato viene indicato da n.

La variabile n deve contenere un valore nel campo da 0 a 8.

Se n = 0 l'istruzione non viene eseguita e gli operandi contenuti negli indirizzi specificati non vengono modificati.

Un operando a bit viene come bit singolo, mentre un operando a word, come valore a 16 bit.

Se è attivo più di un bit, l'elaborazione inizia dal bit più significativo.

Errori di esecuzione

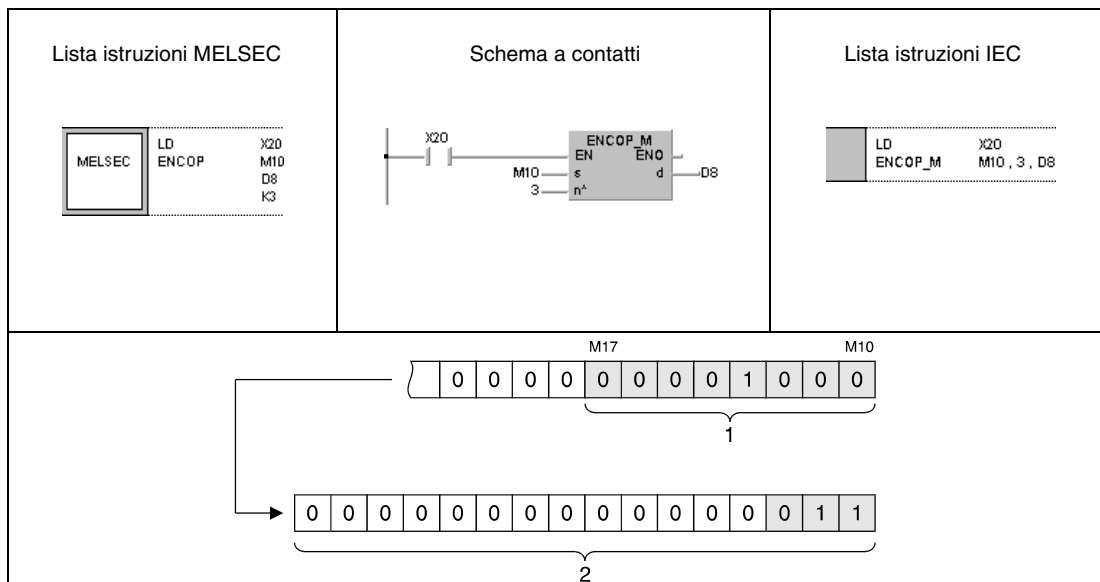
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- La variabile n non contiene un valore fra 0 e 8 (serie Q e System Q = codice di errore 4100).
- Tutti i bit di s fino al bit x sono 0 quando viene eseguita l'istruzione ENCO.
- Il valore x del bit a 1 in s supera il campo che può essere rappresentato in binario con 8 bit (serie Q e System Q = codice di errore 4101).
- Tutti i bit in s fino al bit x sono identici a d (serie Q e System Q = codice di errore 4100).

Esempio di programma

ENCOP

Con il fronte positivo di X20, il programma seguente legge i dati da M10 a M17 e memorizza il valore codificato in binario in D8.



¹ Se il bit a 1 viene codificato con 4 bit, è possibile rappresentare un campo di 8 bit.

² Numero 3 codificato in binario per il bit 3 a 1 (M13)

NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.5.5 SEG, SEGP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
● ¹	● ¹	● ¹	● ¹	● ¹	●

¹ L'istruzione SEG esegue la decodifica a 7 segmenti, solo se il relé interno M9052 NON è attivo. Se il relé interno M9052 è attivo, l'istruzione SEG serve per un rinfresco parziale.

Operandi MELSEC A

	Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Rapporto	Errore						
	Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello											
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z				V	K	H (16#)	P	I	N	M9012	M9010 M9011
s	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●					K1	7	●		
d		●	●	●	●	●	●	●	●	●	●	●	●	●	●							● ¹	● ²			

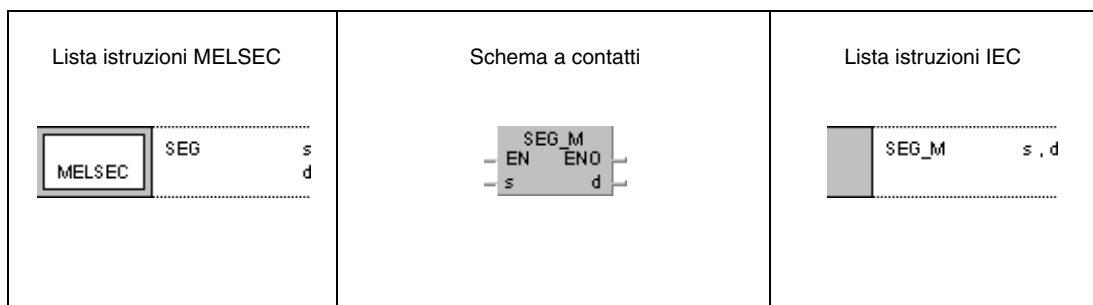
¹ Con le CPU A3H, A3M, o AnN la lunghezza di blocco può essere impostata fra K1 e K4. Su tutte le altre CPU l'indicazione della lunghezza blocco viene ignorata e viene assunto automaticamente il valore K2 (8 bit).

² Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

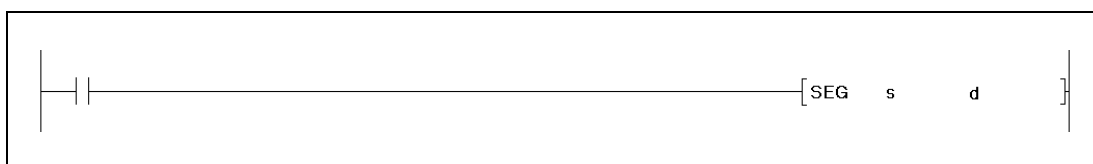
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	●	●	—	—	3
d	●	●	●	●	●	●	●	—	—	—	

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Dato decodificato o primo indirizzo dell'operando che contiene il dato.	BIN 16-bit
d	Primo numero dell'operando che memorizza il dato a 7 segmenti.	

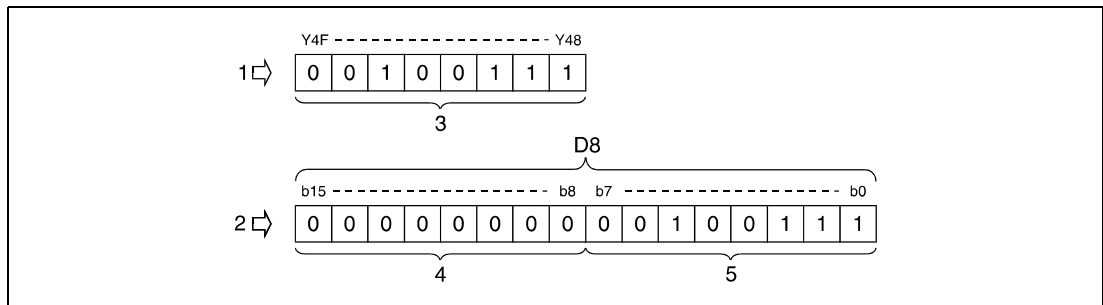
Funzioni **Decodifica 7 segmenti**

SEG (serie A e serie Q/System Q) / SEGP (serie Q e System Q)
Decodifica valore binario a 4 cifre

L'istruzione SEG converte un valore binario a 4 cifre nel codice a 7 segmenti adatto per la visualizzazione di valori da 0 a F. Il valore del dato o l'indirizzo dell'operando che contiene il dato da codificare è specificato da s. Il codice a 7 segmenti viene memorizzato in d.

Se il dato codificato a 7 segmenti viene inviato in uscita su operandi a bit, l'indirizzo del primo operando e la dimensione del campo di operandi deve essere sempre specificata in d. Se indica un operando a word, è necessario solo l'indirizzo dell'operando.

La memorizzazione dei dati su diversi operandi a bit o su un operando a word è definita dallo schema seguente:



- ¹ Operando a bit
- ² Operando a word
- ³ 8-bit
- ⁴ Questi bit vengono sempre impostati a 0
- ⁵ Dato a 7 segmenti

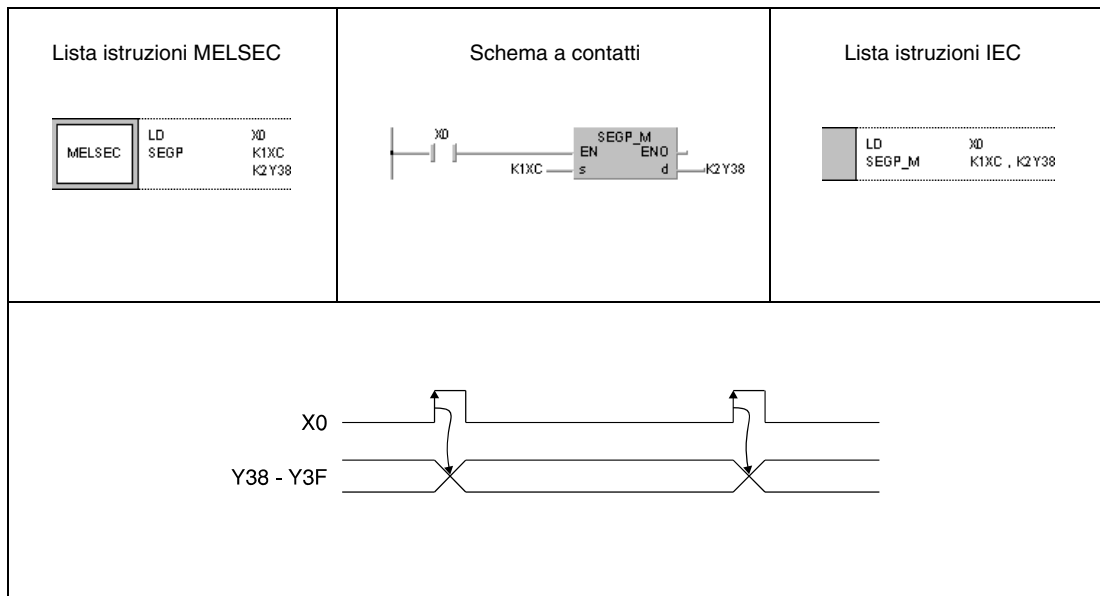
Dato a 7 segmenti

La tabella seguente mostra una panoramica dei dati a 7 segmenti in funzione delle maschere di bit del dato di partenza. Il primo bit (b0) del dato a 7 segmenti rappresenta lo stato del primo operando a bit, o lo stato del bit meno significativo dell'operando a word.

s		Assegnazione dei segmenti	d								Visualizzazione
HEX	Maschera di bit		B7	B6	B5	B4	B3	B2	B1	B0	
0	0000		0	0	1	1	1	1	1	1	0
1	0001		0	0	0	0	0	1	1	0	1
2	0010		0	1	0	1	1	0	1	1	2
3	0011		0	1	0	0	1	1	1	1	3
4	0100		0	1	1	0	0	1	1	0	4
5	0101		0	1	1	0	1	1	0	1	5
6	0110		0	1	1	1	1	1	0	1	6
7	0111		0	0	1	0	0	1	1	1	7
8	1000		0	1	1	1	1	1	1	1	8
9	1001		0	1	1	0	1	1	1	1	9
A	1010		0	1	1	1	0	1	1	1	A
B	1011		0	1	1	1	1	1	0	0	B
C	1100		0	0	1	1	1	0	0	1	C
D	1101		0	1	0	1	1	1	1	0	D
E	1110		0	1	1	1	1	0	0	1	E
F	1111		0	1	1	1	0	0	0	1	F

Esempio di programma SEGP (serie Q e System Q)

Con il fronte positivo di X0, il programma seguente trasforma lo stato degli ingressi da XC a XF nel codice a 7 segmenti inviato sulle uscite da Y38 a Y3F. Lo stato delle uscite da Y38 a Y3F viene mantenuto fino a quando non viene sovrascritto con altri dati.



7.5.6 DIS, DISP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

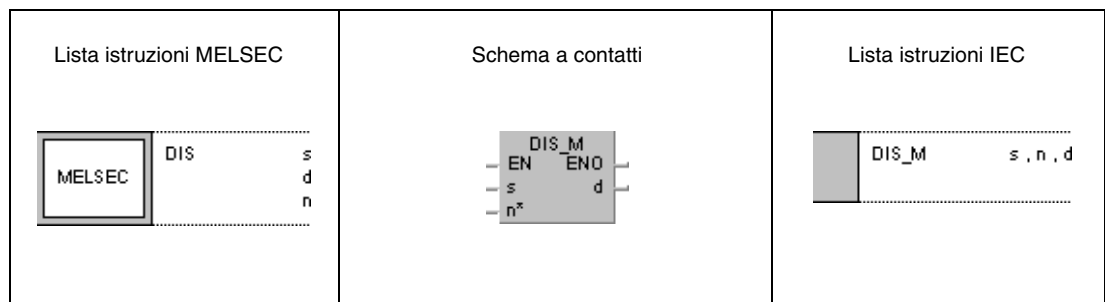
	Operandi utilizzabili														Definizione cifra	Numero di passi	Indice	Rapporto	Errore		
	Operandi a bit				Operandi a word (16-bit)						Costanti		Puntatore	Livello							
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1						Z	V
s							●	●	●	●	●	●	●	●	●	●	●				
d							●	●	●	●	●										
n																●	●				

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

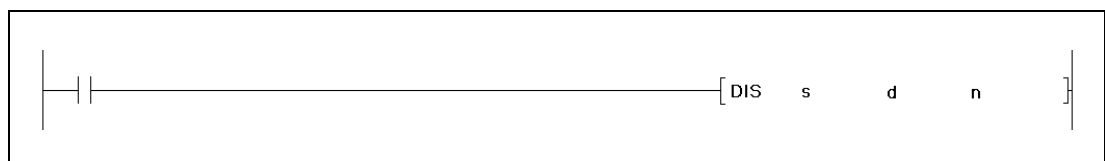
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	●	●	—	SM0	4
d	—	●	●	—	—	—	—	—			
n	●	●	●	●	●	●	●	●	—		

GX IEC Developer



GX Developer

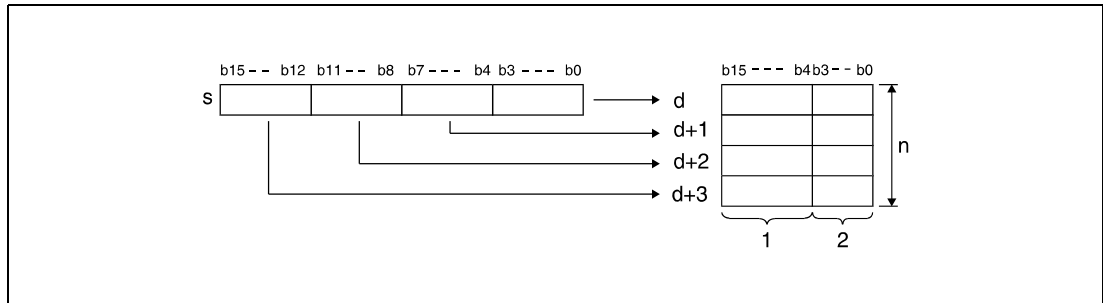


Variabili

Dati impostati	Significato	Tipo dati
s	Primo numero dell'operando che contiene i dati da disaggregare	BIN 16-bit
d	Primo numero dell'operando che memorizza il dato disaggregato.	
n	Numero di gruppi di 4 bit da disaggregare. Nessuna elaborazione se n = 0.	

Funzioni Disaggregazione dati a 16/16-bit**DIS Disaggregazione dati a 16-bit**

L'istruzione DIS disaggrega un valore dati a 16 bit in gruppi di 4 bit e memorizza questi gruppi in 4 operandi di destinazione in successione. Si devono specificare il valore da separare in s, il numero di gruppi da 4 bit in n, e l'indirizzo del primo operando di destinazione in d. I successivi gruppi di 4 bit vengono memorizzati a partire da d+n.



¹ Questi bit vengono impostati a 0.

I 12 bit più pesanti degli operandi di destinazione, iniziando dall'operando di indirizzo d, sono azzerati.

La variabile n può contenere un valore da 1 a 4 (da 4 a 16 bit).

Con n = 0 non viene eseguita nessuna operazione e gli operandi di uscita rimangono inalterati.

Errori di esecuzione

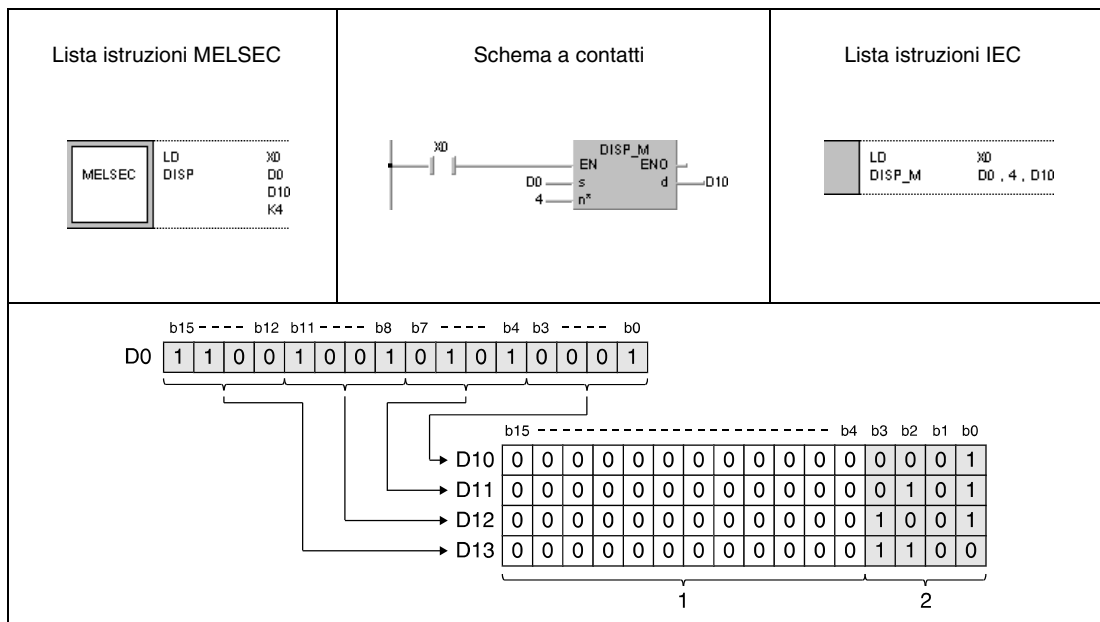
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- La variabile n non contiene un valore fra 0 e 4 (serie Q e System Q = codice di errore 4100).
- Il campo di indirizzi d definito da n supera il campo operando ammesso (serie Q e System Q = codice di errore 4101).

Esempio di programma

DISP

Con il fronte positivo di X0, il seguente programma disaggrega il valore del dato a 16 bit in D0 e memorizza i raggruppamenti di 4 bit da D10 a D13.



¹ Questi bit vengono impostati a 0.

² Campo di memorizzazione

7.5.7 UNI, UNIP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●☆	●

Operandi MELSEC A

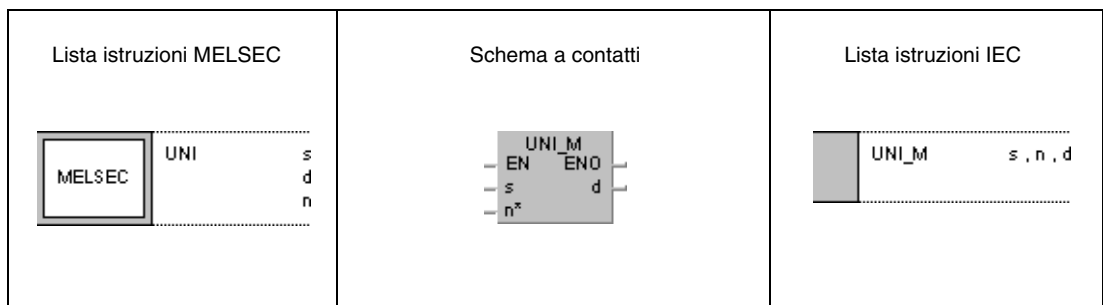
	Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Riporto	Errore						
	Operandi a bit							Operandi a word (16-bit)							Costanti						Puntatore	Livello				
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z						V	K	H (16#)	P	I	N
s							●	●	●	●												K1 ↓ K4	9 ↓ 1	●		●
d							●	●	●	●	●	●	●	●	●											
n																●	●									

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

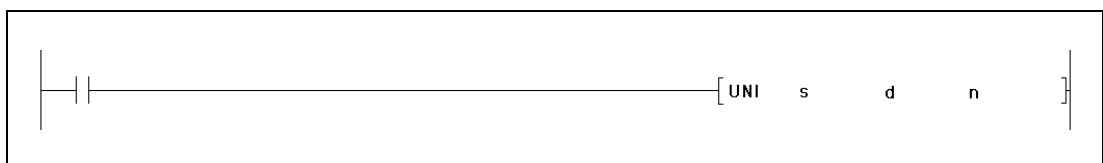
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	●	●	●	—	—	—	—	—	—	SM0	4
d	—	●	●	●	●	●	—	—			
n	●	●	●	●	●	●	●	—			

GX IEC Developer



GX Developer

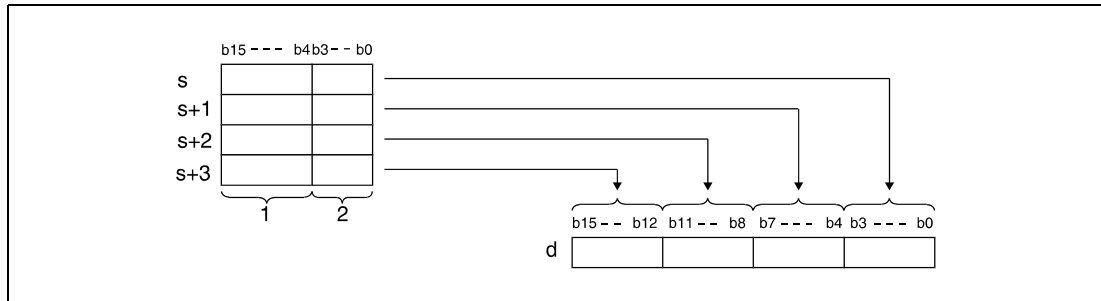


Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene i dati da aggregare.	BIN 16-bit
d	Primo numero dell'operando che memorizza il dato di partenza.	
n	Numero di gruppi di 4 bit da aggregare. Nessuna elaborazione se n = 0.	

Funzioni **Aggregazione dati a 16-bit****UNI** **Aggregazione dati a 16-bit**

L'istruzione UNI separa ciascuno dei 4 bit più leggeri di fino a quattro valori a 16 bit e unisce il loro stato in un valore a 16 bit. Per questa istruzione si devono specificare l'indirizzo del primo operando contenente i dati da aggregare in s, il numero di operandi in n, e l'indirizzo dell'operando di destinazione in d.



¹ Questi bit sono ignorati.

² Gruppi di 4 bit da memorizzare in d

I 4 bit più leggeri degli operandi sorgente, iniziando dall'operando di indirizzo d, sono azzerati.

La variabile n può contenere i valori da 1 a 4.

Con n = 0 non viene eseguita nessuna operazione e gli operandi di uscita rimangono inalterati.

Errori di esecuzione

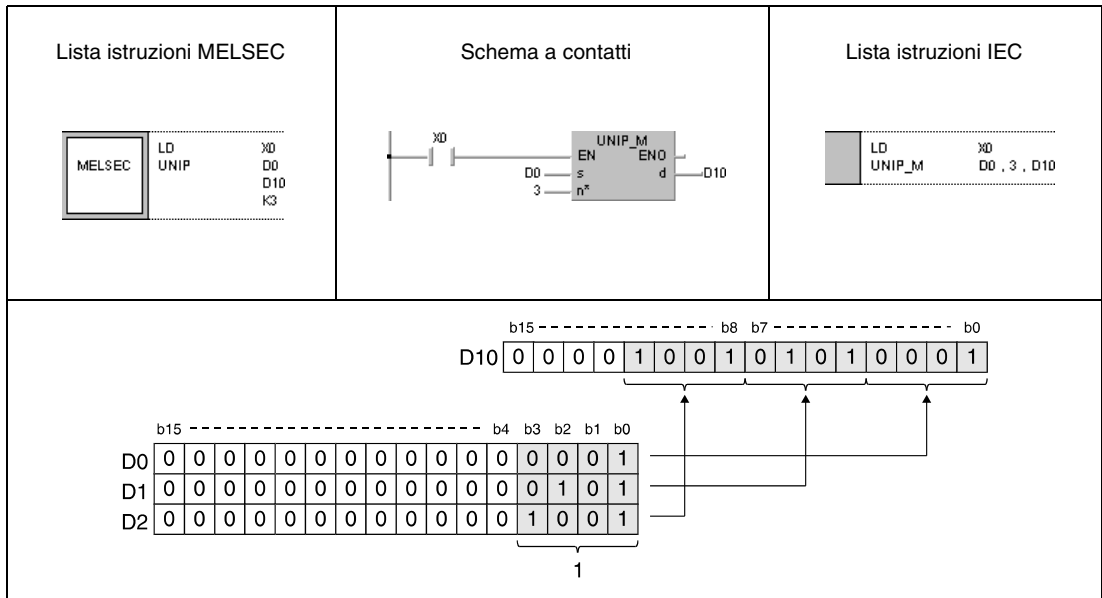
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- La variabile n non contiene un valore fra 0 e 4 (serie Q e System Q = codice di errore 4100).
- Il campo di indirizzi s definito da n supera il campo operando ammesso (serie Q e System Q = codice di errore 4101).

Esempio di programma

UNIP

Con il fronte di salita di X0, il programma seguente unisce i 4 bit più leggeri (da b0 a b3) di ciascuno dei registri dati da D0 a D2, raggruppandoli in un valore dati a 16 bit (4 bit più pesanti = 0) in D10.



¹ Gruppi di 4 bit da memorizzare in D10

7.5.8 NDIS, NDISP, NUNI, NUNIP


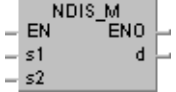
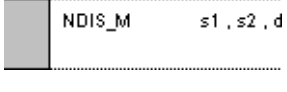
CPU

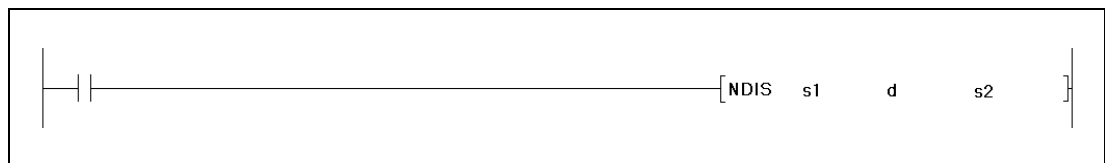
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi
MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	SM0	4	
d	—	●	●	—	—	—	—	—			
s2	—	●	●	—	—	—	—	—			

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
 <p>MELSEC NDIS s1 d s2</p>	 <p>NDIS_M — EN — ENO — — s1 — d — — s2 —</p>	 <p>NDIS_M s1, s2, d</p>

GX
Developer

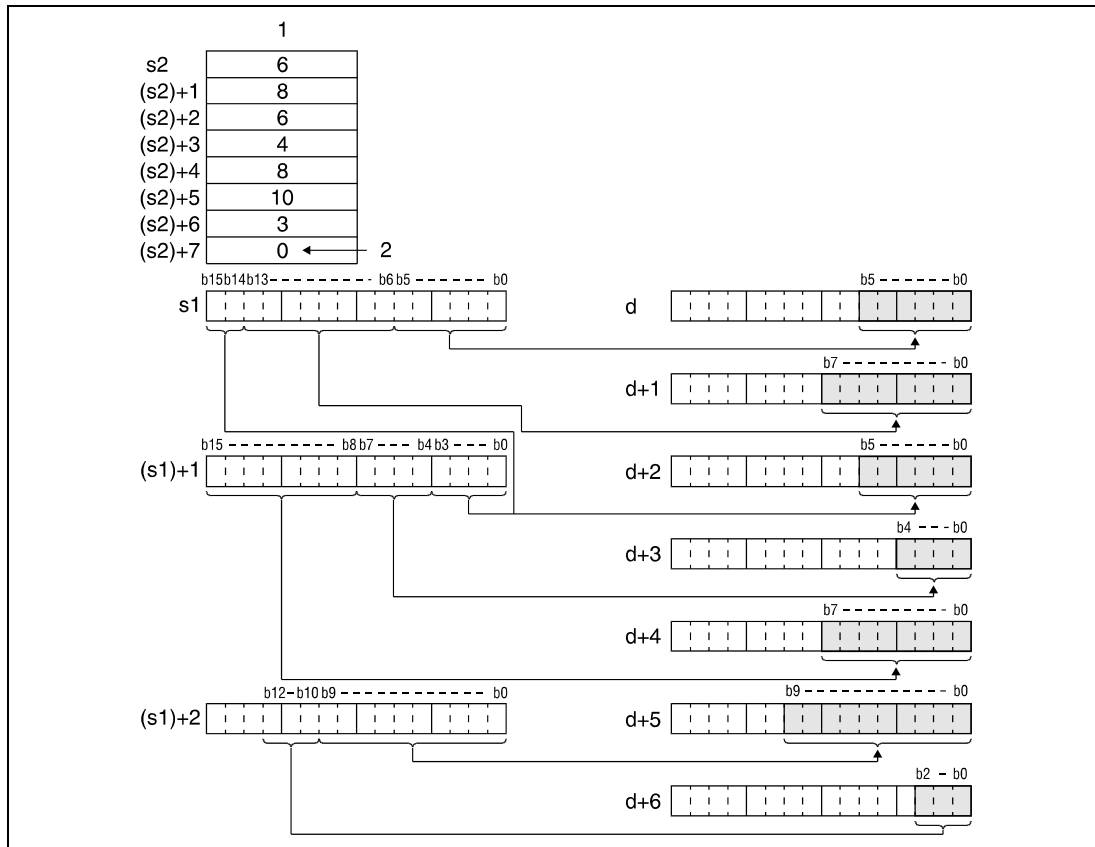
Variabili

Dati impostati	Significato	Tipo dati
s1	Indirizzo del primo operando che contiene i dati da disaggregare/aggiungere.	BIN 16-bit
d	Primo numero dell'operando che memorizza il dato disaggregato/aggiunto.	
s2	Numero di bit da disaggregare/aggiungere in gruppi da 4 bit.	

Funzioni Disaggregazione o aggregazione di dati in raggruppamenti di bit qualsiasi

NDIS Disaggregazione dati

L'istruzione NDIS separa i dati negli operandi a partire da s1 in raggruppamenti a bit in cui il numero di bit è indicato da s2. I gruppi di bit separati vengono memorizzati in sequenza a partire dall'operando indicato da d.

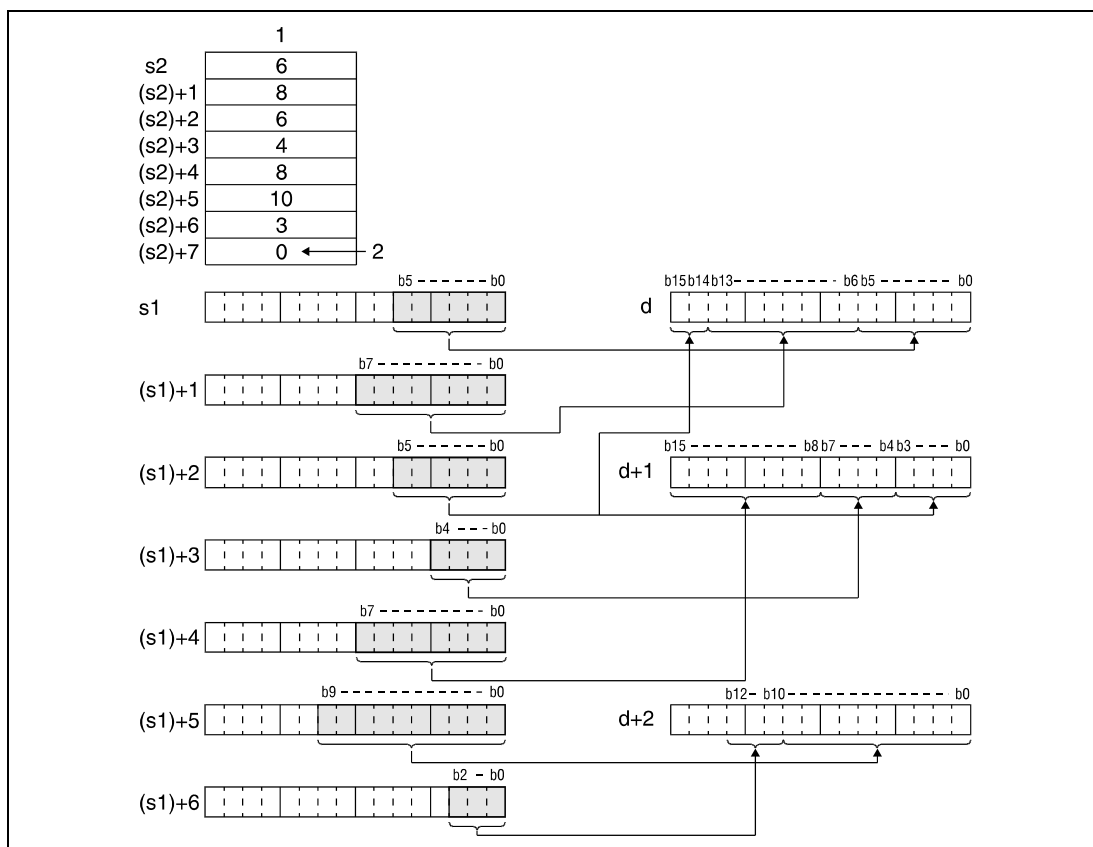


¹ Dimensione dei gruppi di bit
² Il valore 0 indica fine elaborazione

La dimensione dei gruppi di bit indicata da s2 può essere impostata nel campo da 1 e 16.
 I valori in s2 vengono elaborati dal primo operando il cui indirizzo è definito in s2, fino all'operando che contiene il valore 0.

NUNI Aggregazione dati

L'istruzione NUNI separa dei raggruppamenti di bit di dimensione specificata da s2 a partire dagli operandi specificati da s1, ed unisce questi gruppi di bit in un valore dati. I gruppi di bit separati vengono memorizzati in sequenza a partire dall'operando indicato da d.



¹ Dimensione dei gruppi di bit

² Il valore 0 indica fine elaborazione

La dimensione dei gruppi di bit indicata da s2 può essere impostata nel campo da 1 e 16.

I valori in s2 vengono elaborati dal primo operando il cui indirizzo è definito in s2, fino all'operando che contiene il valore 0.

Errori di esecuzione

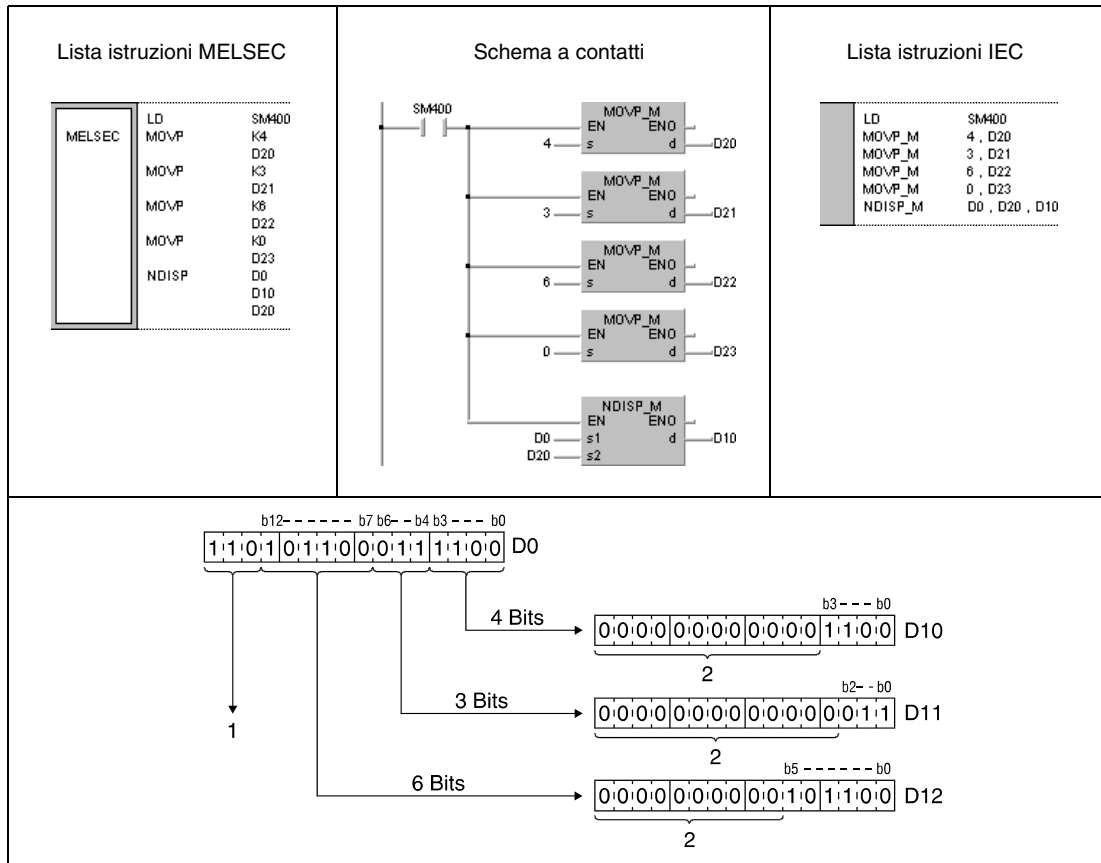
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- I gruppi di bit di dimensione specificata da s2, superano il campo operando ammesso sugli operandi s1 o d (codice di errore 4101).
- La dimensione dei gruppi di bit indicata da s2 supera il campo valido da 1 a 16 bit. (codice di errore 4100).

Programma di esempio 1

NDISP

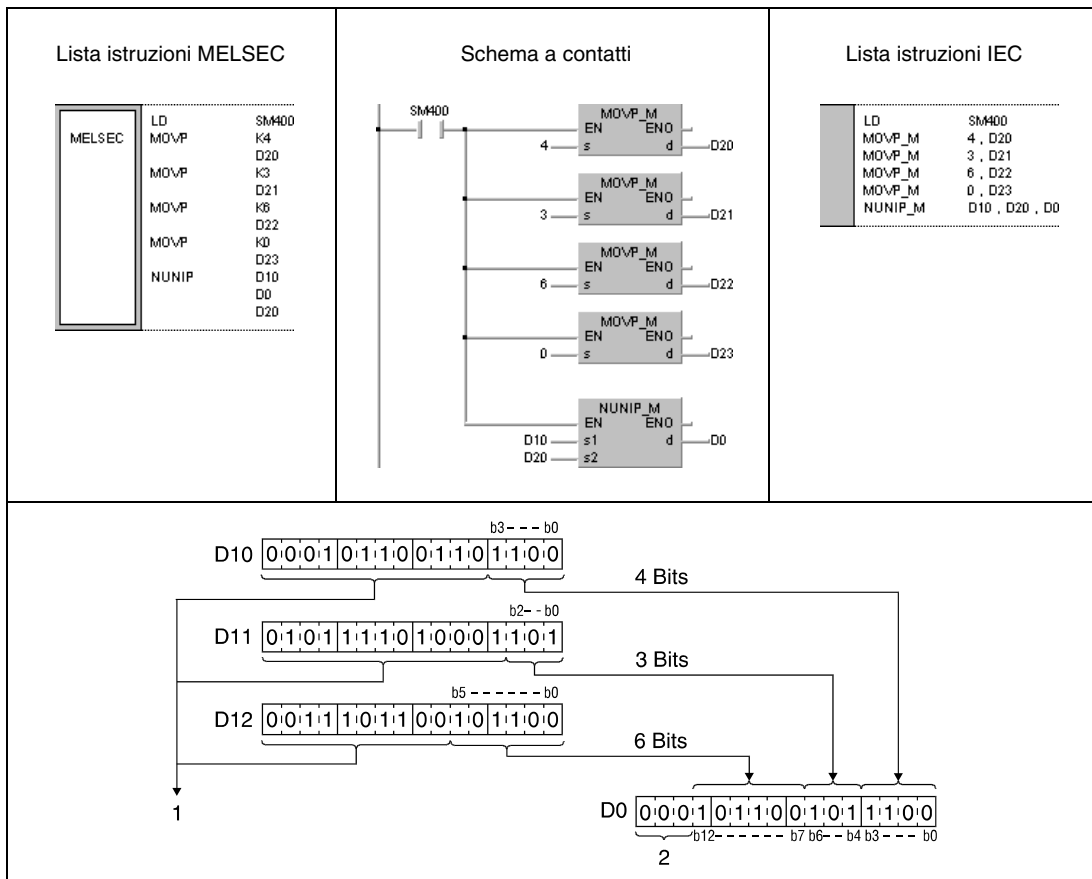
Con il fronte di salita di SM400, il programma che segue separa i gruppi di bit b0 - b3 (4), b4 - b6 (3), e b7 - b12 (6) da D0 e memorizza ciascun singolo gruppo di bit nei bit b0 - b3 da D10 a D12. I valori in parentesi indicano la dimensione dei gruppi di bit da D20 a D22. D23 deve contenere il valore 0 (vedi Funzioni).



¹ Questi bit sono ignorati.
² Questi bit vengono impostati a 0.

Programma di esempio 2 NUNIP

Con il fronte di salita di SM400, il programma seguente separa i gruppi di bit b0 - b3 (4), b0 - b2 (3), e b0 - b5 (6) da D10 a D12 e memorizza in sequenza a partire da D0 iniziando dal gruppo b0 - b3. I valori in parentesi indicano la dimensione dei gruppi di bit da D20 a D22. D23 deve contenere il valore 0 (vedi Funzioni).



¹ Questi bit sono ignorati.
² Questi bit vengono impostati a 0.

7.5.9 WTOB, WTOBP, BTOW, BTOWP

CPU


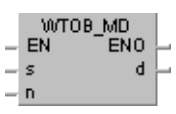
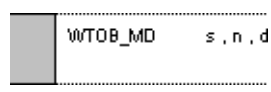
AnS	AnN	AnA, AnAS	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	●

¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

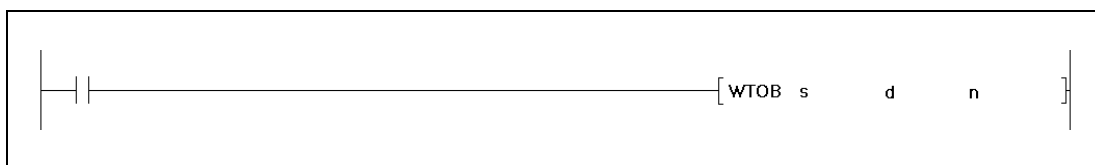
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	—	SM0	4	
d	—	●	●	—	—	—	—	—			
n	●	●	●	●	●	●	●	—			

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
---	---	--

GX Developer



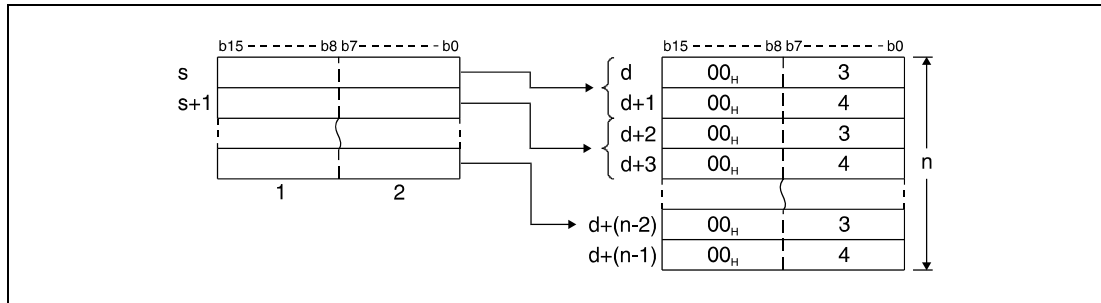
Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene i dati da disaggregare/aggiungere in byte.	BIN 16-bit
d	Primo numero dell'operando che memorizza il dato disaggregato/aggiunto in byte.	
n	Numero di byte da disaggregare/aggiungere.	

Funzioni Disaggregazione e aggregazione dati in byte

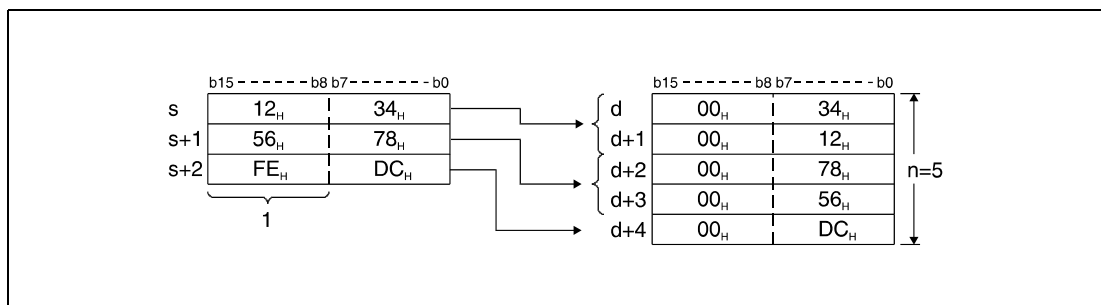
WTOB Disaggregazione dati

L'istruzione WTOB disaggrega un valore a 16 bit in byte e memorizza in sequenza gli stati risultanti negli operandi di destinazione. Per questa istruzione si devono specificare il valore da separare in s , il numero di byte in n , e il primo numero dell'operando di destinazione in d . I byte successivi sono memorizzati a partire a $d+n$. Solo i byte leggeri degli operandi di destinazione indicati da d vengono usati.



- ¹ Byte pesanti
- ² Byte leggeri
- ³ Dati corrispondenti ai byte leggeri
- ⁴ Dati corrispondenti ai byte pesanti

Ad esempio, se $n = 5$, vengono disaggregati 5 byte a partire al dispositivo indicato (da s a $s+2$), e memorizzati in sequenza nei byte leggeri degli operandi da d a $d+4$.

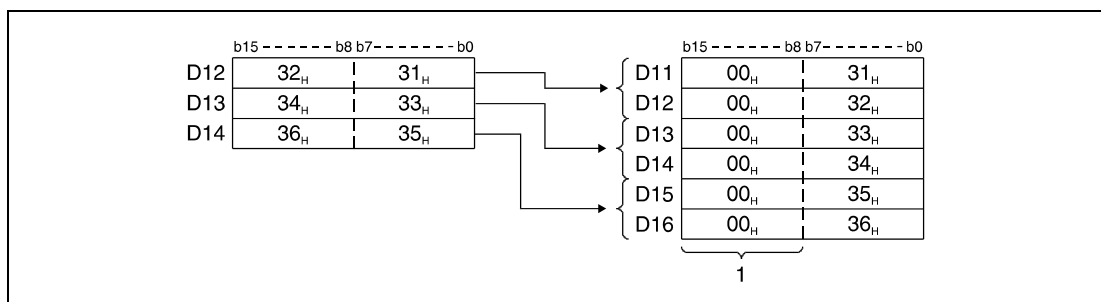


- ¹ Questi byte sono ignorati.

Il numero di byte indicato da n determina automaticamente il campo di dati a 16 bit indicato da s e la dimensione degli operandi di destinazione indicati da d .

Se $n = 0$ l'istruzione non viene eseguita e gli operandi contenuti negli indirizzi specificati non vengono modificati.

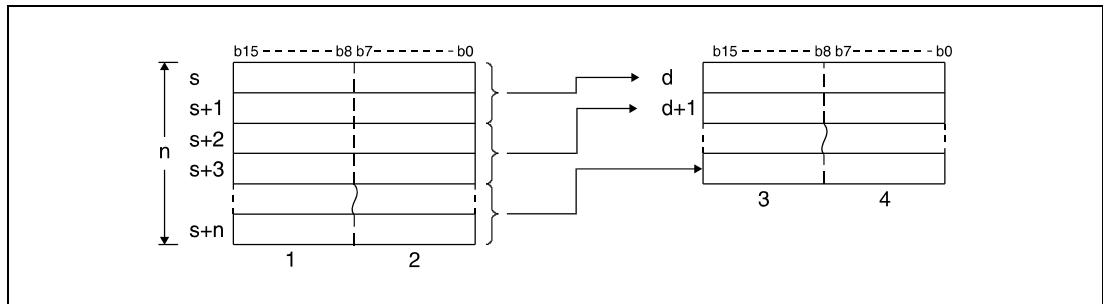
I byte pesanti degli operandi di destinazione indicati da d sono impostati con il valore "00H".



- ¹ Questi byte sono impostati con "00H"

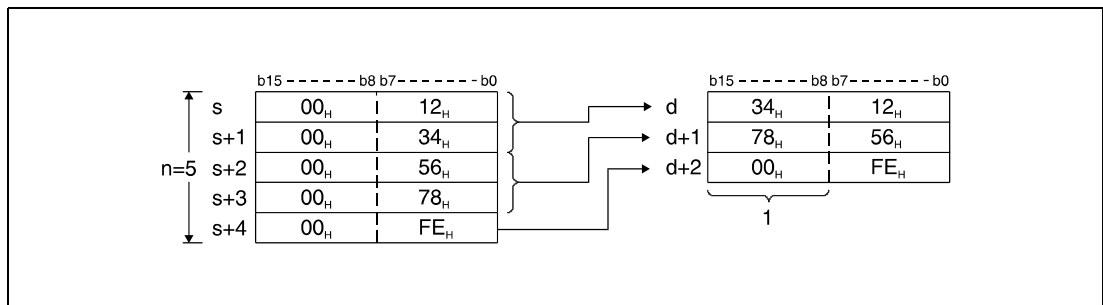
BTOW Aggregazione dati

L'istruzione separa i byte leggeri dei dati a 16 bit e memorizza il loro stato in dati a 16 bit. Occorre specificare il numero iniziale dei valori da unire in s, il numero di byte n, e l'operando di destinazione in d.



- ¹ Questi byte sono ignorati.
- ² Dati dei byte da 1 a n
- ³ Dati dei byte 2, 4, n
- ⁴ Dati dei byte 1, 3, (n-1)

Ad esempio, se n = 5, vengono estratti 5 byte a partire al dispositivo indicato (da s a s+4), e memorizzati in sequenza negli operandi da d a d+2.



- ¹ Questo byte è impostato con "00H"

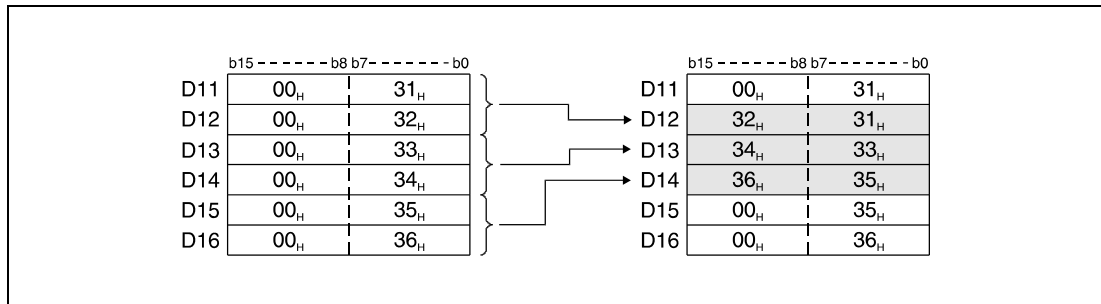
Il numero di byte indicato da n determina automaticamente il campo di dati a byte indicato da s e la dimensione degli operandi di destinazione indicati da d.

Se n = 0 l'istruzione non viene eseguita e gli operandi contenuti negli indirizzi specificati non vengono modificati.

I byte pesanti degli operandi indicati da s vengono ignorati.

L'operazione viene eseguita correttamente anche se i campi degli operandi da s a s+n e da d a d+n sono sovrapposti.

La figura che segue mostra un caso in cui i byte leggeri vengono estratti da D11 a D16 e memorizzati in sequenza da D12 a D14.



Errori di esecuzione

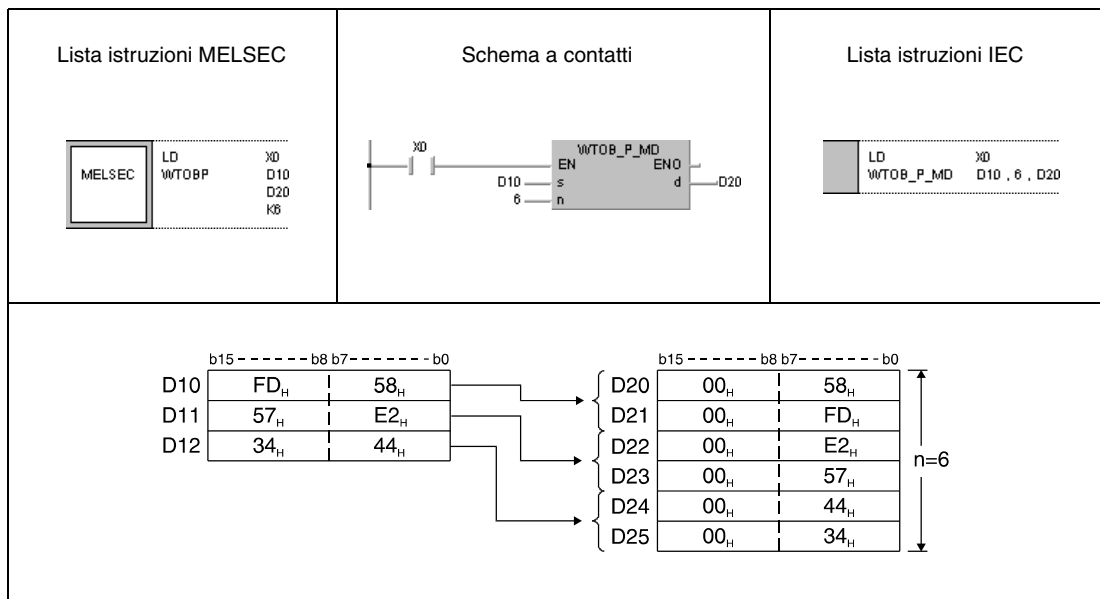
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il numero di byte indicato da n, memorizzati a partire dall'operando indicato da s, supera il campo operando ammesso (serie Q e System Q = codice di errore 4101).
- Il numero di byte indicato da n, memorizzati a partire dall'operando indicato da d, supera il campo operando ammesso (serie Q e System Q = codice di errore 4101).

Programma di esempio 1

WTOBP

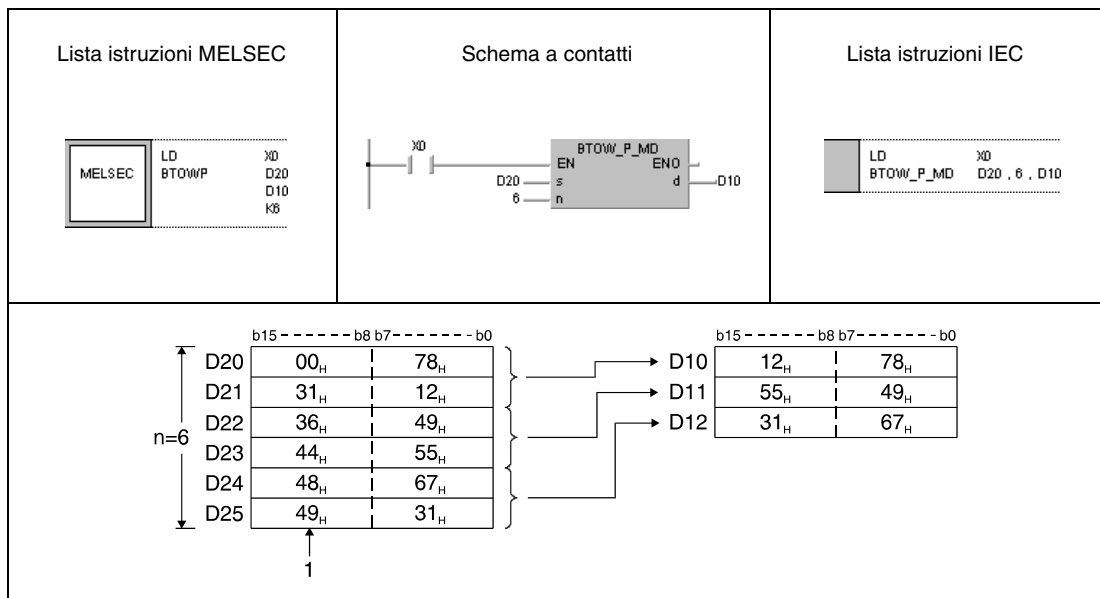
Con il fronte di salita di X0, il programma che segue estrae in sequenza 6 byte da D10 a D12 e memorizza questi byte nei byte leggeri da D20 a D25.



Programma di esempio 2

BTOWP

Con il fronte di salita di X0, il programma che segue estrae i byte leggeri dai registri da D20 a D25 ed unisce in sequenza questi byte da D10 a D12.



¹ Questi byte sono ignorati.

7.5.10 MAX, MAXP, DMAX, DMAXP


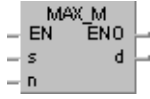
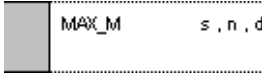
CPU

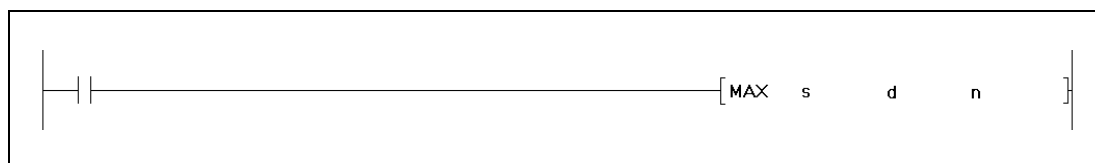
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi
MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	—	SM0	4	
d	—	●	●	—	—	—	—	—			
n	●	●	●	●	●	●	●	—			

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
		

GX
Developer

Variabili

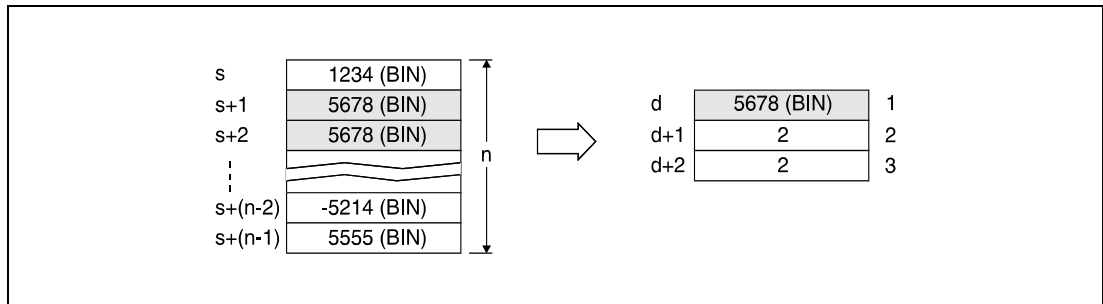
Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che memorizza i dati di cui si vuole cercare il valore massimo.	BIN 16/32-bit
d	Indirizzo del primo operando che memorizza il risultato della ricerca.	
n	Numero di blocchi dati su cui deve essere eseguita la ricerca.	BIN 16-bit

Funzioni Ricerca valore massimo in dati a 16/32 bit

MAX Ricerca valore massimo in dati a 16 bit

L'istruzione MAX ricerca il valore massimo in blocchi dati da 16 bit. Il numero di blocchi dati in cui effettuare la ricerca è indicato da n. Il valore più grande trovato da s a s+(n-1) viene memorizzato in d.

La prima posizione da s a s+(n-1) in cui si trova il valore massimo viene contata iniziando da s = 1 e memorizzata in d+1. Il numero di valori massimi identici viene memorizzato in d+2.

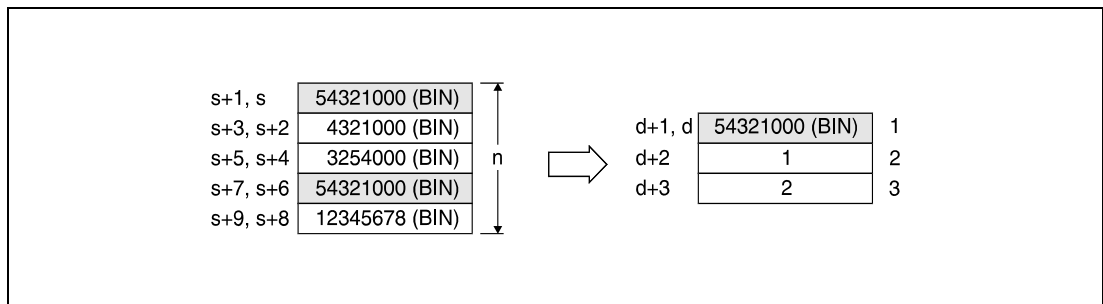


- ¹ Valore massimo trovato
- ² Prima posizione in cui si è trovato il valore
- ³ Numero di valori massimi identici

DMAX Ricerca valore massimo in dati a 32 bit

L'istruzione DMAX ricerca il valore massimo in blocchi dati da 32 bit. Il numero di blocchi dati in cui effettuare la ricerca è indicato da n. Il valore più grande trovato da s a s+(n-1) viene memorizzato in d.

La prima posizione da s a s+(n-1) in cui si trova il valore massimo viene contata iniziando da s = 1 e memorizzata in d+2. Il numero di valori massimi identici viene memorizzato in d+3.



- ¹ Valore massimo trovato
- ² Prima posizione in cui si è trovato il valore
- ³ Numero di valori massimi identici

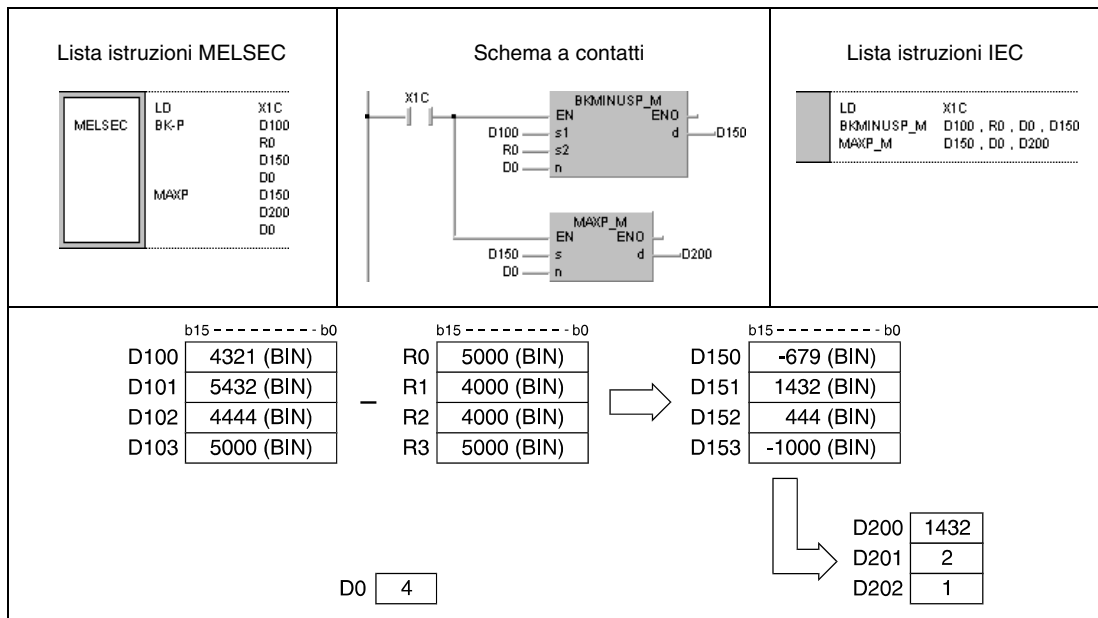
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

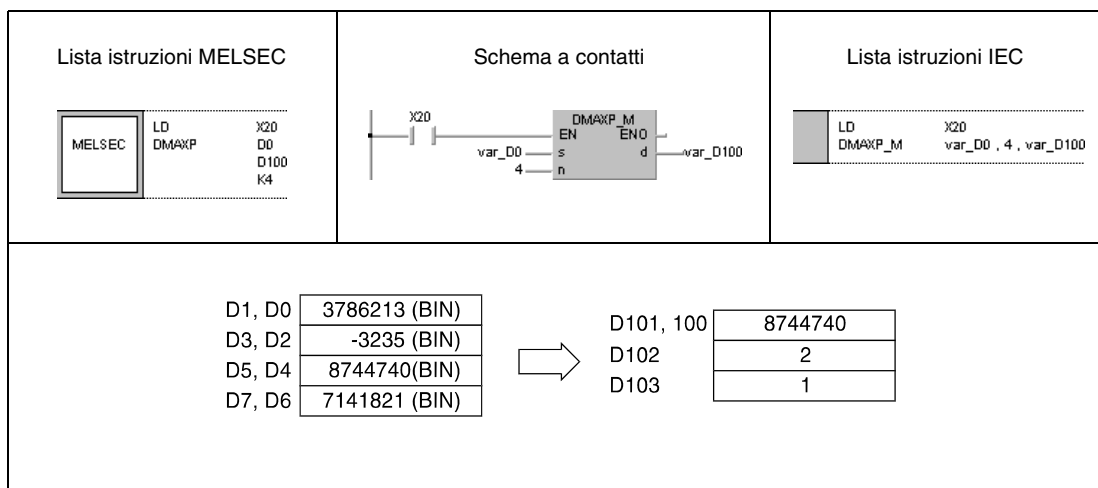
- Il numero di blocchi dati indicato da n, memorizzato negli operandi a partire da s, supera il campo operando ammesso (codice di errore 4101).

Programma di esempio 1 MAXP

Con il fronte di salita di X1C, il programma che segue sottrae i dati da R0 a R3 dai dati da D100 a D103 e memorizza il risultato da D150 a D153. Il numero di blocchi a 16 bit da elaborare (4) è contenuto in D0. Nel passo successivo, come sul fronte di salita di X1C, si ricerca il valore massimo nei registri da D150 a D153. Il valore trovato viene memorizzato in D200, la sua posizione in D201 ed il numero di valori massimi identici in D202.


Programma di esempio 2 DMAXP

Con il fronte positivo di X20, il programma che segue ricerca il valore massimo nei dati a 32 bit D100 e D101. La posizione del valore viene memorizzata in D102, il numero di valori massimi identici in D103.

**NOTA**

Il programma di esempio 2 non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.5.11 MIN, MINP, DMIN, DMINP

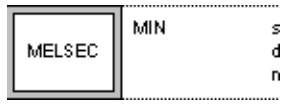
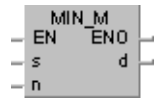
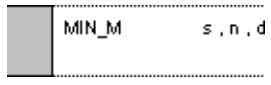
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

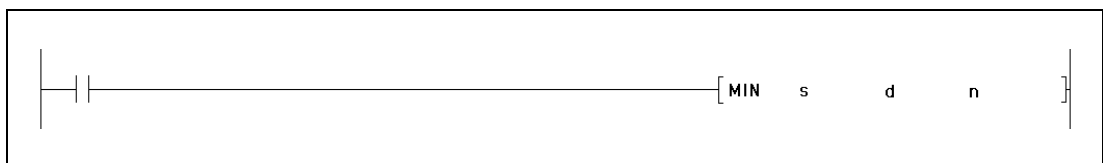
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	—	SM0	4	
d	—	●	●	—	—	—	—	—			
n	●	●	●	●	●	●	●	—			

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
--	--	---

GX Developer



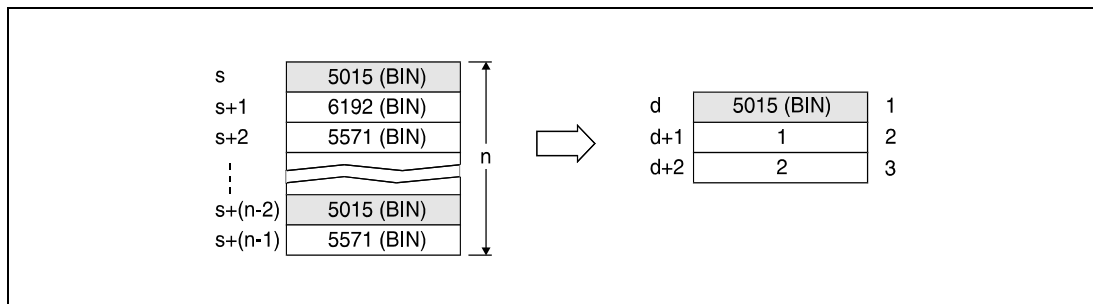
Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che memorizza i dati di cui si vuole cercare il valore minimo.	BIN 16/32-bit
d	Indirizzo del primo operando che memorizza il risultato della ricerca.	
n	Numero di blocchi dati su cui deve essere eseguita la ricerca.	BIN 16-bit

Funzioni Ricerca valore minimo in dati a 16/32 bit**MIN Ricerca valore minimo in dati a 16 bit**

L'istruzione MIN ricerca il valore minimo in blocchi dati da 16 bit. Il numero di blocchi dati in cui effettuare la ricerca è indicato da n. Il valore più piccolo trovato da s a s+(n-1) viene memorizzato in d.

La prima posizione da s a s+(n-1) in cui si trova il valore minimo viene contata iniziando da s = 1 e memorizzata in d+1. Il numero di valori minimi identici viene memorizzato in d+2.



¹ Valore minimo trovato

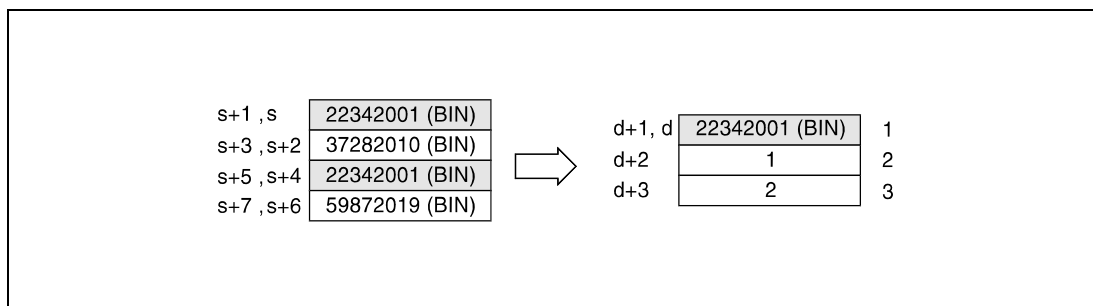
² Prima posizione in cui si è trovato il valore

³ Numero di valori minimi identici

DMIN Ricerca valore minimo in dati a 32 bit

L'istruzione DMIN ricerca il valore minimo in blocchi dati da 32 bit. Il numero di blocchi dati in cui effettuare la ricerca è indicato da n. Il valore più piccolo trovato da s a s+(n-1) viene memorizzato in d e d+1.

La prima posizione da s a s+(n-1) in cui si trova il valore minimo viene contata iniziando da s = 1 e memorizzata in d+2. Il numero di valori minimi identici viene memorizzato in d+3.



¹ Valore minimo trovato

² Prima posizione in cui si è trovato il valore

³ Numero di valori minimi identici

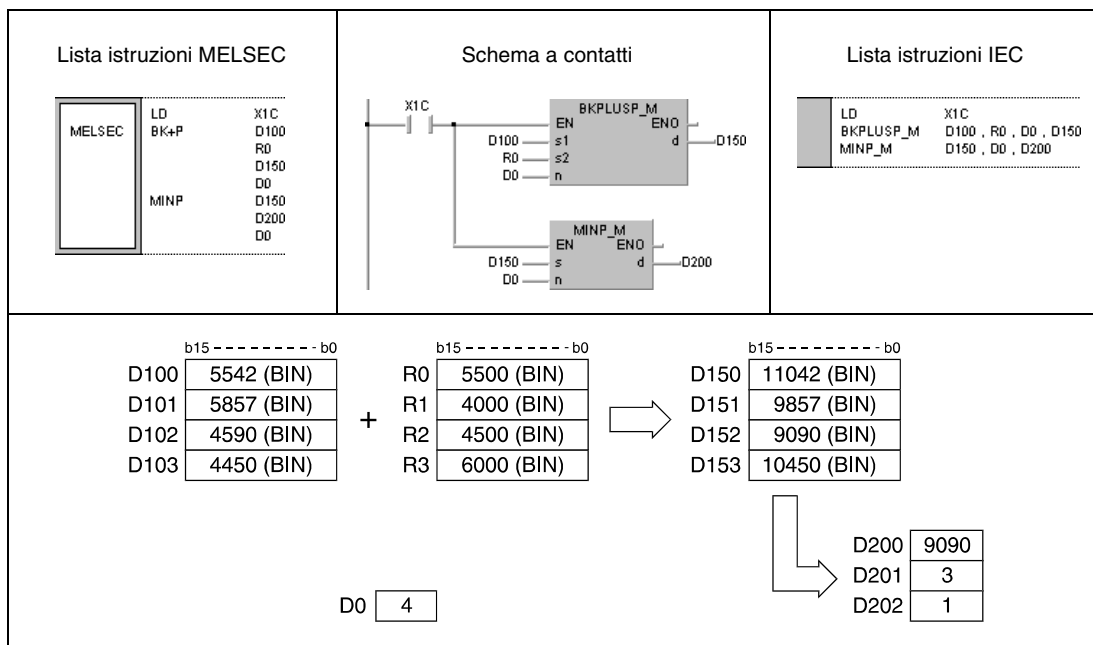
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il numero di blocchi dati indicato da n, memorizzato negli operandi a partire da s, supera il campo operando ammesso (codice di errore 4101).

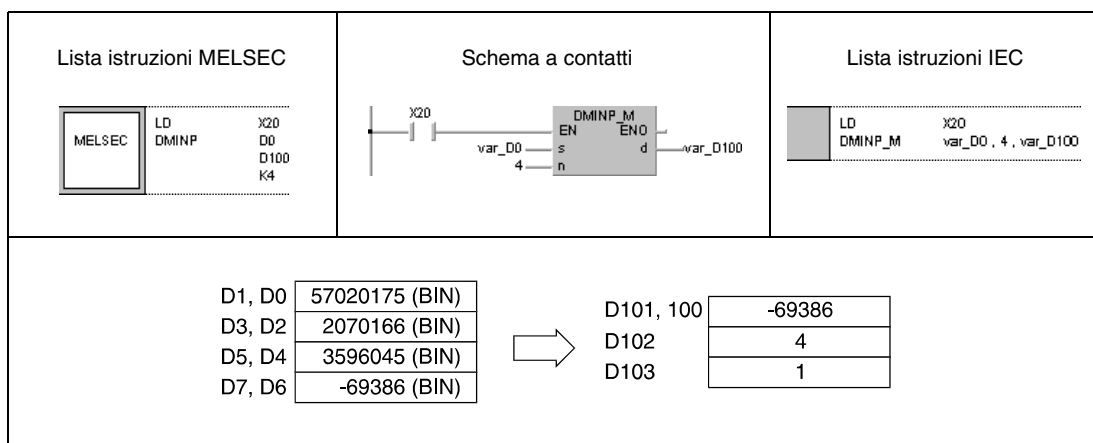
Programma di esempio 1 MINP

Con il fronte di salita di X1C, il programma che segue sottrae i dati da D100 a D103 dai dati da R0 a R3 e memorizza il risultato da D150 a D153. Il numero di blocchi a 16 bit da elaborare (4) è contenuto in D0. Nel passo successivo, come sul fronte di salita di X1C, si ricerca il valore minimo nei registri da D150 a D153. Il valore trovato viene memorizzato in D200, la sua posizione in D201 ed il numero di valori minimi identici in D202.



Programma di esempio 2 DMINP

Con il fronte positivo di X20, il programma che segue ricerca il valore minimo nei dati a 32 bit da D0 a D7 e memorizza il valore in D100 e D101. La posizione del valore viene memorizzata in D102, il numero di valori minimi identici in D103.



NOTA

Il programma di esempio 2 non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.5.12 SORT, SORTP, DSORT, DSORTP

CPU

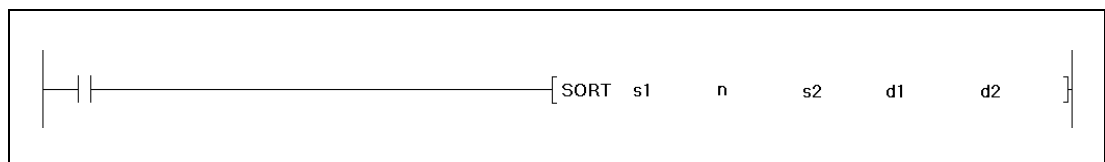
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi
MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	SM0	6	
n	●	●	●	—	—	—	—	—			
s2	●	●	●	●	●	●	●	—			
d1	●	—	—	●	●	●	●	—			
d2	—	●	●	—	—	—	—	—			

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC												
<div style="border: 1px solid black; padding: 5px; width: fit-content;"> MELSEC </div> <table style="margin-left: 20px;"> <tr><td>SORT</td><td>s1</td></tr> <tr><td></td><td>n</td></tr> <tr><td></td><td>s2</td></tr> <tr><td></td><td>d1</td></tr> <tr><td></td><td>d2</td></tr> </table>	SORT	s1		n		s2		d1		d2		<table style="margin-left: 20px;"> <tr><td>SORT_M</td><td>s1, n, s2, d1, d2</td></tr> </table>	SORT_M	s1, n, s2, d1, d2
SORT	s1													
	n													
	s2													
	d1													
	d2													
SORT_M	s1, n, s2, d1, d2													

GX
Developer

Variabili

Dati impostati	Significato	Tipo dati
s1	Indirizzo del primo operando che contiene i dati da ordinare.	BIN 16/32-bit
n	Numero di blocchi dati da ordinare	BIN 16-bit
s2	Numero di blocchi dati da confrontare per ciascuna operazione di ordinamento.	BIN 16-bit
d1	Numero del bit da attivare al termine dell'operazione di ordinamento.	Bit
d2	Solo per uso di sistema	BIN 16-bit

Funzioni Ordina dati a 16-/32-bit

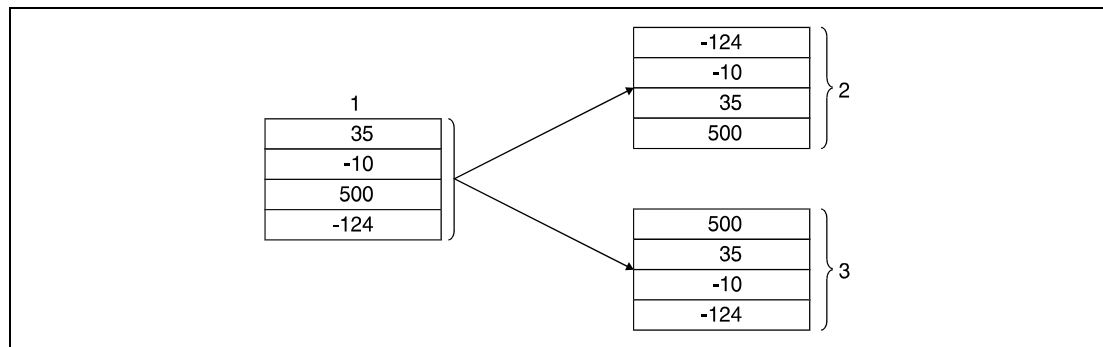
SORT Ordina dati a 16 bit

L'istruzione SORT ordina i dati a 16 bit specificati da s1 in senso crescente o decrescente. Il numero di dati da ordinare è indicato in n.

Il senso di ordinamento viene definito tramite il relé speciale SM703:

SM703 OFF: Ordine ascendente

SM703 ON: Ordine discendente



¹ Dati da ordinare

² Dati ordinati in senso ascendente (SM703 = OFF)

³ Dati ordinati in senso discendente (SM703 = ON)

Per l'esecuzione dell'istruzione SORT sono necessari diversi cicli di scansione. Il numero di scansioni richieste può essere calcolato dividendo il numero massimo di scansioni per il numero a 16 bit contenuto in s2, che indica i blocchi confrontati per ciascuna scansione (valori decimali arrotondati per eccesso). Aumentando il numero dei dati a 16 bit indicati in s2, si ottiene una riduzione del numero di scansioni necessarie per l'ordinamento, ma aumenta il tempo di elaborazione necessario per una scansione.

Il numero richiesto di cicli di scansione per l'esecuzione dell'operazione di ordinamento può essere calcolato con l'equazione:

$$\text{Numero cicli di scansione richiesti} = ((n) \times (n-1)) / (2 \times (s2))$$

Ad esempio, se $n = 10$ e $s2 = 1$, sono necessari 45 cicli di scansione per terminare l'operazione di ordinamento.

Se $n = 10$ e $s2 = 2$, il risultato è 22,5, arrotondato a 23 cicli di scansione necessari.

Il bit specificato da d1 viene azzerato durante l'operazione di ordinamento e viene impostato a 1 al termine dell'operazione. Il bit rimane attivo e deve essere azzerato esplicitamente dal programma.

Gli operandi indicati da d2 e (d2)+1 vengono utilizzati internamente dal sistema durante l'ordinamento. Questi operandi non devono essere modificati dal programma.

Se il valore in n viene modificato durante l'esecuzione, l'operazione viene eseguita sul numero di dati a 16 bit attualmente impostato.

Disattivando la condizione di esecuzione, l'operazione viene terminata. Riattivando la condizione di esecuzione l'operazione di ordinamento viene riavviata.

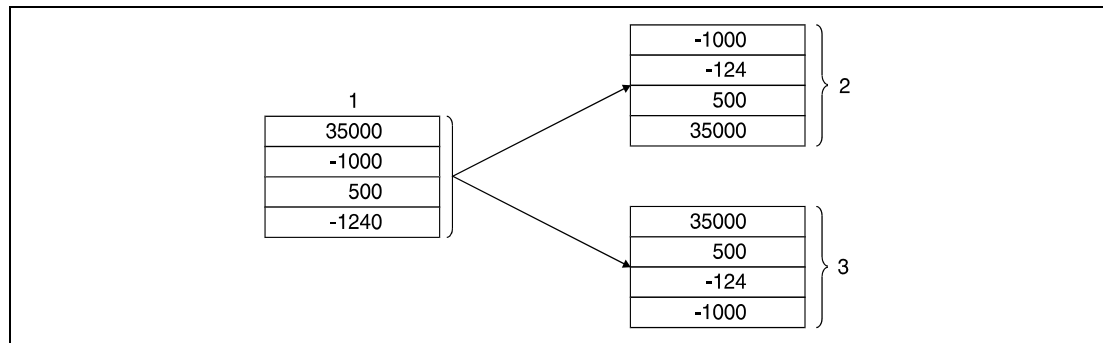
DSORT Ordina dati a 32 bit

L'istruzione DSORT ordina i dati a 32 bit specificati da s1 in senso crescente o decrescente. Il numero di dati da ordinare è indicato in n.

Il senso di ordinamento viene definito tramite il relé speciale SM703:

SM703 OFF: Ordine ascendente

SM703 ON: Ordine discendente



¹ Dati da ordinare

² Dati ordinati in senso ascendente (SM703 = OFF)

³ Dati ordinati in senso discendente (SM703 = ON)

Per l'esecuzione dell'istruzione DSORT sono necessari diversi cicli di scansione. Il numero di scansioni richieste può essere calcolato dividendo il numero massimo di scansioni per il numero a 32 bit contenuto in s2, che indica i blocchi confrontati per ciascuna scansione (valori decimali arrotondati per eccesso). Aumentando il numero dei dati a 32 bit indicati in s2, si ottiene una riduzione del numero di scansioni necessarie per l'ordinamento, ma aumenta il tempo di elaborazione necessario per una scansione.

Il numero richiesto di cicli di scansione per l'esecuzione dell'operazione di ordinamento può essere calcolato con l'equazione:

$$\text{Numero cicli di scansione richiesti} = ((n) \times (n-1)) / (2 \times (s2))$$

Ad esempio, se $n = 10$ e $s2 = 1$, sono necessari 45 cicli di scansione per terminare l'operazione di ordinamento.

Se $n = 10$ e $s2 = 2$, il risultato è 22,5, arrotondato a 23 cicli di scansione necessari.

Il bit specificato da d1 viene azzerato durante l'operazione di ordinamento e viene impostato a 1 al termine dell'operazione. Il bit rimane attivo e deve essere azzerato esplicitamente dal programma.

Gli operandi indicati da d2 e (d2)+1 vengono utilizzati internamente dal sistema durante l'ordinamento. Questi operandi non devono essere modificati dal programma.

Se il valore in n viene modificato durante l'esecuzione, l'operazione viene eseguita sul numero di dati a 32 bit attualmente impostato.

Disattivando la condizione di esecuzione, l'operazione viene terminata. Riattivando la condizione di esecuzione l'operazione di ordinamento viene riavviata.

Errori di esecuzione

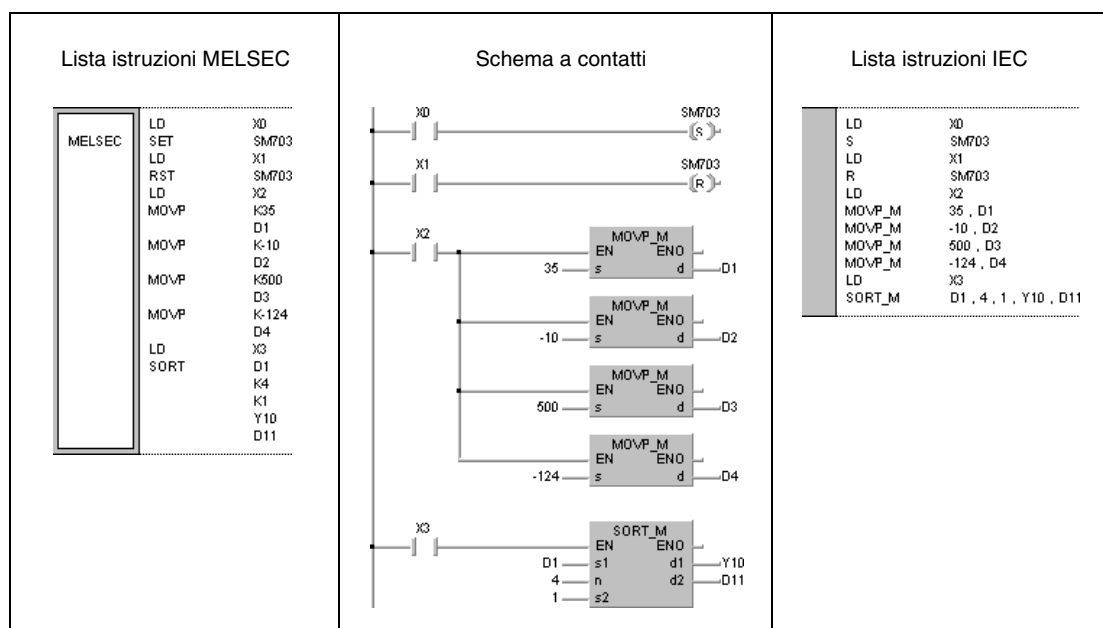
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il campo specificato da n o 2 x n per l'operando indicato da s1 supera il campo operando ammesso (codice di errore 4101).
- Il valore specificato da s2 è uguale o minore di zero (codice di errore 4100).
- Il valore in d2 è superiore a quello in n (codice di errore 4101).
- Il valore in (d2)+1 è superiore a quello in d2 (codice di errore 4101).

Esempio di programma

SORT

Mentre X3 è attivo, il programma seguente ordina i dati a 16 bit da D1 a D4. In un primo passo, sul fronte positivo di X2, vengono scritti i valori 35, -10, 500, e -124 nei registri da D1 a D4. Poi viene avviato l'ordinamento. Il senso di ordinamento viene definito da X0 (attiva SM703) e X1 (disattiva SM703). Al termine dell'operazione di ordinamento l'uscita Y10 viene attivata.



7.5.13 WSUM, WSUMP

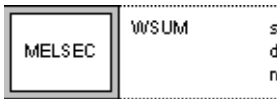
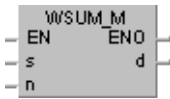
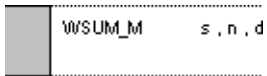
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

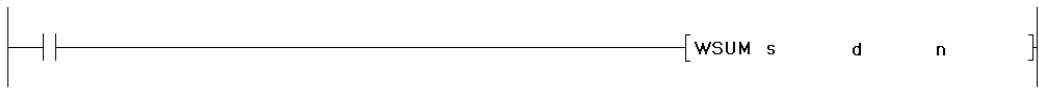
Operandi
MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	—	SM0	4	
d	●	●	●	●	●	●	●	—			
n	●	●	●	●	●	●	●	—			

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
		

GX
Developer


--

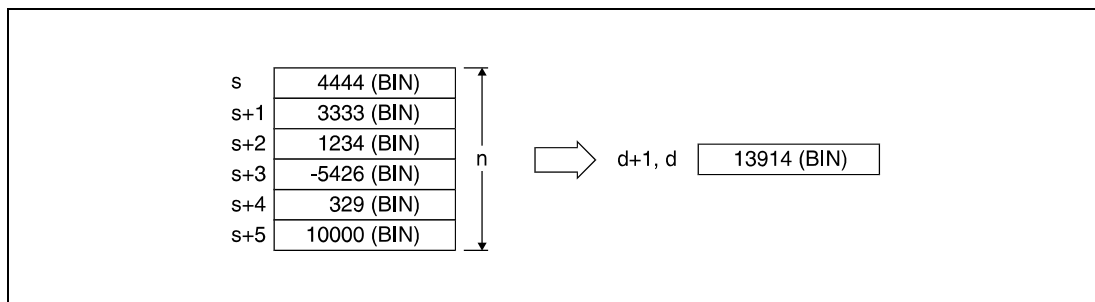
Variabili

Dati impostati	Significato	Tipo dati
s	Primo numero dell'operando che contiene i dati da sommare.	BIN 16-bit
d	Primo numero dell'operando che memorizza il risultato.	BIN 32-bit
n	Numero di blocchi dati da sommare	BIN 16-bit

Funzioni **Calcolo del totale di blocchi dati BIN a 16 bit**

WSUM **Calcolo di totali**

L'istruzione WSUM calcola il totale dei blocchi dati a 16 bit contenuti negli operandi indicati da s. Il numero di blocchi dati da sommare è indicato da n. Il risultato viene memorizzato nell'operando indicato da d.



Errori di esecuzione

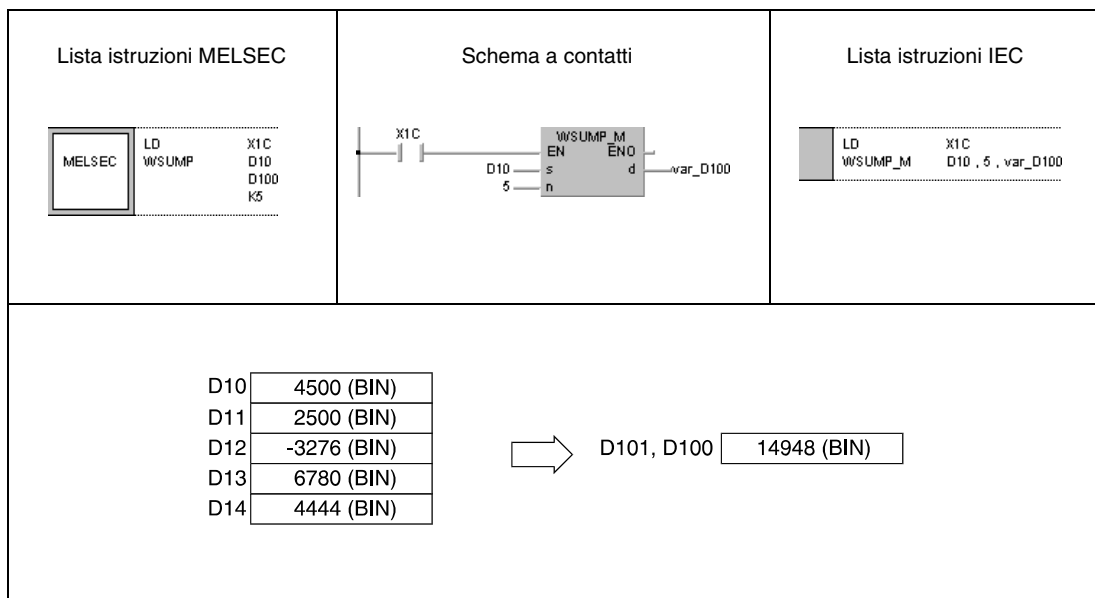
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il campo specificato da n per l'operando indicato da s supera il campo operando ammesso (codice di errore 4101).

Esempio di programma

WSUMP

Con il fronte positivo di X1C, il programma che segue somma blocchi dati a 16 bit da D10 a D14 e memorizza il risultato in D100 e D101.



7.5.14 DWSUM, DWSUMP


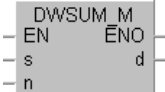

CPU

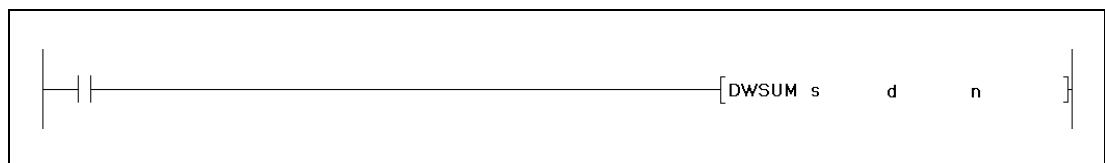
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi
MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	—	SM0	4	
d	●	●	●	—	—	—	—	—			
n	●	●	●	●	●	●	●	—			

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
 DWSUM s d n	 DWSUM_M — EN ENO — — s d — — n	 DWSUM_M s, n, d

GX
Developer

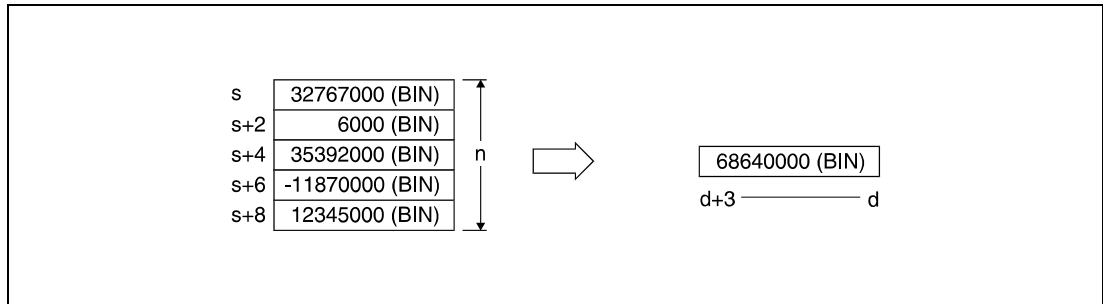
Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s	Primo numero dell'operando che contiene i dati da sommare.	BIN 32-bit	ANY32
d	Primo numero dell'operando che memorizza il risultato.	BIN 64-bit	Array [1..4] di ANY16
n	Numero di blocchi dati da sommare	BIN 16-bit	ANY16

Funzioni **Calcolo del totale di blocchi dati BIN a 32 bit**

DWSUM Calcolo di totali

L'istruzione DWSUM calcola il totale dei blocchi dati a 32 bit contenuti negli operandi indicati da s. Il numero di blocchi dati da sommare è indicato da n. Il risultato viene memorizzato nell'array [1..4] nell'operando indicato da d.



Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il campo specificato da n per l'operando indicato da s supera il campo operando ammesso (codice di errore 4101).

Esempio di programma

DWSUMP

Con il fronte positivo di X20, il programma che segue somma blocchi dati a 32 bit da D100 a D107 e memorizza il risultato da D10 a D13.

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC													
<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;">MELSEC</div> <table border="1" style="margin: 10px auto; border-collapse: collapse;"> <tr><td>LD</td><td>X20</td></tr> <tr><td>DWSUMP</td><td>D100 D10 K5</td></tr> </table>	LD	X20	DWSUMP	D100 D10 K5		<table border="1" style="margin: 10px auto; border-collapse: collapse;"> <tr><td>LD</td><td>X20</td></tr> <tr><td>DWSUMP_M</td><td>var_D100 , 4 , var_D10</td></tr> </table>	LD	X20	DWSUMP_M	var_D100 , 4 , var_D10					
LD	X20														
DWSUMP	D100 D10 K5														
LD	X20														
DWSUMP_M	var_D100 , 4 , var_D10														
<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%; border-right: 1px solid black; padding-right: 10px;"> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>D101, D100</td><td>11245600 (BIN)</td></tr> <tr><td>D103, D102</td><td>27543200 (BIN)</td></tr> <tr><td>D105, D104</td><td>558800 (BIN)</td></tr> <tr><td>D107, D106</td><td>-15675000 (BIN)</td></tr> </table> </td> <td style="width: 20%; text-align: center; vertical-align: middle;">➔</td> <td style="width: 50%; padding-left: 10px;"> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>D13 — D10</td><td>23672600 (BIN)</td></tr> </table> </td> </tr> </table>			<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>D101, D100</td><td>11245600 (BIN)</td></tr> <tr><td>D103, D102</td><td>27543200 (BIN)</td></tr> <tr><td>D105, D104</td><td>558800 (BIN)</td></tr> <tr><td>D107, D106</td><td>-15675000 (BIN)</td></tr> </table>	D101, D100	11245600 (BIN)	D103, D102	27543200 (BIN)	D105, D104	558800 (BIN)	D107, D106	-15675000 (BIN)	➔	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>D13 — D10</td><td>23672600 (BIN)</td></tr> </table>	D13 — D10	23672600 (BIN)
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>D101, D100</td><td>11245600 (BIN)</td></tr> <tr><td>D103, D102</td><td>27543200 (BIN)</td></tr> <tr><td>D105, D104</td><td>558800 (BIN)</td></tr> <tr><td>D107, D106</td><td>-15675000 (BIN)</td></tr> </table>	D101, D100	11245600 (BIN)	D103, D102	27543200 (BIN)	D105, D104	558800 (BIN)	D107, D106	-15675000 (BIN)	➔	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>D13 — D10</td><td>23672600 (BIN)</td></tr> </table>	D13 — D10	23672600 (BIN)			
D101, D100	11245600 (BIN)														
D103, D102	27543200 (BIN)														
D105, D104	558800 (BIN)														
D107, D106	-15675000 (BIN)														
D13 — D10	23672600 (BIN)														

NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.6 Istruzioni per programmi strutturati

Le istruzioni per programmi strutturati richiamano programmi o parti di programma, oppure commutano fra di essi. Sono inoltre disponibili istruzioni per indicizzazione e ripetizione di programma (loop).

La tabella che segue offre una panoramica di queste istruzioni:

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Istruzioni di ripetizione	FOR	FOR_M
	NEXT	NEXT_M
	BREAK	BREAK_MD
	BREAKP	BREAK_P_MD
Richiamo di sottoprogramma	CALL	CALL_M
	CALLP	CALLP_M
	RET	RET_M
	FCALL	FCALL_MD
	FCALLP	FCALL_P_MD
Richiamo sottoprogrammi fra file di programma (solo con GX Developer)	ECALL	ECALL_M
	ECALLP	ECALLP_M
	EFCALL	EFCALL_M
	EFCALLP	EFCALLP_M
Commutazione programma principale/sottoprogramma	CHG	CHG_M
Chiamata a programma microcomputer	SUB	SUB_M
	SUBP	SUBP_M
Indicizzazione di interi segmenti	IX	IX_MD
	IXEND	IXEND_MD
Definizione di valori di indicizzazione di interi segmenti	IXDEV	IXDEV_M
	IXSET	IXSET_M

7.6.1 FOR, NEXT

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

Operandi utilizzabili														Definizione cifra	Numero di passi	Indice	Riporto	Errore					
Operandi a bit						Operandi a word (16-bit)						Costanti							Puntatore		Livello	M9012	M9010 M9011
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1				Z	V	K	H (16#)	P	I	N
n							●	●	●	●	●	●	●	●	●	●					3/1	●	●

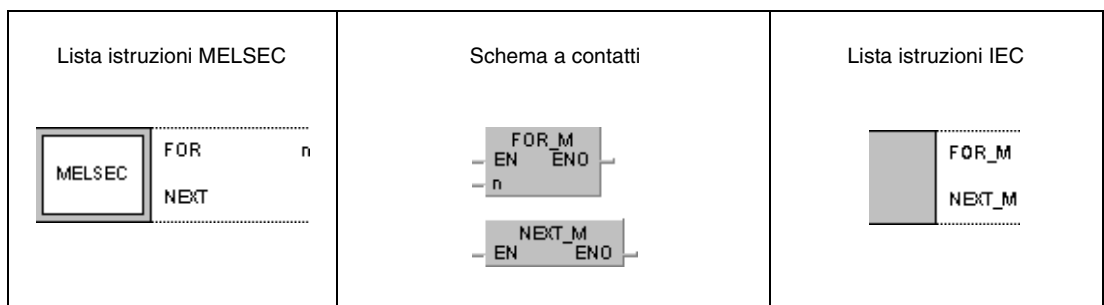
¹ L'istruzione FOR richiede tre passi, mentre l'istruzione NEXT richiede un passo. Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

Operandi MELSEC Q

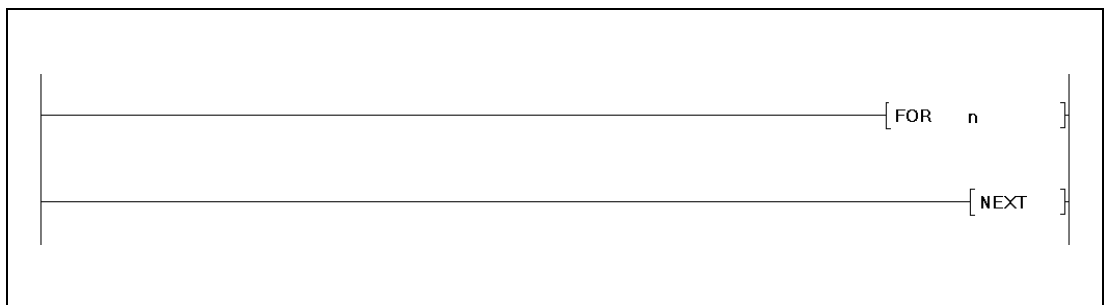
Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word							
n	●	●	●	●	●	●	●	●	—	SM0	2/1 ●

¹ L'istruzione FOR richiede due passi, mentre l'istruzione NEXT richiede un passo.

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
n	Numero ripetizioni per loop FOR/NEXT (da 1 a 32767).	BIN 16-bit

Funzioni Istruzione loop FOR/NEXT**FOR/NEXT Istruzione di loop**

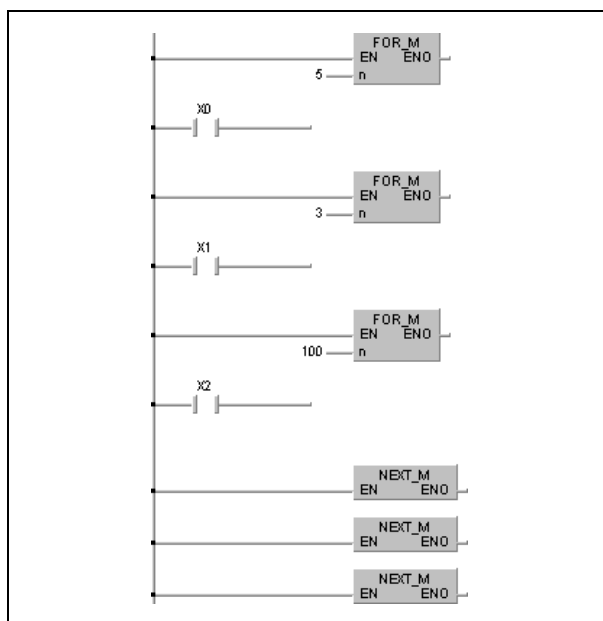
Il loop FOR/NEXT ripete singole sequenze di programma senza impostare una condizione di ingresso. La sequenza di programma posta fra i comandi FOR e NEXT viene ripetuta n volte.

Dopo aver eseguito il loop FOR/NEXT per n volte, viene eseguito il passo di programma successivo all'istruzione NEXT.

La variabile n può contenere un numero da 1 a 32767. Se n è inferiore o uguale a 0, viene considerato come 1. Quindi il loop FOR/NEXT viene eseguito almeno una volta.

Se una sezione di programma all'interno di un loop FOR/NEXT non deve essere eseguita, può essere scavalcata con una istruzione di salto (CJ o SCJ).

Si possono annidare in totale fino a 16 loop FOR/NEXT (5 nella serie A) La figura seguente illustra il principio di annidamento:

**Errori di esecuzione**

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- L'istruzione END/FEND viene eseguita dopo una istruzione FOR e prima di una istruzione NEXT (serie Q e System Q = codice di errore 4200).
- L'istruzione NEXT viene eseguita prima dell'istruzione FOR. (serie Q e System Q = codice di errore 4201).
- Il numero delle istruzioni FOR non corrisponde al numero di istruzioni NEXT.
- Una istruzione JMP con una destinazione all'esterno del loop FOR/NEXT viene eseguita all'interno di un loop FOR/NEXT.
- Una istruzione STOP è programmata all'interno di un loop FOR/NEXT. (serie Q e System Q = codice di errore 4200).
- Superato il massimo numero di livelli di annidamento (serie Q e System Q = codice di errore 4202).

NOTA

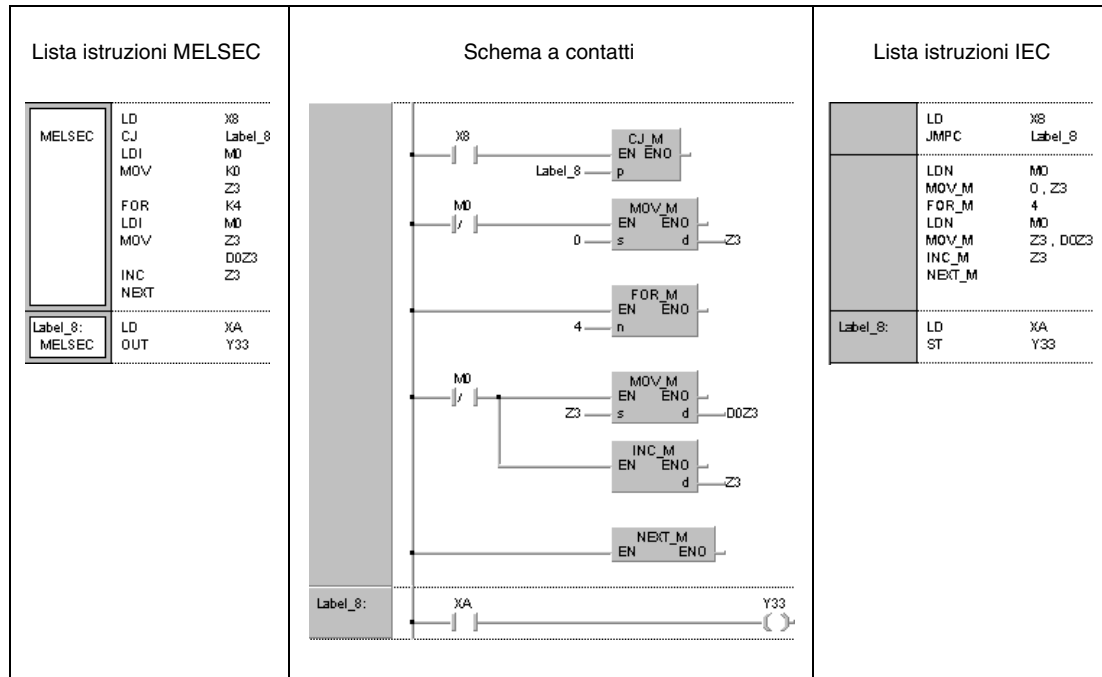
Solo serie Q e System Q:

Per interrompere l'esecuzione di un loop FOR/NEXT prima del suo termine, si può programmare una istruzione BREAK.

Utilizzare l'istruzione EGP/EGF per collegare lo stato di un contatto con l'istruzione FOR/NEXT.

Esempio di programma

Il programma seguente elabora la sequenza di programma fra le istruzioni FOR e NEXT per quattro volte, se X8 non è attivo. Il loop FOR/NEXT viene saltato se X8 è attivo.



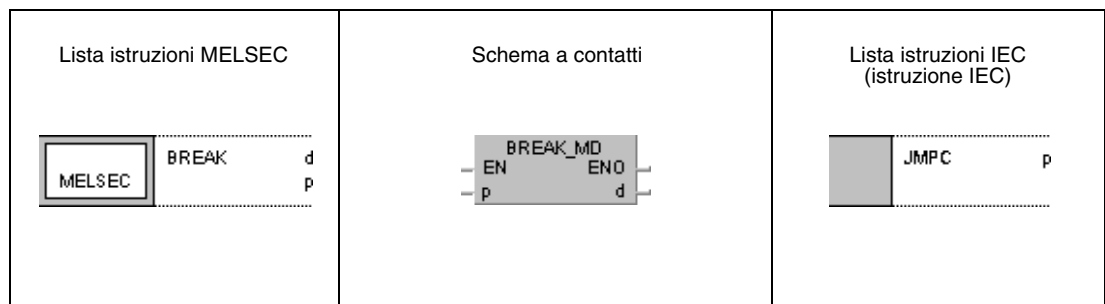
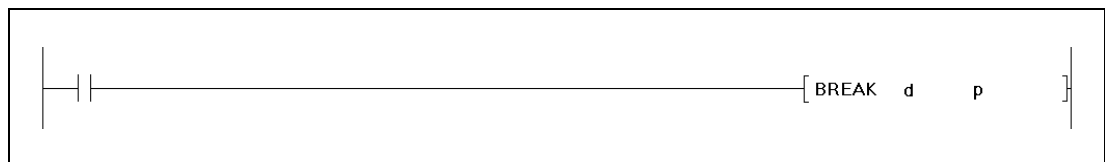
7.6.2 BREAK, BREAKP**CPU**

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	●

¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro P
	Bit	Word		Bit	Word						
d	●	●	●	●	●	●	—	—	SM0	3	
p	●	●	●	●	●	●	—	●			

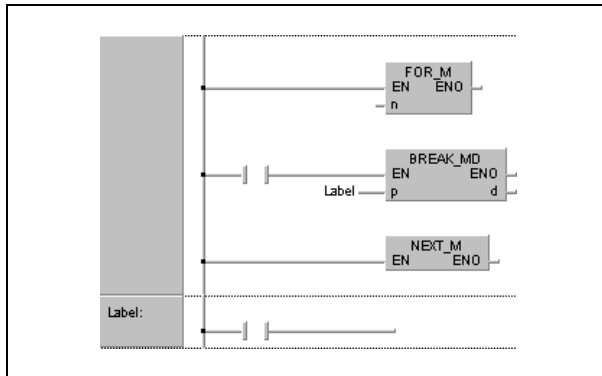
GX IEC Developer**GX Developer****Variabili**

Dati impostati	Significato	Tipo dati
d	Operando che contiene il numero di loop FOR/NEXT restanti	BIN 16-bit
p	Operando di destinazione (puntatore/label) a cui saltare dopo l'esecuzione dell'istruzione BREAK.	Puntatore/Label

Funzioni Terminazione di un loop FOR/NEXT

BREAK Termina l'esecuzione di un loop FOR/NEXT

L'istruzione BREAK interrompe l'esecuzione di un loop FOR/NEXT e salta al puntatore/etichetta specificato da p.



Il numero di loop FOR/NEXT ancora da eseguire viene memorizzato nell'operando indicato da d.

L'istruzione BREAK può essere utilizzata solo all'interno di un loop FOR/NEXT.

L'istruzione BREAK interrompe solo l'esecuzione del livello di annidamento attuale. Per interrompere diversi livelli di annidamento, si deve eseguire il numero appropriato di istruzioni BREAK.

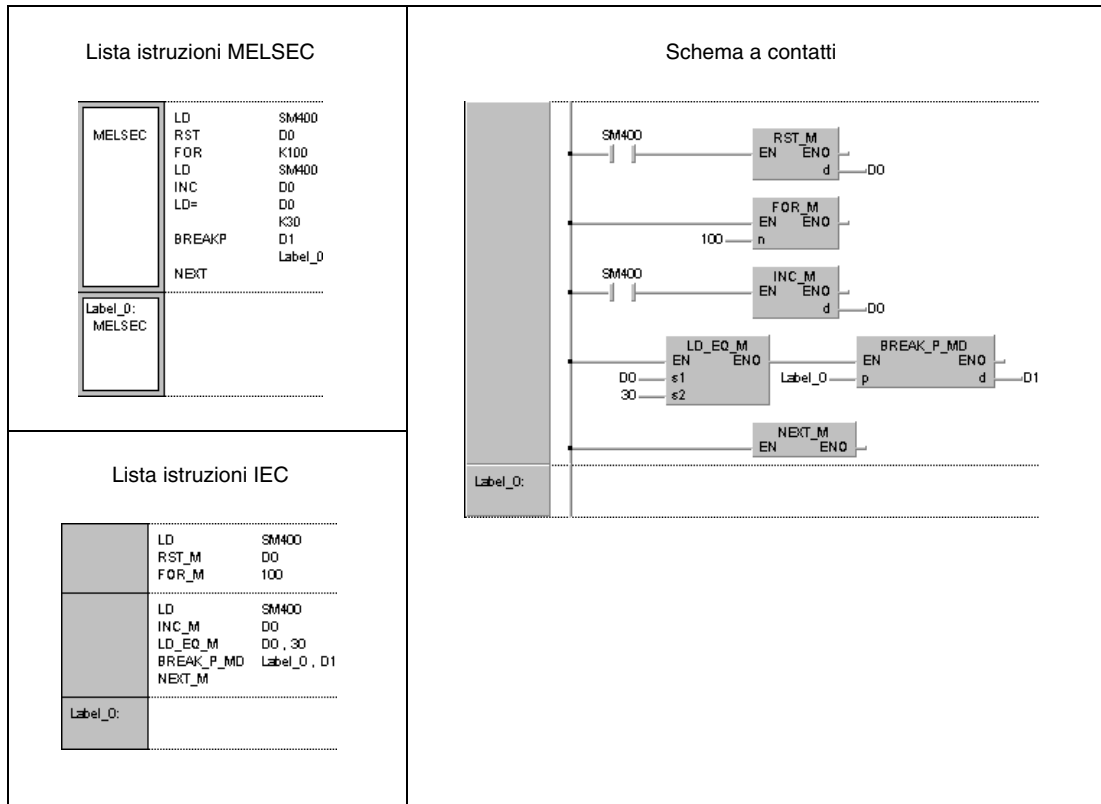
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- L'istruzione BREAK è stata eseguita fuori da un loop FOR/NEXT. (serie Q e System Q = codice di errore 4203).
- Nessun sottoprogramma disponibile all'indirizzo specificato dal puntatore/etichetta (serie Q e System Q = codice di errore 4210).

Esempio di programma BREAKP

Il programma seguente interrompe l'esecuzione durante il trentesimo loop FOR/NEXT e salta alla parte di programma contrassegnata con label_0. Il numero di loop FOR/NEXT ancora da eseguire viene memorizzato in D1.



7.6.3 CALL, CALLP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

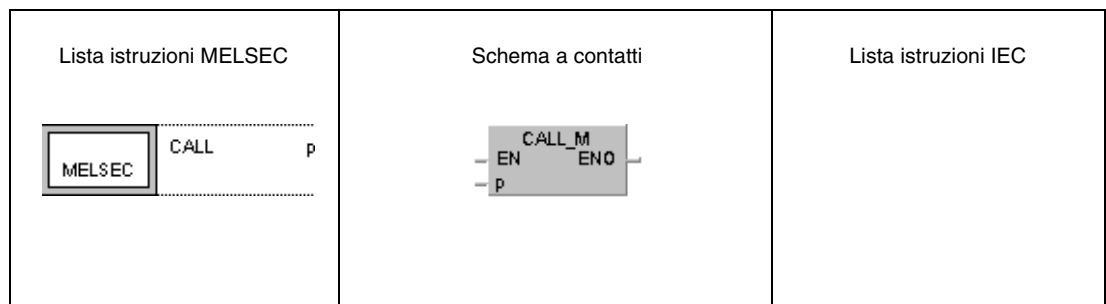
Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Rapporto	Errore				
Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello									
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H (16#)	P	I	N	M9012	M9010 M9011	
p																		●				● ³ ₁	●

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

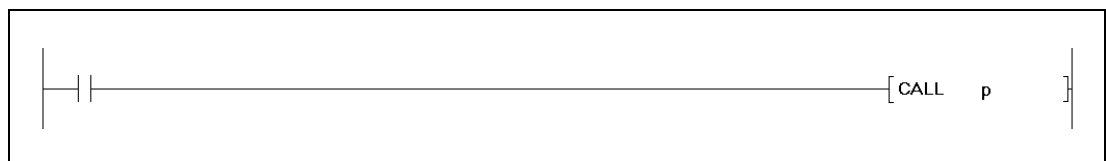
Operandi MELSEC Q

Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti	Altro			
Bit	Word		Bit	Word							
p	—	●	●	—	—	—	—	—	—	SM0	2

GX IEC Developer



GX Developer



Variabili

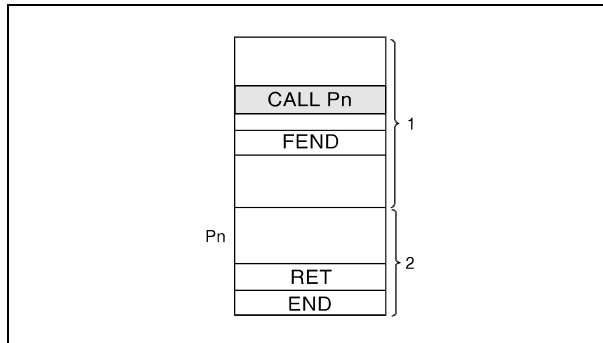
Dati impostati	Significato	Tipo dati
pn	Indirizzo (puntatore/etichetta) del sottoprogramma.	Puntatore/Label

NOTA

L'istruzione CALL non deve essere usata con l'editor IEC perché la struttura del sottoprogramma è stata generata da GX IEC Developer.

Funzioni **Richiamo di un sottoprogramma****CALL** **Richiamo sottoprogramma**

L'istruzione CALL richiama una subroutine specificata da un puntatore Pxx in GX Developer o da una label in GX IEC Developer. Gli indirizzi del puntatore (label) nella serie A variano da P(label)0 a P(label)255. Gli indirizzi del puntatore (label) nella serie Q e System Q variano da P(label)0 a P(label)4095. Fare riferimento alle note sulla programmazione degli indirizzi di puntatori(label) per le istruzioni di salto (CJ, SCJ, JMP).



¹ Programma principale

² Sottoprogramma

L'istruzione CALL richiama un sottoprogramma specificato da un puntatore (label). Si possono utilizzare in totale fino a 5 livelli di annidamento per sottoprogrammi nella serie A e fino a 16 livelli per la serie Q.

Gli operandi che vengono attivati durante l'esecuzione di un sottoprogramma, rimangono attivi anche se il sottoprogramma non viene più eseguito. Per disattivare questi operandi, utilizzare l'istruzione FCALL.

Errori di esecuzione

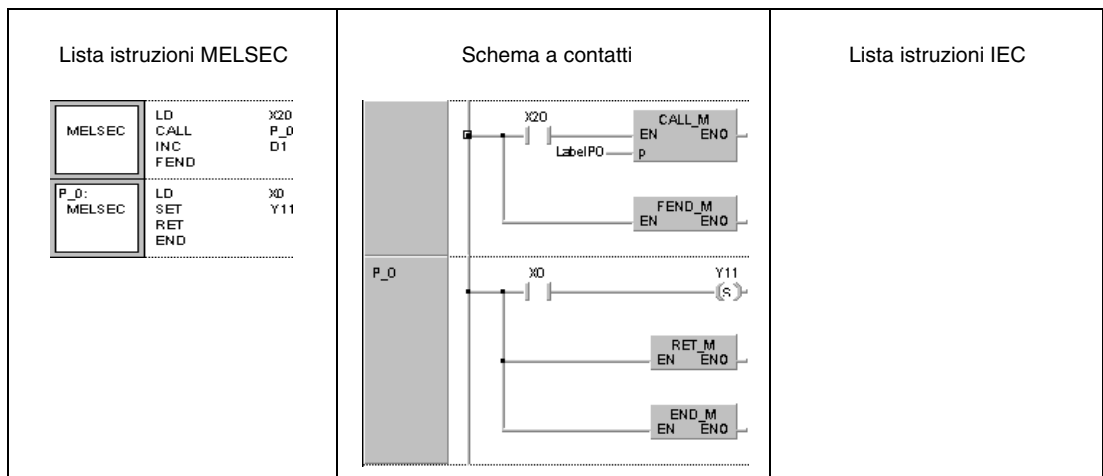
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Dopo l'esecuzione di una istruzione CALL, viene eseguita una istruzione END, FEND, GOEND, o STOP prima dell'istruzione RET (serie Q e System Q = codice di errore 4211).
- Una istruzione RET viene eseguita prima di una istruzione CALL. (serie Q e System Q = codice di errore 4212).
- Vengono utilizzati più di 5 livelli di annidamento (serie A) o 16 livelli di annidamento (serie Q). (serie Q e System Q = codice di errore 4210).
- Nessun sottoprogramma disponibile all'indirizzo specificato dal puntatore/etichetta (serie Q e System Q = codice di errore 4210).
- Una istruzione CALL specifica un indirizzo di puntatore (label) superiore a P(label)255 (serie A).
- Il sottoprogramma viene terminato da una istruzione JMP prima di una istruzione RET (serie A).

Esempio di programma

CALL

Mentre X20 è attivo, il programma seguente esegue il sottoprogramma specificato dal puntatore/label P_0.



NOTA

Nella modalità MELSEC, le istruzioni FEND, END e RET devono essere programmate dall'utente. Al termine dell'elaborazione di questa unità organizzativa di programma, le successive non verranno eseguite, in quanto successive all'istruzione FEND.

In alternativa a questa programmazione, è possibile utilizzare l'editor IEC. In questo caso l'istruzione FEND viene inserita automaticamente dal compilatore di GX IEC Developer.

7.6.4 RET

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A



Operandi utilizzabili																	Definizione cifra	Numero di passi	Indice	Riporto	Errore		
Operandi a bit							Operandi a word (16-bit)							Costanti	Puntatore	Livello				M9012	M9010 M9011		
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K						H (16#)	P
																					1	●	●

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

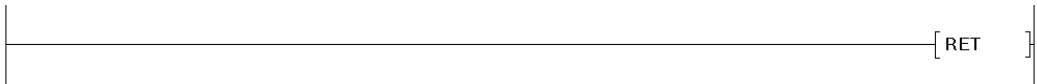
Operandi MELSEC Q

Operandi utilizzabili										Segnale di errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti	Altro			
Bit	Word		Bit	Word							
—	—	—	—	—	—	—	—	—	—	SM0	1

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
		

GX Developer

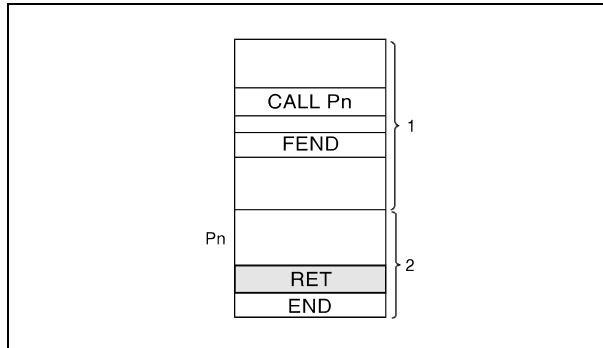

--

Variabili

Dati impostati	Significato	Tipo dati
—	—	—

Funzioni **Fine sottoprogramma****RET** **Ritorno al programma principale**

L'istruzione RET contrassegna la fine di una subroutine. Il programma ritorna al passo del programma principale successivo a quello dell'istruzione CALL, FCALL, ECALL o EFCALL.



¹ Programma principale

² Sottoprogramma

NOTA

Fra l'istruzione RET del sottoprogramma e l'istruzione END del programma principale, è necessario inserire una istruzione NOP, altrimenti la CPU non elabora correttamente il programma (solo serie A).

Nella modalità MELSEC, le istruzioni FEND, END e RET devono essere programmate dall'utente. Al termine dell'elaborazione di questa unità organizzativa di programma, le successive non verranno eseguite, in quanto successive all'istruzione FEND.

In alternativa a questa programmazione, è possibile utilizzare l'editor IEC. In questo caso l'istruzione FEND viene inserita automaticamente dal compilatore di GX IEC Developer.

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Dopo l'esecuzione di una istruzione CALL, viene eseguita una istruzione END, FEND, GOEND, o STOP prima dell'istruzione RET (serie Q e System Q = codice di errore 4211).
- Una istruzione RET viene eseguita prima di una istruzione CALL. (serie Q e System Q = codice di errore 4212).

7.6.5 FCALL, FCALLP

CPU

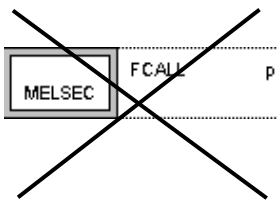
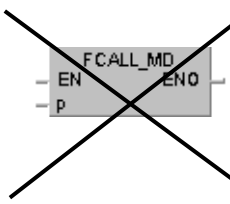
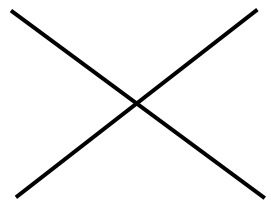
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	●

¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

Operandi
MELSEC Q

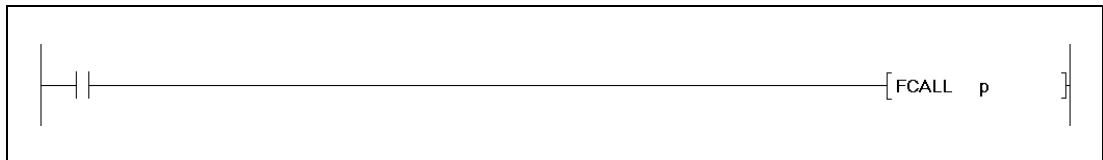
	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti			Altro
	Bit	Word		Bit	Word						
p	—	●	●	—	—	—	—	—	—	SM0	2

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
		

NOTA

Queste istruzioni non sono disponibili in GX IEC Developer.

GX
Developer

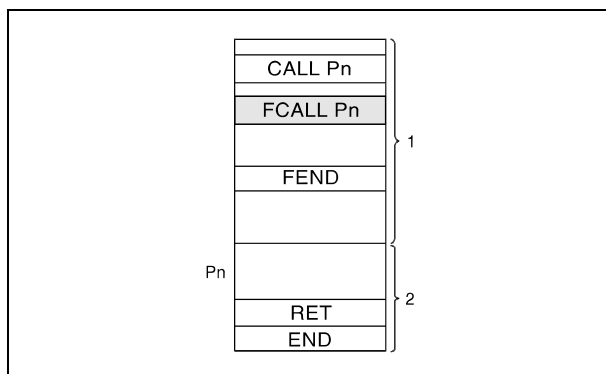
Variabili

Dati impostati	Significato	Tipo dati
pn	Indirizzo (puntatore/etichetta) del sottoprogramma.	Puntatore/Label

Funzioni Disattivazione uscite in un sottoprogramma

FCALL Disattiva le uscite (unitamente ad una istruzione CALL)

Se la condizione di esecuzione dell'istruzione FCALL viene resettata, i contatti e bobine all'interno della subroutine specificata da p (puntatore/etichetta) vengono considerati come se l'istruzione corrispondente non fosse settata.



¹ Programma principale

² Sottoprogramma

Lo stato di bobine e contatti dopo l'esecuzione di una istruzione FCALL, oppure lo stato di bobine e contatti se la condizione di esecuzione non è attiva, viene riportato di seguito:

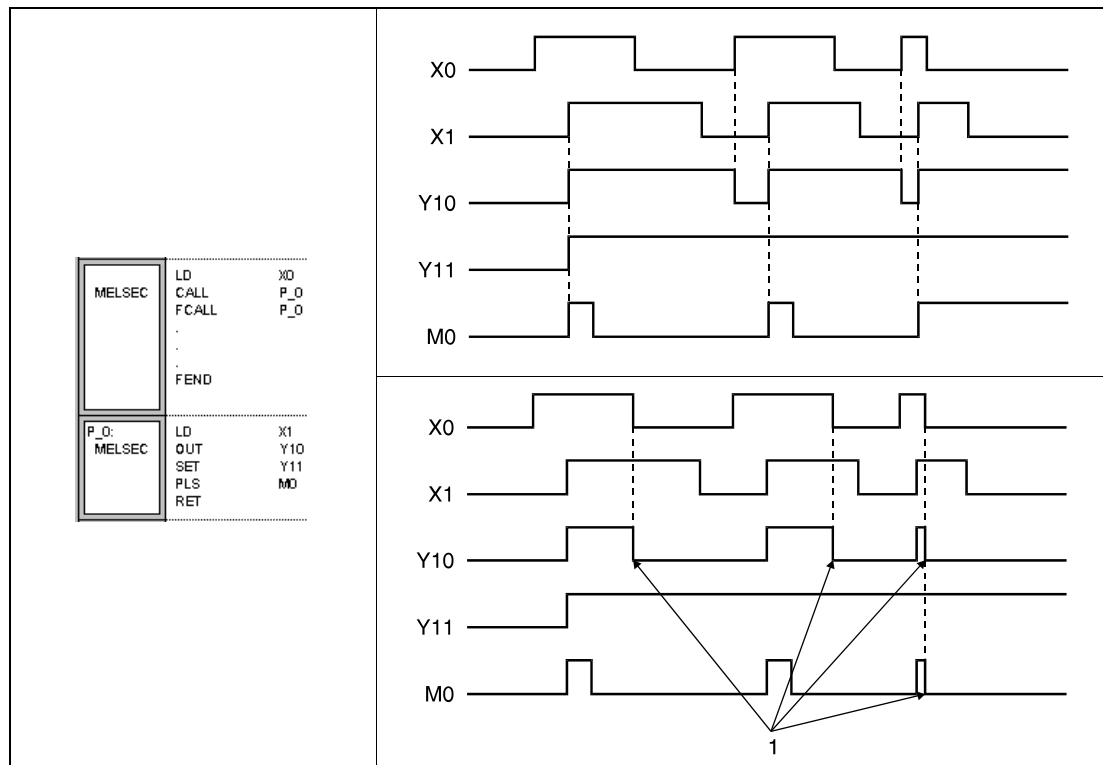
Istruzione	Stato contatto o bobina
Istruzione OUT	Tutti i contatti e bobine specificati dall'istruzione OUT vengono disattivati.
Istruzione SET	Tutti i contatti e bobine specificati da queste istruzioni rimangono nel loro stato.
Istruzione RST	
Istruzione SFT	
Istruzioni di base	
Istruzioni di applicazione	
Istruzione PLS	Tutti i contatti e bobine specificati da queste istruzioni assumono lo stato corrispondente a quello se le condizioni di esecuzione delle istruzioni non fosse attivo.
Istruzioni che generano un impulso di uscita	
Impostazione valori di timer lento o veloci	I valori impostati vengono azzerati.
Valori di impostazione dei timer retentivi	I valori impostati non vengono modificati.
Valori di impostazione dei contatori	

L'istruzione FCALL viene utilizzata insieme all'istruzione CALL.

La figura seguente mostra un programma che utilizza le istruzioni CALL e FCALL. Le figure sulla destra mostrano lo stato di diversi contatti utilizzati da diverse istruzioni. La figura in alto a destra mostra lo stato dei contatti senza l'uso di FCALL. La figura in basso a destra mostra lo stato dei contatti con l'uso di FCALL.

Se viene usata la sola istruzione CALL, lo stato di contatti e bobine utilizzate nel sottoprogramma rimane inalterato dopo la disattivazione della condizione di esecuzione dell'istruzione CALL (figura in alto a destra).

Se viene usata l'istruzione FCALL, lo stato di contatti e bobine utilizzate nel sottoprogramma viene azzerato dopo la disattivazione della condizione di esecuzione dell'istruzione FCALL (figura in basso a destra). Lo stesso vale per bobine e contatti usati dalle istruzioni OUT e PLS, o da una istruzione che genera un impulso.



¹ Disattivazione forzata con istruzione FCALL

L'istruzione FCALL richiama un sottoprogramma specificato tramite puntatore (label). Si possono programmare fino a 16 livelli annidati in totale.

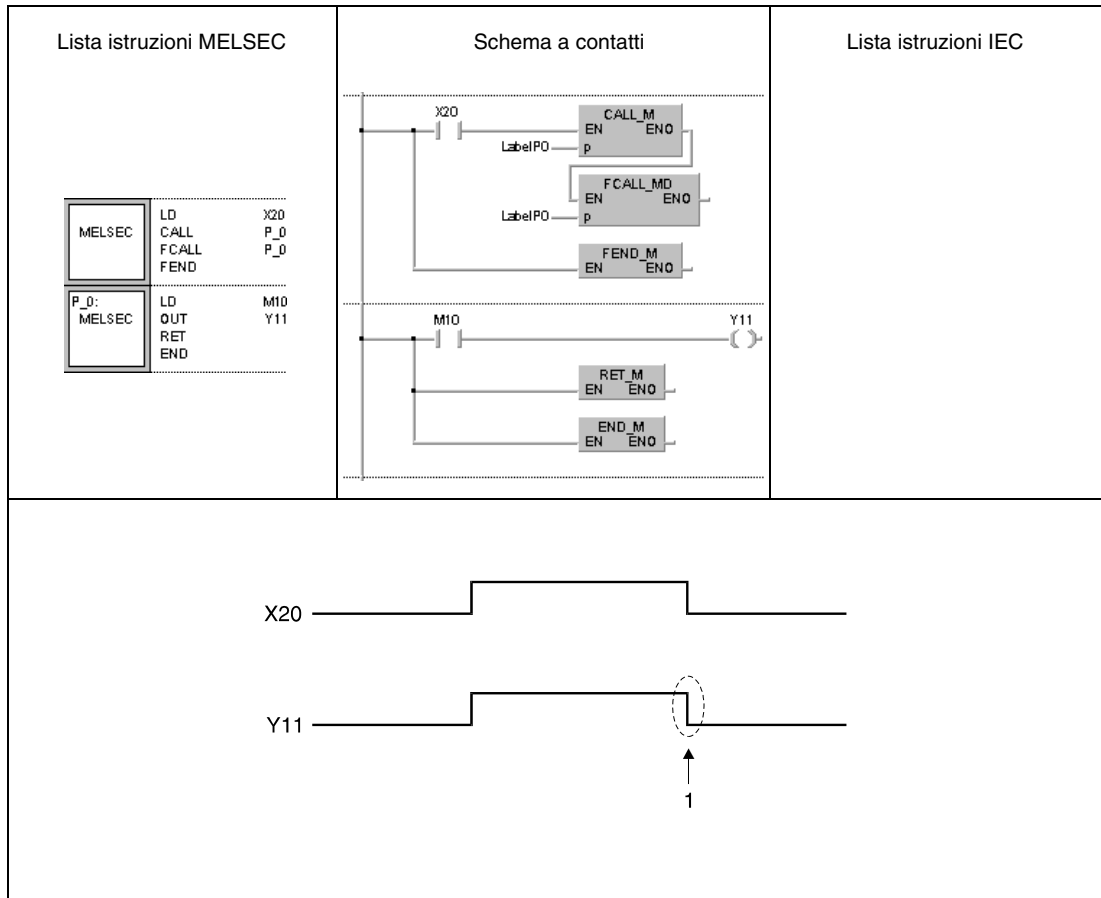
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Dopo l'esecuzione di una istruzione FCALL, viene eseguita una istruzione END, FEND, GOEND, o STOP prima dell'istruzione RET (serie Q e System Q = codice di errore 4211).
- Una istruzione RET viene eseguita prima di una istruzione FCALL. (serie Q e System Q = codice di errore 4212).
- Superato il massimo numeri di 16 livelli annidati (serie Q e System Q = codice di errore 4213).
- Nessun sottoprogramma disponibile all'indirizzo specificato dal puntatore/etichetta (serie Q e System Q = codice di errore 4210).

Esempio di programma FCALL

Fino a quando X20 rimane attivo, il programma seguente esegue la chiamata al sottoprogramma indicato del puntatore (label) P_0. Quando X0 si disattiva, l'istruzione FCALL disattiva a sua volta l'uscita Y11 (1).



7.6.6 ECALL, ECALLP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti			Altro
	Bit	Word		Bit	Word						
p	—	—	—☆	—	—	—	—	—	●	SMO	3
s1	● ¹	●	●	●	●	●	●	●	—		
s2	● ¹	●	●	●	●	●	●	●	—		
s3	● ¹	●	●	●	●	●	●	●	—		
s4	● ¹	●	●	●	●	●	●	●	—		
s5	● ¹	●	●	●	●	●	●	●	—		

¹ Gli annunciatori (F) non possono essere usati

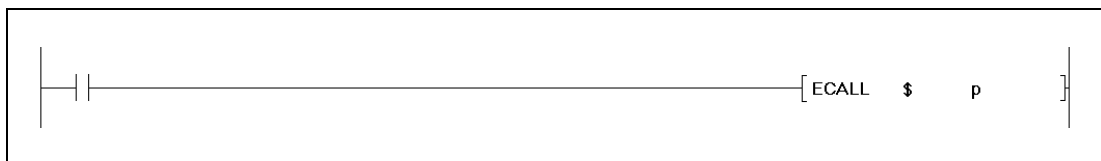
GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC

NOTA

Queste istruzioni non sono disponibili in GX IEC Developer.

GX Developer



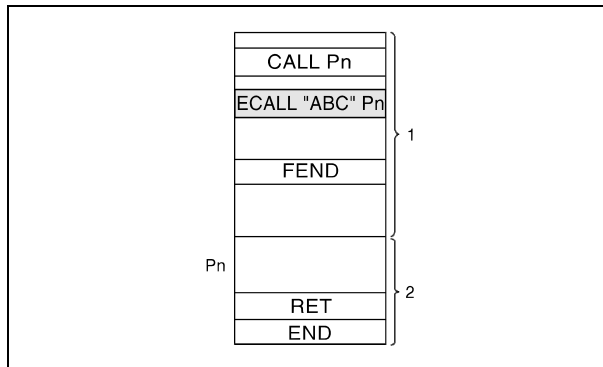
Variabili

Dati impostati	Significato	Tipo dati
Nome file	Nome del file che contiene il sottoprogramma.	Stringa caratteri
pn	Indirizzo (puntatore/etichetta) del sottoprogramma.	Puntatore/Label
da s1 a s5	Numero operando passato al sottoprogramma	Bit BIN16-bit BIN 32-bit

Funzioni Richiamo di un sottoprogramma in un file

ECALL Richiamo sottoprogramma

L'istruzione ECALL richiama una subroutine specificata da un puntatore (etichetta) in un file di programma indicato da un nome file. L'indirizzo del puntatore (etichetta) varia da P(label)0 a P(label)4085. Fare riferimento alle note sulla programmazione dei puntatori (label) per le istruzioni di salto (CJ, SCJ, JMP).



¹ Programma principale (nome file: MAIN)

² Sottoprogramma (nome file: ABC)

Il nome file può indicare solo file contenuti nella memoria interna (drive 0).

Quando si richiamano dei file di programma non è necessario indicare l'estensione.

L'istruzione ECALL richiama un sottoprogramma specificato tramite puntatore (label). Si possono programmare fino a 16 livelli annidati in totale. Il numero totale di 16 livelli si riferisce complessivamente ai livelli delle istruzioni CALL, FCALL, ECALL, e EFCALL.

Gli operandi che vengono attivati durante l'esecuzione di un sottoprogramma, rimangono attivi anche se il sottoprogramma non viene più eseguito. Per disattivare questi operandi, utilizzare l'istruzione EFCALL.

Se un sottoprogramma utilizza operandi funzione (FX, FY, FC), si possono passare argomenti alla funzione tramite gli operandi s1 - s5. Prima di eseguire un sottoprogramma, i dati a bit vengono passati a FX e i dati a word a FD. Al termine dell'esecuzione del sottoprogramma, i contenuti di FY e FD vengono inviati ai rispettivi operandi.

La quantità di dati che possono essere spostati in un registro funzione FD dipende dagli operandi specificati da s1 a s5. Si possono memorizzare fino a 2 word di costanti, registri indice e blocchi di operandi a bit, oppure fino a 4 word di operandi a word. Ad esempio se s2 indica il registro D0, i registri D0, D1, D2 e D3 verranno memorizzati in FD1.

Il numero di operandi funzione usati dai sottoprogrammi deve essere uguale al numero di operandi passati dall'istruzione ECALL tramite gli operandi da s1 a s5.

Il tipo degli operandi funzione deve essere identico ai tipi di operandi passati dall'istruzione ECALL.

Gli operandi specificati da s1 a s5 non devono essere sovrapposti.

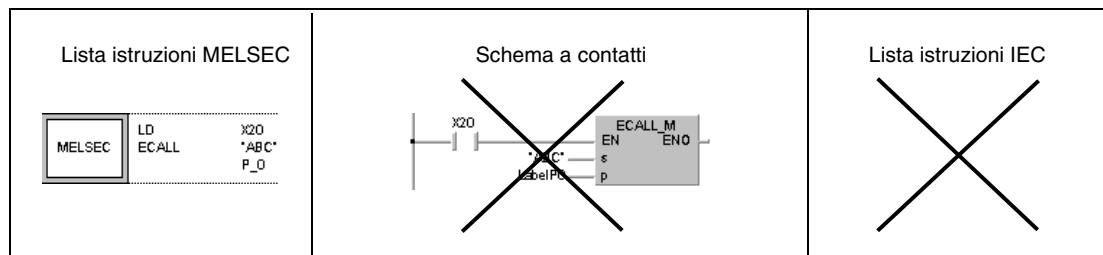
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Dopo l'esecuzione di una istruzione ECALL, viene eseguita una istruzione END, FEND, GOEND, o STOP prima dell'istruzione RET (serie Q e System Q = codice di errore 4211).
- Una istruzione RET viene eseguita prima di una istruzione ECALL (codice di errore 4212).
- Superato il massimo numero di 16 livelli annidati (codice di errore 4213).
- Gli operandi da s1 a s5 specificano un operando funzione (FX, FY, FD) (codice di errore 4101)
- Nessun sottoprogramma disponibile all'indirizzo specificato dal puntatore/etichetta (codice di errore 4210).
- Il file di programma specificato non esiste (codice di errore 4210).
- Il file di programma specificato non può essere eseguito (codice di errore 2411).

Esempio di programma**ECALL**

Mentre X20 è attivo, il programma seguente esegue il sottoprogramma specificato dal puntatore/label P_0 nel file di programma "ABC".



7.6.7 EFCALL, EFCALLP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

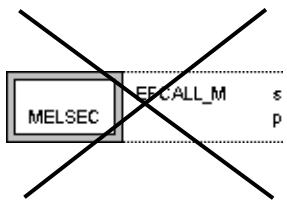
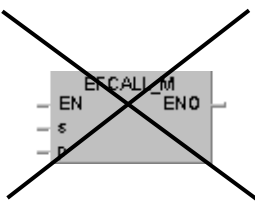
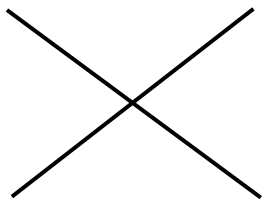
¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti			Altro
	Bit	Word		Bit	Word						
p	—	—	—	—	—	—	—	●	SM0	3	
s1	● ¹	●	●	●	●	●	●	—			
s2	● ¹	●	●	●	●	●	●	—			
s3	● ¹	●	●	●	●	●	●	—			
s4	● ¹	●	●	●	●	●	●	—			
s5	● ¹	●	●	●	●	●	●	—			

¹ Gli annunciatori (F) non possono essere usati

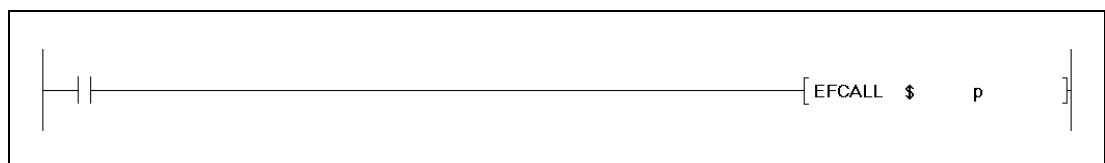
GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
--	---	---

NOTA

Queste istruzioni non sono disponibili in GX IEC Developer.

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
Nome file	Nome del file che contiene il sottoprogramma.	Stringa caratteri
pn	Indirizzo (puntatore/etichetta) del sottoprogramma.	Puntatore/Label
da s1 a s5	Numero operando passato al sottoprogramma	Bit BIN16-bit BIN 32-bit

Funzioni Disattivazione uscite in un sottoprogramma di un file**EFCALL Disattiva le uscite (unitamente ad una istruzione ECALL)**

Se la condizione di esecuzione dell'istruzione EFCALL viene resettata, i contatti e bobine all'interno della subroutine specificata da p (puntatore/etichetta) vengono considerati come se l'istruzione corrispondente non fosse settata.

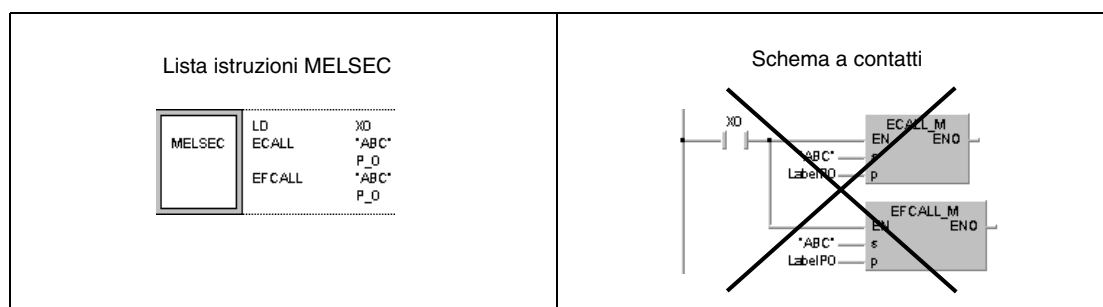
L'istruzione EFCALL esegue un sottoprogramma contenuto in un file diverso da quello del programma chiamante.

Lo stato di bobine e contatti dopo l'esecuzione di una istruzione EFCALL, oppure lo stato di bobine e contatti se la condizione di esecuzione non è attiva, viene riportato di seguito:

Istruzione	Stato contatto o bobina
Istruzione OUT	Tutti i contatti e bobine specificati dall'istruzione OUT vengono disattivati.
Istruzione SET	Tutti i contatti e bobine specificati da queste istruzioni rimangono nel loro stato.
Istruzione RST	
Istruzione SFT	
Istruzioni di base	
Istruzioni di applicazione	
Istruzione PLS	Tutti i contatti e bobine specificati da queste istruzioni assumono lo stato corrispondente a quello se le condizioni di esecuzione delle istruzioni non fosse attivo.
Istruzioni che generano un impulso di uscita	
Impostazione valori di timer lento o veloci	I valori impostati vengono azzerati.
Valori di impostazione dei timer retentivi	I valori impostati non vengono modificati.
Valori di impostazione dei contatori	

L'istruzione EFCALL viene utilizzata insieme all'istruzione ECALL.

La figura seguente mostra un programma che utilizza le istruzioni ECALL e EFCALL.



L'istruzione EFCALL richiama un sottoprogramma specificato tramite puntatore (label). Si possono programmare fino a 16 livelli annidati in totale. Il numero totale di 16 livelli si riferisce complessivamente ai livelli delle istruzioni CALL, FCALL, ECALL, e EFCALL.

Gli operandi che vengono attivati durante l'esecuzione di un sottoprogramma, rimangono attivi anche se il sottoprogramma non viene più eseguito. Per disattivare questi operandi, utilizzare l'istruzione EFCALL.

Se un sottoprogramma utilizza operandi funzione (FX, FY, FC), si possono passare argomenti alla funzione tramite gli operandi s1 - s5. Prima di eseguire un sottoprogramma, i dati a bit vengono passati a FX e i dati a word a FD.

Al termine dell'esecuzione del sottoprogramma, i contenuti di FY e FD vengono inviati ai rispettivi operandi.

La quantità di dati che possono essere spostati in un registro funzione FD dipende dagli operandi specificati da s1 a s5. Si possono memorizzare fino a 2 word di costanti, registri indice e blocchi di operandi a bit, oppure fino a 4 word di operandi a word. Ad esempio se s2 indica il registro D0, i registri D0, D1, D2 e D3 verranno memorizzati in FD1.

Il numero di operandi funzione usati dai sottoprogrammi deve essere uguale al numero di operandi passati dall'istruzione ECALL tramite gli operandi da s1 a s5.

Il tipo degli operandi funzione deve essere identico ai tipi di operandi passati dall'istruzione ECALL.

Gli operandi specificati da s1 a s5 non devono essere sovrapposti.

Errori di esecuzione

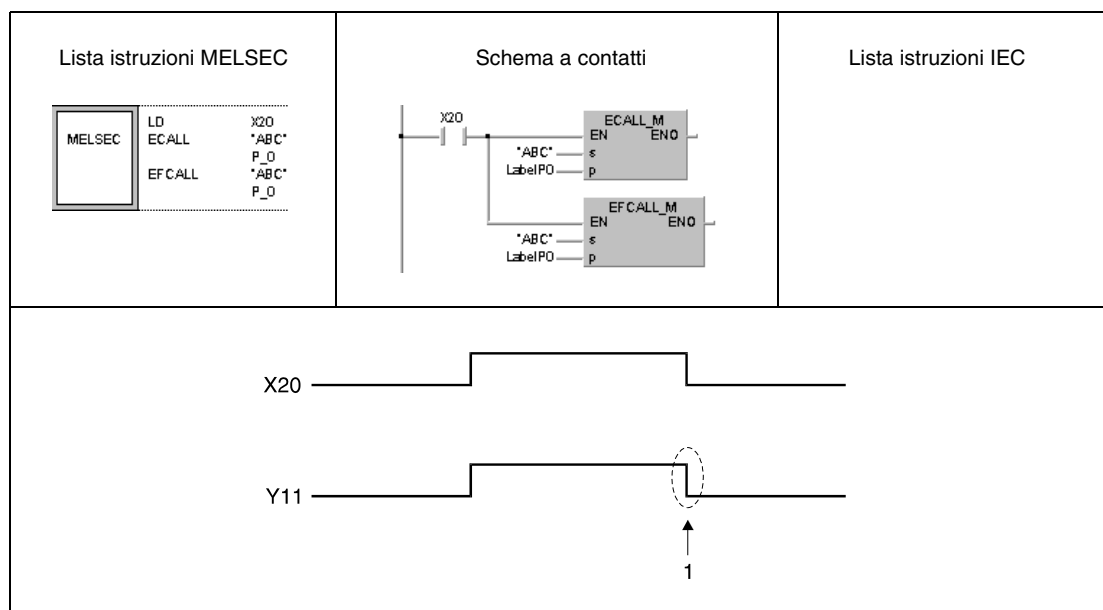
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Dopo l'esecuzione di una istruzione EFCALL, viene eseguita una istruzione END, FEND, GOEND, o STOP prima dell'istruzione RET (serie Q e System Q = codice di errore 4211).
- Una istruzione RET viene eseguita prima di una istruzione EFCALL (codice di errore 4212).
- Superato il massimo numero di 16 livelli annidati (codice di errore 4213).
- Nessun sottoprogramma disponibile all'indirizzo specificato dal puntatore/etichetta (codice di errore 4210).
- Gli operandi da s1 a s5 specificano un operando funzione (FX, FY, FD) (codice di errore 4101)
- Il file di programma specificato non esiste (codice di errore 4210).
- Il file di programma specificato non può essere eseguito (codice di errore 2411).

Esempio di programma

EFCALL

Mentre X20 è attivo, il programma seguente esegue il sottoprogramma specificato dal puntatore (label) P_0 nel file di programma "ABC". Se X20 viene disattivato, l'istruzione EFCALL azzerava anche l'uscita Y11 (1).



7.6.8 CHG

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
	● ¹	● ²	● ³		

¹ Solo CPU A3N


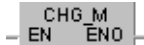
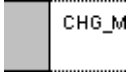
² Solo CPU A3A

³ Solo CPU A3U


Operandi
MELSEC A

Operandi utilizzabili																	Definizione cifra	Numero di passi	Indice	Riporto	Errore					
Operandi a bit							Operandi a word (16-bit)							Costanti		Puntatore						Livello				
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H (16#)	P	I	N						
																						1				

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
		

GX
Developer


--

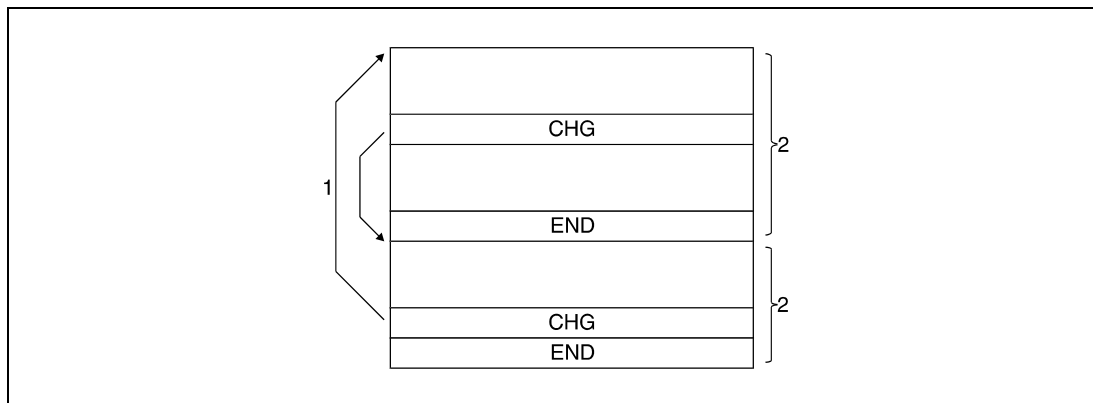
Variabili

Dati impostati	Significato	Tipo dati
—	—	—

Funzioni **Commutazione fra programma MAIN e SUB****CHG** **Istruzione di commutazione**

Con la condizione di ingresso attiva, l'istruzione CHG abilita la commutazione fra i programmi MAIN e SUB. La commutazione avviene dopo l'elaborazione di timer, contatori ed autodiagnosi.

Fare riferimento al capitolo 7.6.9 di questo manuale per le funzioni e uso delle parti di programma SUB.



¹ Elaborazione timer, contatori, autodiagnosi

² Programma sequenza

Commutazione fra programma MAIN e SUB

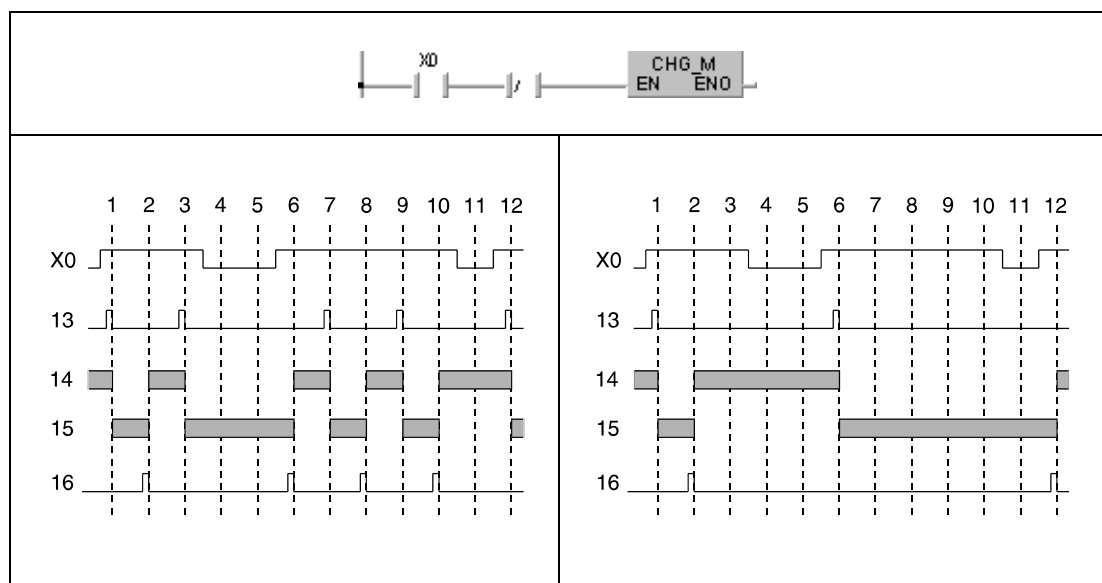
CHG Usando una CPU A3□

Con una CPU A3□ l'istruzione CHG viene eseguita solo sul fronte di salita della condizione di ingresso. Il risultato prodotto dalla condizione d'ingresso dipende dallo stato del relé interno M9050. La funzione svolta dall'istruzione CHG cambia quindi in funzione dello stato di M9050.

Le CPU A3N non supportano il relé interno M9050. L'elaborazione svolta corrisponde a quella con il relé M9050 attivo.

La parte superiore della figura seguente mostra la programmazione di una istruzione CHG. Questa parte di programma è inserita prima di una istruzione END o FEND all'interno di un programma MAIN o SUB.

La parte inferiore della figura mostra lo stato dei segnali corrispondenti. Gli stati sulla sinistra corrispondono al relé interno M9050 disattivato. Gli stati sulla destra corrispondono al relé M9050 attivato. La tabella seguente mostra l'elaborazione svolta in funzione dello stato di X0.



L'esecuzione dell'istruzione CHG nella sequenza MAIN è indicata con 13, la sequenza MAIN è indicata con 14, la sequenza SUB è indicata con 15 e l'esecuzione dell'istruzione CGH nella sequenza SUB è indicata con 16.

Stato di X0	Stato di M9050	
	OFF	ON
0	Nessuna commutazione fra le sequenze di programma MAIN e SUB (4, 5, 11).	Nessuna commutazione fra le sequenze di programma MAIN e SUB (4, 5, 11).
1	L'istruzione CHG viene eseguita ad ogni scansione e commuta fra le sequenze di programma MAIN e SUB (2, 3, 7, 8, 9, 10).	La sequenza MAIN viene commutata verso la sequenza SUB, poi ritorna alla sequenza MAIN sul primo fronte di salita di X0 (2).
0 → 1	Commutazione fra le sequenze di programma MAIN e SUB (1, 6, 12).	Commutazione fra le sequenze di programma MAIN e SUB (1, 6, 12).

Dopo l'esecuzione dell'istruzione CHG, viene eseguita l'elaborazione END per il programma corrente. L'elaborazione riparte con il passo 0 dell'altro programma. GX IEC Developer esegue automaticamente una commutazione alla fine della sequenza MAIN o SUB.

Istruzione CHG insieme all'istruzione PLS

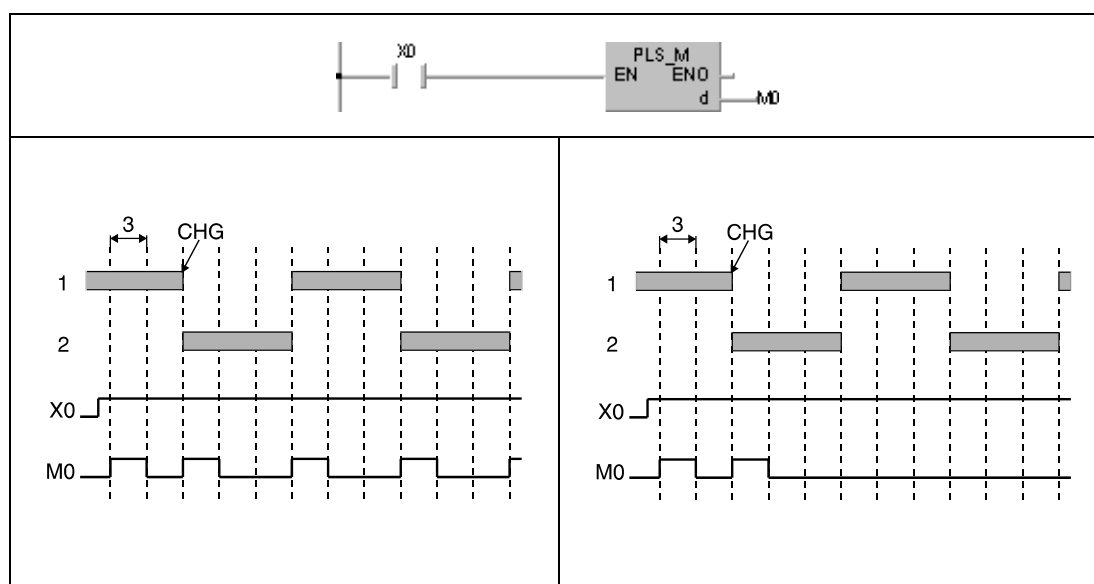
CHG Usando una CPU A3□

Con una CPU A3□ le funzioni svolte dall'istruzione PLS dipendono dallo stato del relé interno M9050.

Le CPU A3N non supportano il relé interno M9050. L'elaborazione svolta corrisponde a quella con il relé M9050 attivo.

La parte superiore della figura seguente mostra la programmazione di una istruzione PLS. La arte di programma è posta all'inizio (passo 0) della sequenza MAIN o SUB.

La parte inferiore della figura mostra lo stato dei segnali corrispondenti. Gli stati sulla sinistra corrispondono al relé interno M9050 disattivato. Gli stati sulla destra corrispondono al relé M9050 attivato. La tabella seguente mostra l'elaborazione svolta in funzione dello stato di X0.



L'elaborazione della sequenza MAIN è indicato con 1, l'elaborazione della sequenza SUB è indicato con 2, ed una scansione di programma è indicata con 3.

Stato di X0	Stato di M9050	
	OFF	ON
0	M0 non attivo.	M0 non attivo.
1	M0 è attivo solo per la durata della prima scansione dopo la commutazione con l'istruzione CHG.	M0 è attivo solo durante la prima scansione della sequenza SUB selezionata dall'istruzione CHG dopo che X0 è stato attivato.
0 → 1	M0 è attivo solo per una scansione.	M0 è attivo solo per una scansione.

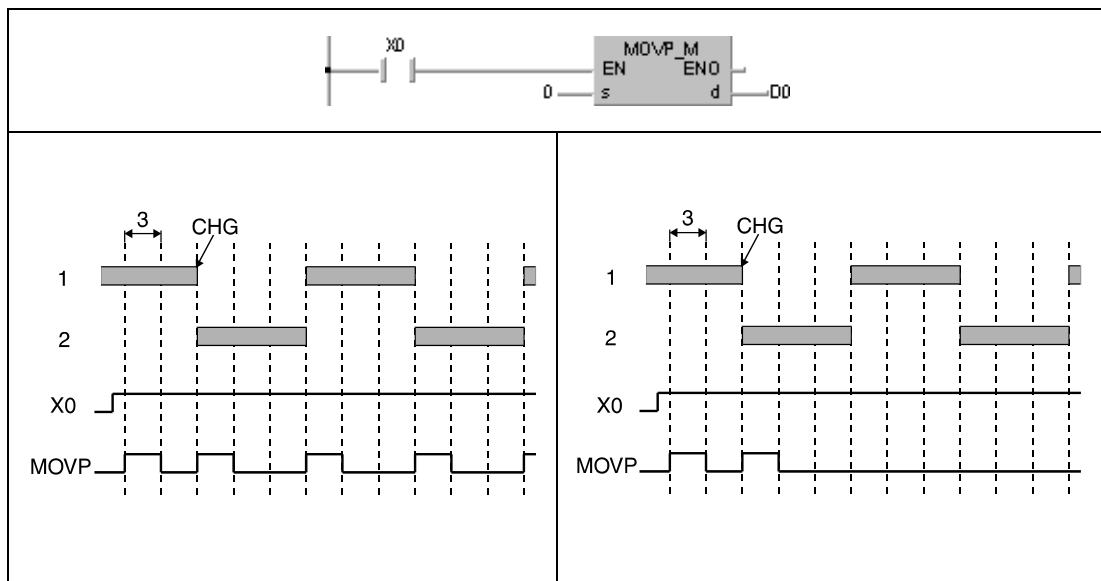
Istruzione CHG insieme ad una istruzione impulsiva (xP)**CHG Usando una CPU A3□**

Con una CPU A3□ le funzioni svolte da una istruzione impulsiva (xP) dipendono dallo stato del relé interno M9050.

Le CPU A3N non supportano il relé interno M9050. L'elaborazione svolta corrisponde a quella con il relé M9050 attivo.

La parte superiore della figura seguente mostra la programmazione di una istruzione impulsiva. La arte di programma è posta all'inizio (passo 0) della sequenza MAIN o SUB.

La parte inferiore della figura mostra lo stato dei segnali corrispondenti. Gli stati sulla sinistra corrispondono al relé interno M9050 disattivato. Gli stati sulla destra corrispondono al relé M9050 attivato. La tabella seguente mostra l'elaborazione svolta in funzione dello stato di X0.



L'elaborazione della sequenza MAIN è indicato con 1, l'elaborazione della sequenza SUB è indicato con 2, ed una scansione di programma è indicata con 3.

Stato di X0	Stato di M9050	
	OFF	ON
0	L'istruzione MOV_P non viene eseguita.	L'istruzione MOV_P non viene eseguita.
1	L'istruzione MOV_P viene eseguita solo durante la prima scansione dopo la commutazione con l'istruzione CHG.	L'istruzione MOV_P viene eseguita solo durante la prima scansione della sequenza SUB selezionata dall'istruzione CHG dopo che X0 è stato attivato.
0 → 1	L'istruzione MOV_P viene eseguita una volta.	L'istruzione MOV_P viene eseguita una volta.

Istruzione CHG e conteggio dei contatori

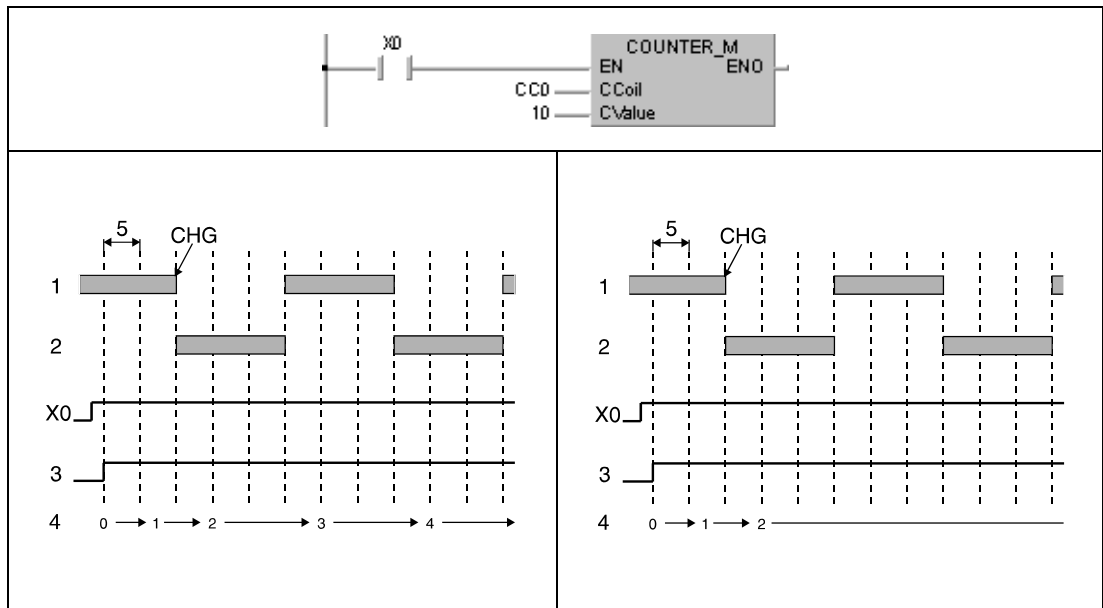
CHG Usando una CPU A3□

Con una CPU A3□CPU le funzioni dei contatori dipendono dallo stato del relé interno M9050, se tutte le altre condizioni di ingresso rimangono attive.

Le CPU A3N non supportano il relé interno M9050. L'elaborazione svolta corrisponde a quella con il relé M9050 attivo.

La parte superiore della figura seguente mostra la programmazione di una istruzione relativa a un contatore. La arte di programma è posta all'inizio (passo 0) della sequenza MAIN o SUB.

La parte inferiore della figura mostra lo stato dei segnali corrispondenti. Gli stati sulla sinistra corrispondono al relé interno M9050 disattivato. Gli stati sulla destra corrispondono al relé M9050 attivato. La tabella seguente mostra l'elaborazione svolta in funzione dello stato di X0.



L'elaborazione della sequenza MAIN è indicata con 1, l'elaborazione della sequenza SUB è indicata con 2, il contatto C0 è indicato con 3, il valore attuale di C0 è indicato con 4, ed una scansione di programma è indicata con 5.

Stato di X0	Stato di M9050	
	OFF	ON
0	Il valore attuale del contatore non è cambiato.	Il valore attuale del contatore non è cambiato.
1	Il valore attuale del contatore è incrementato di 1, dopo aver eseguito END (FEND, CHG) durante la prima scansione del programma selezionato dall'istruzione CHG.	Il valore attuale del contatore è incrementato di 1, dopo aver eseguito END (FEND, CHG) durante la prima scansione del programma selezionato dall'istruzione CHG eseguita dopo l'attivazione di X0.
0 → 1	Il valore attuale del contatore è incrementato di 1, dopo l'esecuzione di END (FEND, CHG).	Il valore attuale del contatore è incrementato di 1, dopo l'esecuzione di END (FEND, CHG).

Istruzione CHG e temporizzazioni dei timer

Tutte le CPU capaci di gestire l'istruzione CHG possiedono due diverse zone di memorizzazione per l'impostazione dei valori dei timer. Una per la sequenza MAIN e l'altra per la sequenza SUB. Quindi i timer vengono elaborati in base alla zona di memorizzazione attualmente usata (MAIN/SUB).

I valori di impostazione dei timer non utilizzati vengono azzerati nell'area di memorizzazione corrispondente. Un valore di impostazione 0 corrisponde ad un valore infinito che non trascorre mai.

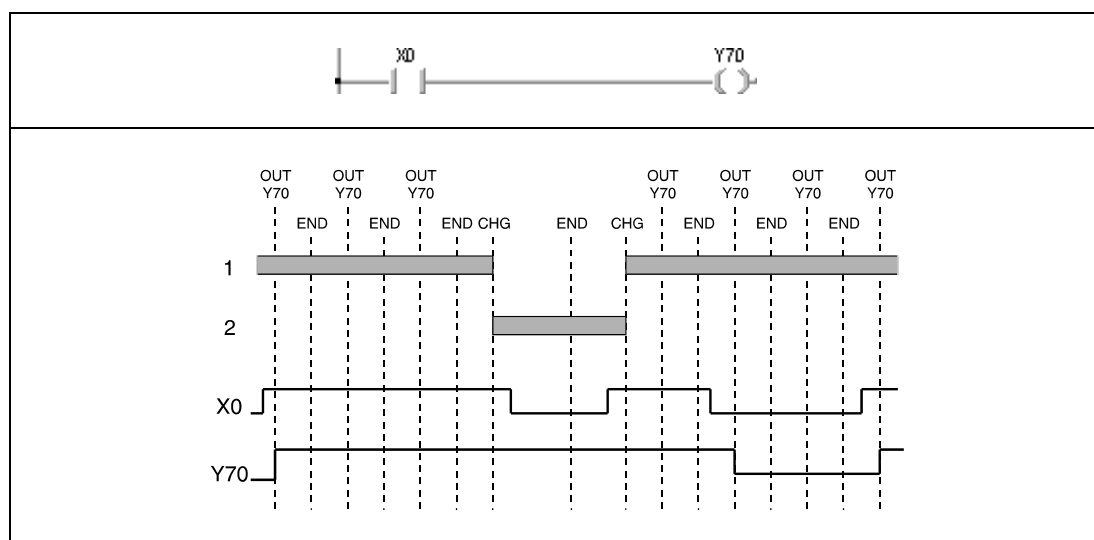
Se dopo aver avviato un timer, la zona di memorizzazione viene commutata tra MAIN/SUB tramite l'istruzione CHG, il timer non viene più elaborato nella sequenza di programma disattivata. Questo perché il timer era programmato nel programma sospeso, ed il suo valore di impostazione è considerato 0 nel programma attivato. Dopo il ritorno al programma sospeso, l'elaborazione del timer viene ripresa. Il timer scade quando il valore attuale supera il valore impostato, o diventa inferiore a 0. Quando il timer è scaduto, il suo contatto si chiude.

Istruzione CHG ed elaborazione delle istruzioni OUT

Tutte le CPU capaci di gestire l'istruzione CHG commutano i contatti di uscita in funzione del programma attualmente elaborato.

I contatti di uscita mantengono il loro stato dopo la commutazione dal programma attivo ad un altro programma (zona MAIN/SUB). Il loro stato rimane invariato anche se le condizioni di uscita cambiano.

La parte superiore della figura seguente mostra la programmazione di una istruzione OUT. Questa parte di programma è inserita nella zona di memorizzazione MAIN. L'uscita Y70 non è utilizzata nell'area di memorizzazione SUB.



La parte inferiore della figura mostra lo stato dei segnali. L'elaborazione della zona MAIN è indicata con 1, l'elaborazione della zona SUB è indicata con 2.

Durante l'elaborazione della zona MAIN, l'uscita Y70 viene attivata e disattivata in funzione dello stato dell'ingresso X0. Durante l'elaborazione della zona SUB, lo stato di Y70 rimane invariato anche se le condizioni di ingresso cambiano.

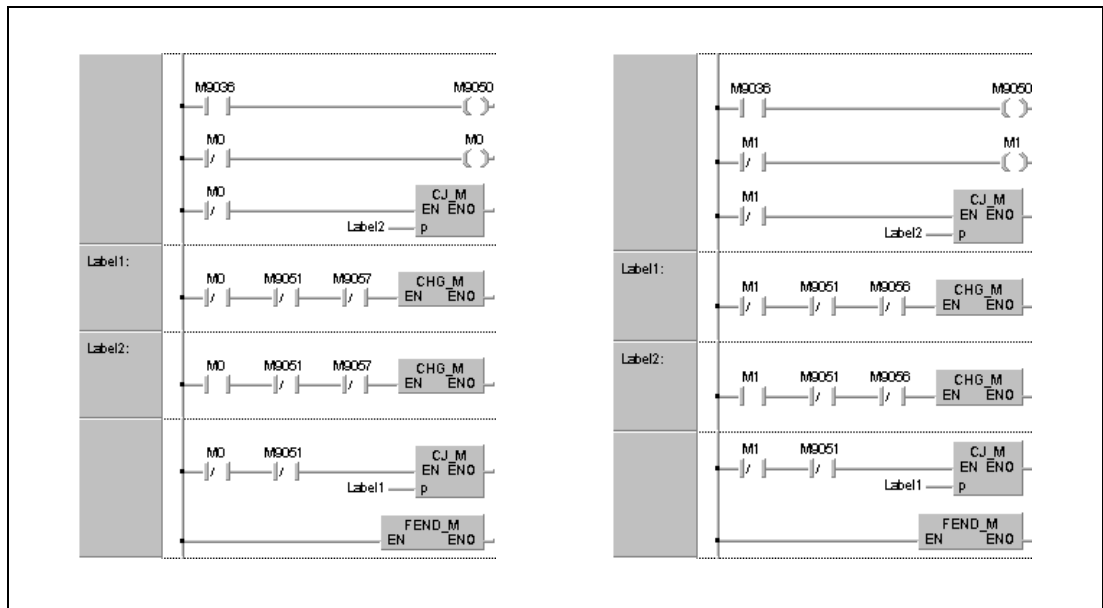
Programma di esempio 1

CHG (CPU A3□)

Per il corretto funzionamento dell'istruzione CHG, il risultato dell'operazione di una scansione deve essere confrontato con quello della scansione precedente. Per questa ragione, il relé interno M9050 deve essere attivato prima dell'istruzione CHG per caricare il risultato operazione della scansione precedente dal buffer di memoria in memoria centrale.

Dato che l'istruzione CHG viene eseguita da una CPU A3□ con la condizione di ingresso attiva, i programmi devono rispettare la struttura seguente. Il programma di sinistra è posto nella zona MAIN, mentre il programma di destra è posto nella zona SUB.

Il relé interno M9036 è sempre attivo-



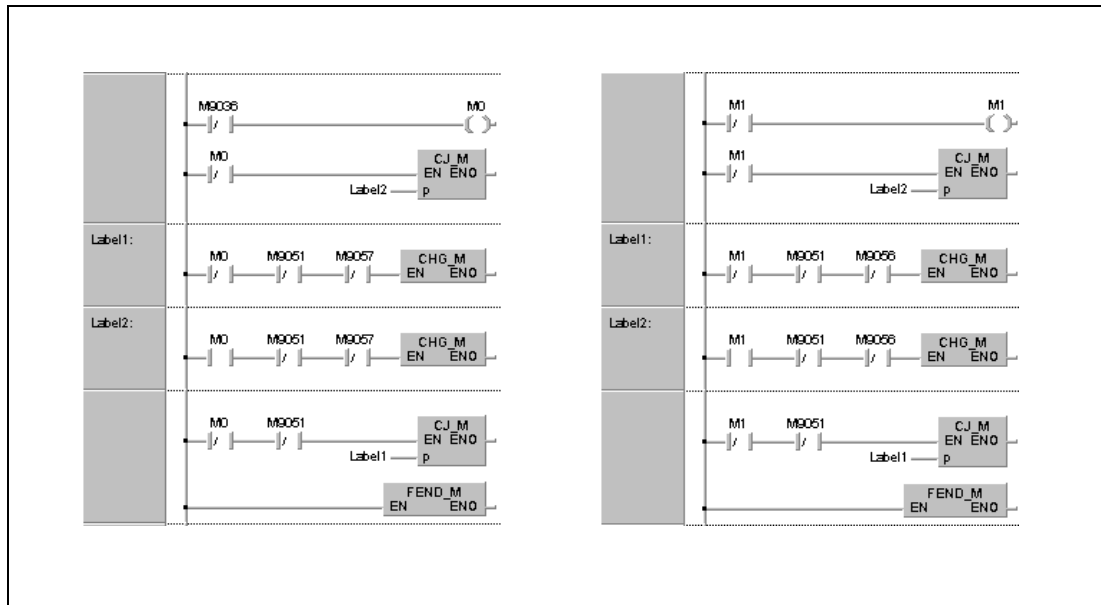
NOTA

Se si modifica un programma SUB quando è attivo il programma MAIN o viceversa, si devono utilizzare i relé interni M9051, M9056 e M9057 per disabilitare l'istruzione CHG, in modo che questa non possa commutare dal programma attivo a quello che si sta correggendo.

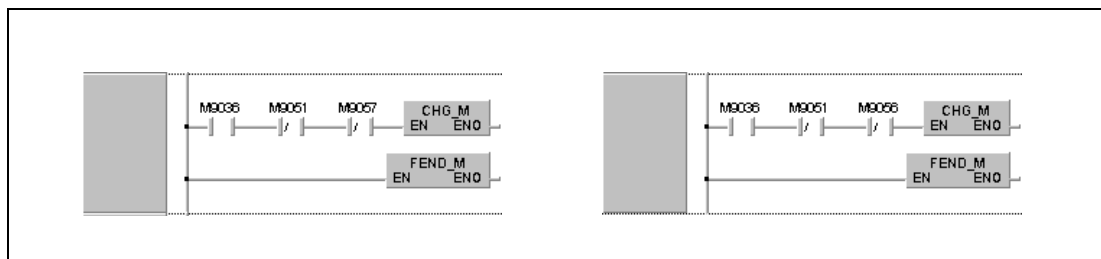
Quindi, durante una modifica online nella zona SUB, la zona MAIN non viene elaborata. Con GX IEC Developer e GX Developer questo tipo di programmazione viene raggiunto automaticamente.

Programma di esempio 2 CHG (CPU A3N)

Dato che l'istruzione CHG viene eseguita da una CPU A3N con la condizione di ingresso attiva, i programmi devono rispettare la struttura seguente. Il programma di sinistra è posto nella zona MAIN, mentre il programma di destra è posto nella zona SUB.

**Programma di esempio 3** CHG (CPU A3H)

I programmi devono essere scritti secondo la struttura seguente. Il programma di sinistra è posto nella zona MAIN, mentre il programma di destra è posto nella zona SUB.



7.6.9 SUB, SUBP

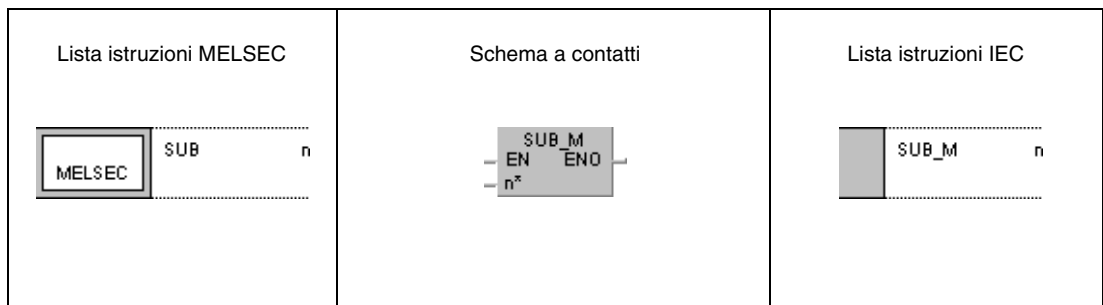
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●				

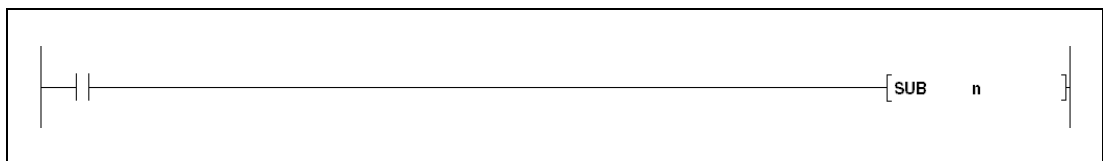
Operandi MELSEC A

Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Riporto	Errore					
Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello										
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H (16#)	P	I	N	M9012	M9010 M9011		
n							●	●	●	●	●	●	●	●	●	●						3	●	●

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
n	Indirizzo del programma microcomputer da richiamare.	Indirizzo

Funzioni Chiamata a programma microcomputer

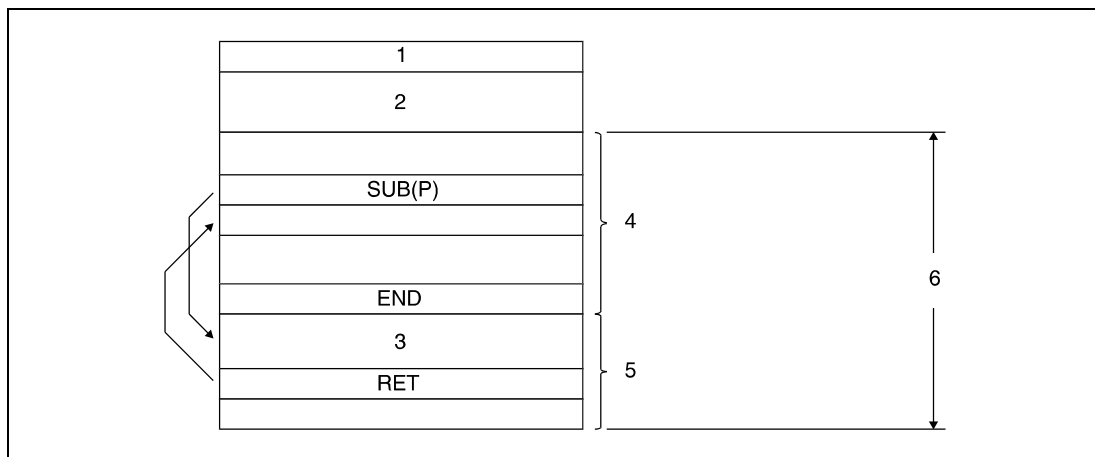
SUB Chiamata a programma microcomputer

Le istruzioni SUB/SUBP richiamano un programma microcomputer creato dall'utente.

Se la condizione di ingresso è attiva, l'istruzione SUB richiama il programma microcomputer posto all'indirizzo "n".

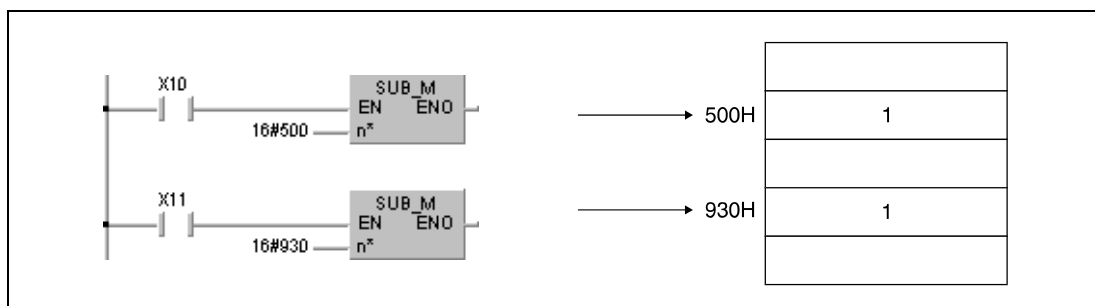
Dopo l'esecuzione del programma microcomputer, la sequenza di programma riprende ad essere elaborata dal passo successivo a quello dell'istruzione SUB/SUBP.

L'istruzione SUB/SUBP può essere programmata nelle zone programma MAIN e SUB.



- 1 Parametro
- 2 Valore di impostazione di timer e contatori
- 3 Programma microcomputer
- 4 Zona sequenza di programma
- 5 Zona programma microcomputer
- 6 Zona memorizzazione MAIN o SUB

La zona dei programmi microcomputer può contenere diversi programmi.



- 1 Programma microcomputer

NOTA *Fra le istruzioni dedicate delle CPU AnA, AnAS e AnU, l'istruzione SUB specifica una costante a 16 bit nel blocco dell'istruzione.*

Fare riferimento al capitolo 10 di questo manuale per ulteriori dettagli sui programmi microcomputer.

Errori di esecuzione Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- La massima capacità dedicata ai programmi microcomputer è stata superata.
- L'indirizzo definito da n supera il campo operando ammesso

NOTA *Il tempo di elaborazione di un programma microcomputer richiamato da una istruzione SUB(P) non deve superare 5ms. Se il limite di 5ms viene superato, si possono avere conflitti con il programma a sequenza ed il PLC può non funzionare correttamente.*

Se si deve eseguire un programma microcomputer che richiede più di 5ms di tempo di elaborazione, questo deve essere suddiviso in diversi blocchi da richiamare in sequenza. Questo metodo può abbreviare il tempo di elaborazione di un programma microcomputer richiamato da una istruzione SUB.

7.6.10 IX, IXEND

CPU

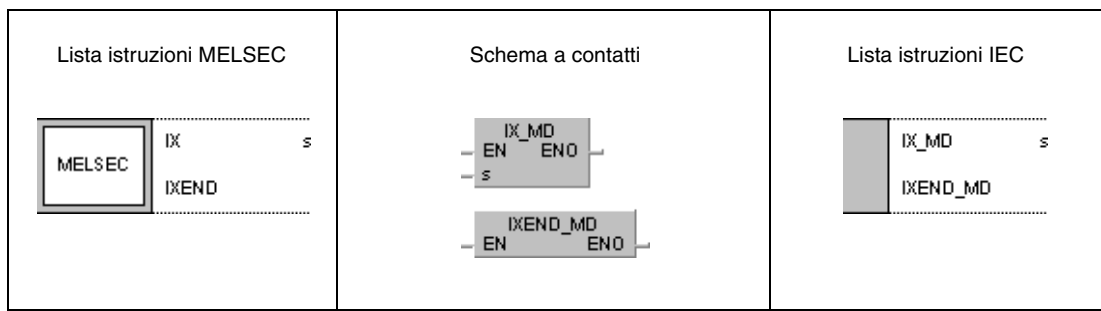
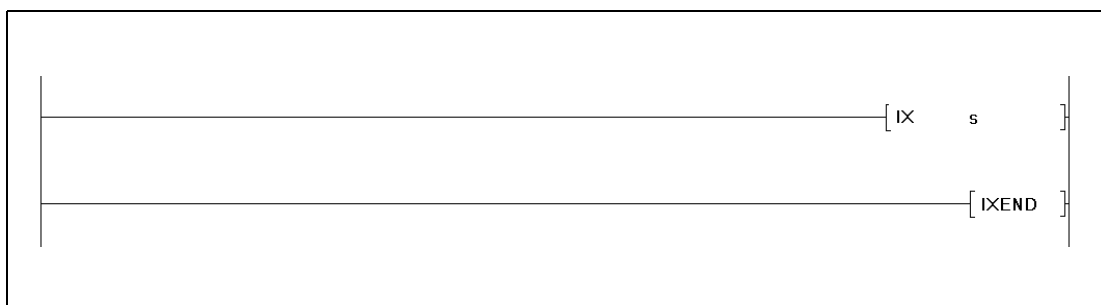
AnS	AnN	AnA, AnAS	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	●

¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

Operandi
MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	—	SM0	2/1 ● ¹	

¹ L'istruzione IX richiede due passi, mentre l'istruzione IXEND richiede un passo.

GX IEC
DeveloperGX
Developer

Variabili

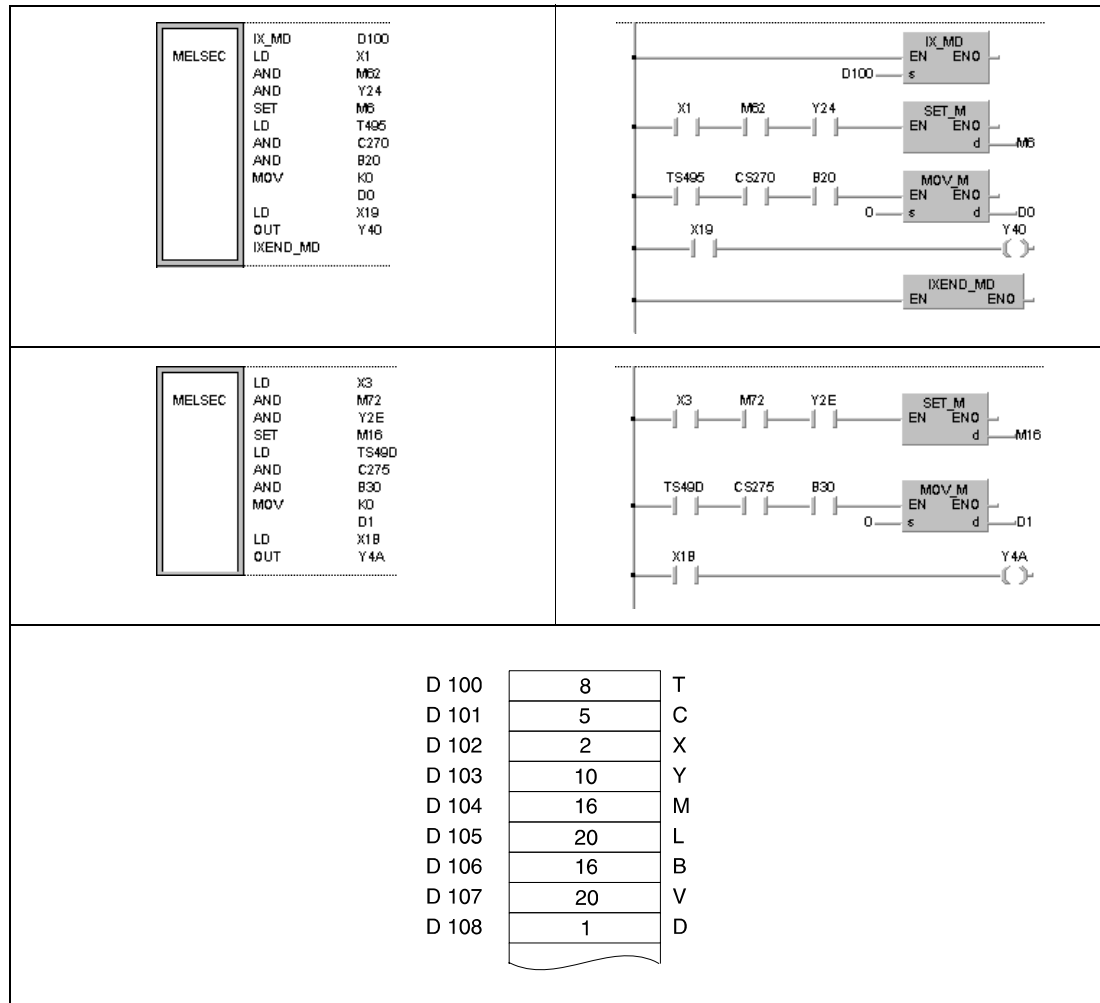
Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene i dati per l'indicizzazione.	BIN 16-bit

Funzioni Indicizzazione di intere sezioni di programma

IX, IXEND Istruzione di indicizzazione

Le istruzioni IX e IXEND sono supportate solo in modo MELSEC con GX IEC Developer. Le istruzioni IX e IXEND eseguono una indicizzazione per quegli operandi compresi nella porzione di programma compresa fra IX e IXEND.

Con l'indicizzazione, i valori decimali contenuti in una tabella indice (s) vengono sommati agli indirizzi degli operandi. Questo nuovo indirizzo in formato esadecimale diventa l'indirizzo valido per l'elaborazione ulteriore. A ciascun operando specificato da s viene assegnato un tipo specifico di operando, su cui deve essere eseguita l'operazione di somma. Le figure che seguono illustrano l'indicizzazione:



Il valore in D100 (8) viene sommato all'indirizzo del timer TS495. Il nuovo indirizzo diventa TS49D.

Il valore in D101 (5) viene sommato all'indirizzo del contatore CS270. Il nuovo indirizzo diventa CS275.

Il valore in D102 (2) viene sommato agli indirizzi degli ingressi X1 e X19. I nuovi indirizzi diventano X3 e X1B.

Il valore in D103 (10) viene sommato agli indirizzi delle uscite Y24 e Y40. I nuovi indirizzi diventano Y2E e Y4A.

Il valore in D104 (16) viene sommato agli indirizzi dei relé interni M6 e M62. I nuovi indirizzi diventano M16 e M72.

Il valore in D106 (16) viene sommato agli indirizzi del relé di collegamento B20. Il nuovo indirizzo diventa B30.

Il valore in D108 (1) viene sommato all'indirizzo del registro D0. Il nuovo indirizzo diventa D1.

Le istruzioni PLS, PLF e impulsive che sono eseguite una sola volta con l'attivazione della condizione di ingresso, non possono essere soggette a indicizzazione tramite le istruzioni IX/IXEND.

Nei casi in cui i nuovi indirizzi risultanti dalla somma superino i campi operandi ammessi, l'istruzione non può essere elaborata correttamente.

Anche se le istruzioni IX e IXEND vengono eseguite durante una modifica online fra due scansioni di programma (modifica in modo RUN), l'istruzione non può essere elaborata.

I valori sommati agli indirizzi di operandi a word, in cui è possibile accedere a ciascun bit, vengono memorizzati come dati binari. Gli indirizzi iniziali degli operandi a cui si riferiscono questi valori, sono contenuti in s.

In un programma, fra le istruzioni IX e IXEND, non è possibile eseguire nessuna indicizzazione.

Quando un programma viene espanso, gli indirizzi indicizzati degli operandi di una sezione di programma posta fra le istruzioni IX e IXEND vengono trasformati in indirizzi che utilizzano i registri indice (Zn). L'assegnazione di indirizzi indicizzati ai corrispondenti registri indice è mostrata di seguito:

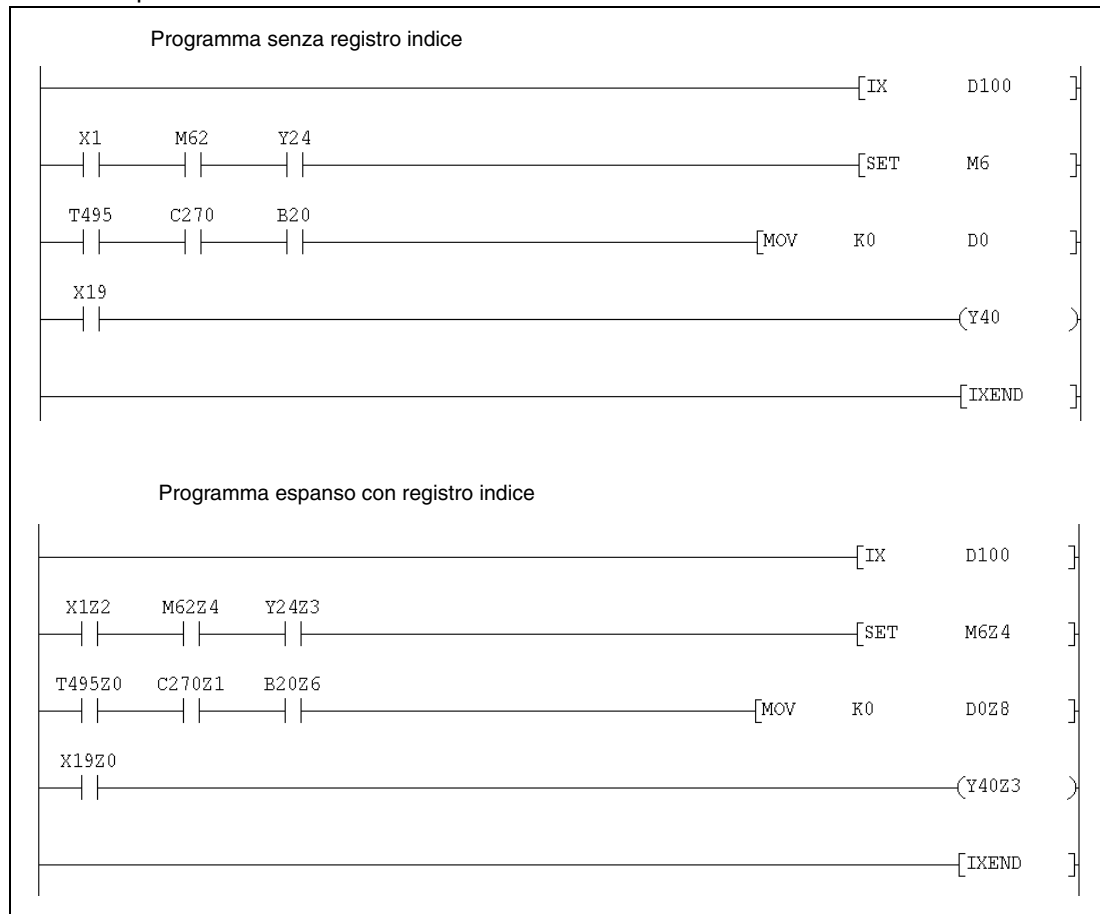
s	Operando	Indice file	s	Operando	Indice file
s	Valore indicizzazione timer (T)	Z0	s+8	Valore indicizzazione registri dati (D)	Z8
s+1	Valore indicizzazione contatori (C)	Z1	s+9	Valore indicizzazione registri collegamento W	Z9
s+2	Valore indicizzazione ingressi (X)	Z2	s+10	Valore indicizzazione file registri (R)	Z10
s+3	Valore indicizzazione uscite (Y)	Z3	s+11	Valore indicizzazione registri buffer I/O (U)	Z11
s+4	Valore indicizzazione relé interni (M)	Z4	s+12	Valore indicizzazione registri buffer (G)	Z12
s+5	Valore indicizzazione relé retentivi (L)	Z5	s+13	Valore indicizzazione numeri di rete di operandi di collegamento con accesso diretto (J)	Z13
s+6	Valore indicizzazione relé collegamento (B)	Z6	s+14	Valore indicizzazione file registri (ZR)	Z14
s+7	Valore indicizzazione relé impulsivi (V)	Z7	s+15	Valore indicizzazione puntatore (label)	Z15

I registri indice da Z10 a Z15 non sono disponibili nelle CPU Q00JCPU, Q00CPU, e Q01CPU.

A seconda del software di programmazione utilizzato, l'utente deve sommare manualmente il registro indice nel programma fra le istruzioni IX e IXEND.

Esempio

GX Developer



Il registro indice usato fra le istruzioni IX e IXEND (da Z0 a Z15) non influenza i registri indice usati altrove da altre istruzioni del programma.

NOTA

Per indicizzare sezioni di programma, i dispositivi periferici devono essere avviati in modo general purpose e deve essere eseguita l'espansione del programma (solo serie Q).

Se i dispositivi periferici vengono inizializzati da una CPU Q2A, Q2A-S1, Q3A o Q4A e si esegue l'indicizzazione di una sezione di programma tra le istruzioni IX e IXEND, non è possibile una esecuzione corretta del programma.

Se si usano le istruzioni IX e IXEND sia in un normale programma di sequenza che in un programma a interruzione, si deve prevedere un interblocco per evitare l'esecuzione simultanea. Disabilitare le interruzioni fra le istruzioni IX e IXEND.

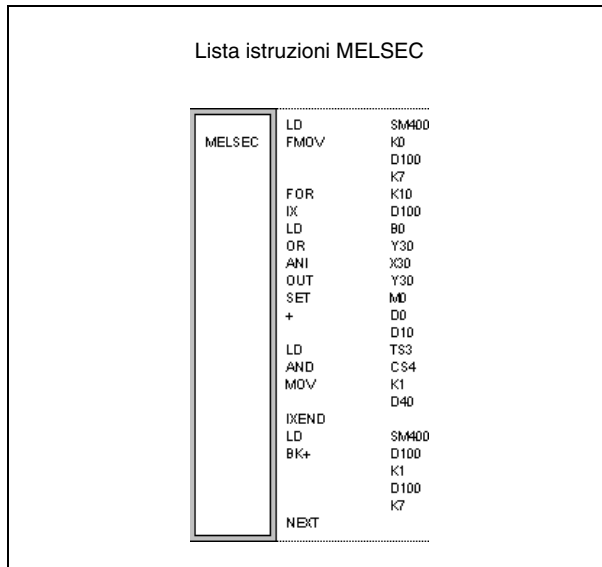
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Le istruzioni IX e IXEND non sono programmate in coppia (codice di errore 4231).
- Dopo l'esecuzione di una istruzione IX, viene eseguita una istruzione END, FEND, GOEND, o STOP prima dell'istruzione IXEND (codice di errore 4231).

Esempio di programma IX, IXEND

Il programma seguente elabora per 10 volte il loop di programma fra le istruzioni IX e IXEND. In ogni loop gli indirizzi degli operandi programmati all'interno del loop vengono incrementati di 1. La tabella sottostante mostra i registri che contengono i valori degli operandi corrispondenti da sommare. Sono inoltre mostrate le variazioni degli indirizzi operandi per il primo, secondo e decimo loop.



D	Operando	Modifica indirizzo operando / loop			
		1.	2.	3.	10.
D100	Indicizzazione valore di un timer (T)	T3	T4	T5	TC
D101	Indicizzazione valore di un contatore (C)	C4	C5	C6	CD
D102	Indicizzazione valore di un ingresso (X)	X10	X11	X12	X19
D103	Indicizzazione valore di una uscita (Y)	Y30	Y31	Y32	Y39
D104	Valore indicizzazione relé interni (M)	M0	M1	M2	M9
D106	Valore indicizzazione relé collegamento (B)	B0	B1	B2	B9
D108	Valore indicizzazione registri dati (D)	D0	D1	D2	D9
		D10	D11	D12	D19
		D40	D41	D42	D49

7.6.11 IXDEV, IXSET

CPU

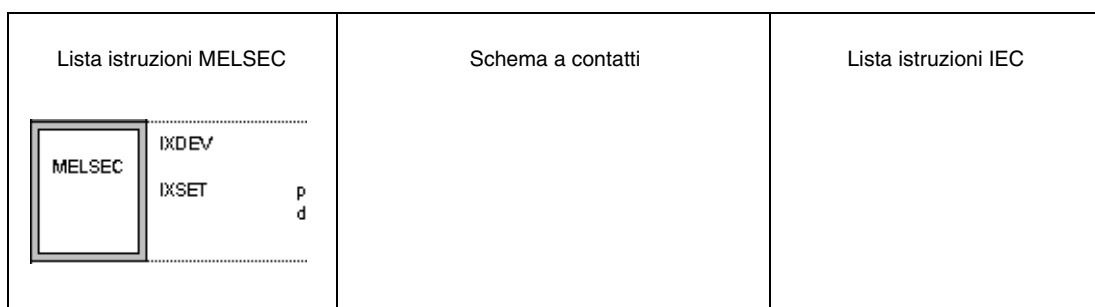
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi MELSEC Q

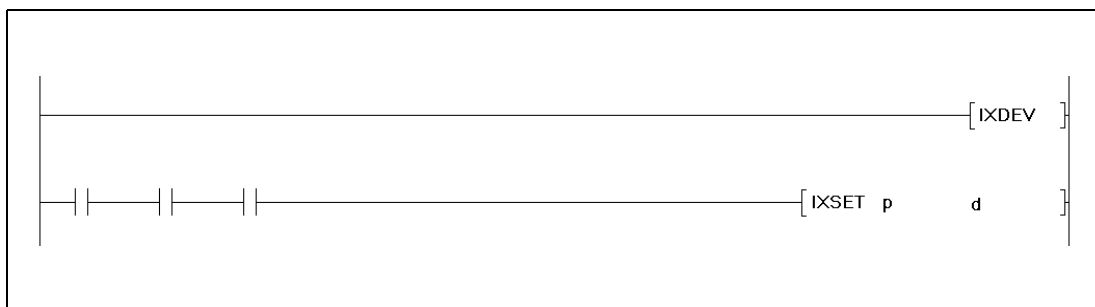
	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro P
	Bit	Word		Bit	Word						
p	—	—	—	—	—	—	—	—	●	SM0	1/3 ● ¹
d	—	●	●	—	—	—	—	—	—		

¹ L'istruzione IXDEV richiede un passo; l'istruzione IXSET richiede tre passi.

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
p	Indirizzo del primo operando (solo puntatore/label) che contiene i dati per l'indicizzazione.	Puntatore/Label
d	Primo indirizzo del dispositivo contenente l'indirizzo operando indicizzato.	BIN 16-bit

Funzioni **Memorizzazione indirizzi operandi in una tabella di indicizzazione****IXDEV/IXSET Istruzione per la compilazione di una tabella di indicizzazione**

Le istruzioni IXDEV e IXSET sono supportate solo in GX Developer o nel modo MELSEC di GX IEC Developer.

Le istruzioni IXDEV e IXSET leggono gli indirizzi degli operandi nell'area di definizione offset e scrivono questi numeri di offset in una tabella di indicizzazione nell'operando indicato da d.

Fare riferimento alle istruzioni IX e IXEND per l'assegnazione dei tipi di operando ai registri corrispondenti.

Se un tipo di operando non è stato assegnato nella indicazione dell'offset, in tabella viene memorizzato il valore 0.

I singoli bit di dispositivi a word vengono processati come contatti dummy, cioè solo l'indirizzo di un singolo bit può essere letto e scritto nella tabella indice. Per indirizzare il dummy, si specifica il bit corrispondente. Il bit 0 (b0) del registro dati D0 viene indirizzato come D0.0. Per l'indicazione di un bit in una word dati a 16 bit, vengono utilizzati i valori esadecimali da 0 a F.

La lettura del valore degli offset avviene come segue:

- Lettura degli operandi: T□, C□, X□, Y□, M□, L□, V□, B□
Il valore dell'offset indicato □ viene letto e scritto nel registro corrispondente..
- Lettura degli operandi: D□.XX, W□.XX, R□.XX¹, U□\G□.XX¹, ZR□.XX¹
Il valore dell'offset indicato □ viene letto e scritto nel registro corrispondente..
I valori indicati come XX indicano la variabile per l'indicazione del bit.
¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU
- Lettura degli operandi: J□/B□¹, J□/W□¹, J□/X□¹, J□/Y□¹
Il valore dell'offset indicato □ viene letto e scritto nel registro corrispondente..
Se non si devono scrivere valori di offset per l'operando successivo a J□/, questo valore deve essere impostato a 0.
¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU
- Programmando l'istruzione IXSET il valore offset dell'operando P□ è indicato direttamente per indirizzo (puntatore/label).

Se vengono specificati due tipi di operandi identici nella zona di definizione dell'offset, vale il valore di offset dell'ultimo operando.

Le istruzioni IXDEV e IXSET devono essere programmate a coppie.

Il valore di offset dell'operando ZR□.XX può variare da 0 a 32767. Il valore di offset è il resto del quoziente dell'indirizzo dell'operando diviso per 32767, e viene scritto nel registro corrispondente.

Per i contatti dummy nella zona di indicazione dell'offset, sono valide solo le istruzioni LD e AND. Tutte le altre istruzioni vengono ignorate.

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Le istruzioni IXDEV e IXSET non sono programmate in coppia (codice di errore 4231).

Esempio di programma IXDEV, IXSET

Il programma seguente scrive gli indirizzi (valori offset) dei contatti dummy nella zona di definizione offset, nei registri corrispondenti. Il valore di offset del puntatore/label viene specificato dall'istruzione IXSET. Fare riferimento alle istruzioni IX e IXEND per l'assegnazione dei tipi di operando ai registri corrispondenti.

Lista istruzioni MELSEC

MELSEC	IXDEV	%5
	LD	Y13
	AND	T#4
	AND	D#3
	IXSET	P5
		D0

D0	4	T
	0	C
	5	X
	3	Y
	0	M
	0	L
	0	B
	0	V
	8	D
	0	
	:	
	:	
	0	
D15	5	

7.7 Istruzioni per operazioni su tabelle dati

Le istruzioni per operazioni su dati tabellari scrivono e leggono dati da e per tabelle dati. I dati attuali vengono scritti nella tabella e letti in ordine diverso per ulteriore elaborazione. Inoltre queste istruzioni consentono di cancellare e inserire blocchi dati specifici.

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Scrittura dati in una tabella dati	FIFW	FIFW_M
	FIFWP	FIFWP_M
Leggi i primi dati inseriti da una tabella	FIFR	FIFR_M
	FIFRP	FIFRP_M
Leggi gli ultimi dati inseriti da una tabella	FPOP	FPOP_M
	FPOPP	FPOPP_M
Cancella i blocchi dati specificati da una tabella	FDEL	FDEL_M
	FDELP	FDELP_M
Inserisci i blocchi dati specificati nella tabella	FINS	FINS_M
	FINSP	FINSP_M

7.7.1 FIFW, FIFWP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

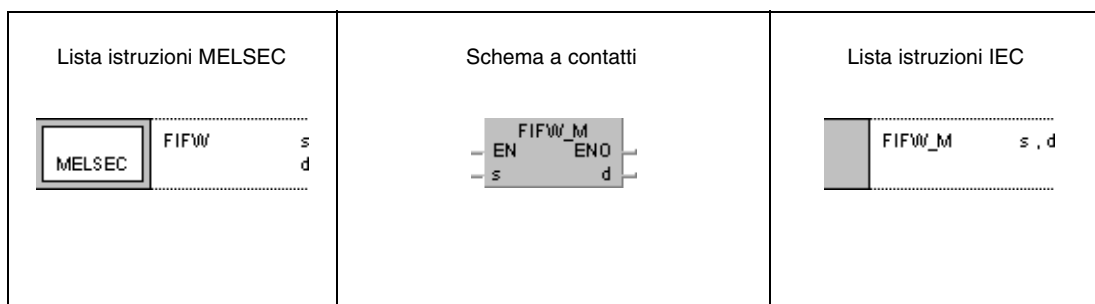
	Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Riporto	Errore				
	Operandi a bit						Operandi a word (16-bit)						Costanti		Puntatore							Livello			
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V						K	H (16#)	P	I
s	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●							●
d							●	●	●	●	●														●

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

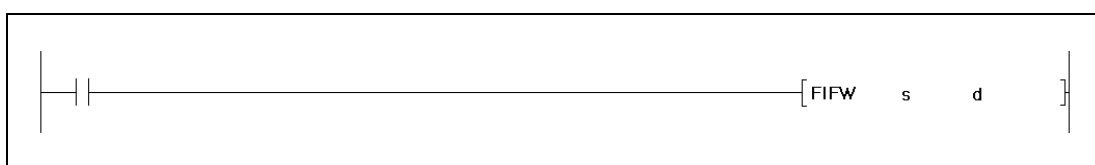
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J		Modulo funzione speciale U	Registro indice Zn	Costanti			Altro
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	—	—	SM0	3	
d	—	●	●	—	—	—	—	—			

GX IEC Developer



GX Developer

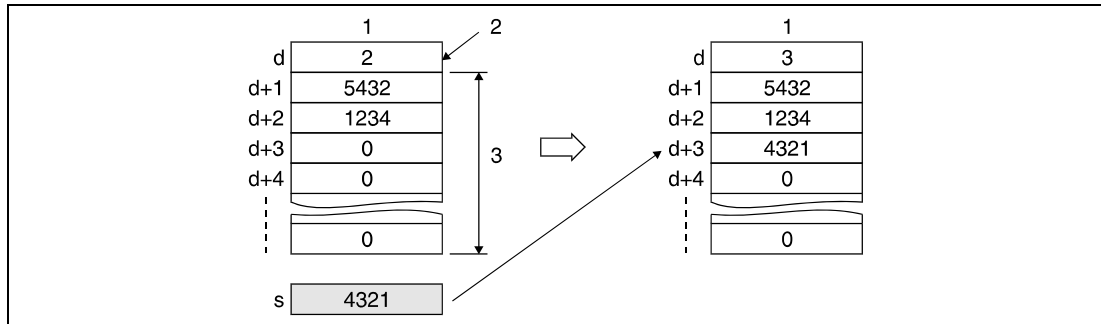


Variabili

Dati impostati	Significato	Tipo dati
s	Dati da scrivere nella tabella o operandi che contengono questi dati.	BIN 16-bit
d	Primo indirizzo della tabella.	

Funzioni **Scrittura dati in una tabella dati****FIFW** **Istruzione per inserimento dati**

L'istruzione FIFW scrive in dati in sequenza specificati da s in una tabella dati. La tabella è specificata dal campo di indirizzi definito da d e contiene i dati secondo la sequenza di inserimento. Il numero totale di record dati è contenuto nel primo indirizzo del campo dati indicato da d. Quindi il dato contenuto in questo indirizzo è il puntatore di posizione usato per scrivere dati nella tabella. All'atto dell'esecuzione dell'istruzione FIFW questo valore viene incrementato di 1. Quindi il dato successivo viene memorizzato a partire dall'indirizzo d+1.



¹ Tabella dati

² Puntatore posizione

³ Campo tabella dati

Prima della prima istruzione FIFW il contenuto dell'indirizzo specificato da d deve essere azzerato.

Il numero di record dati da memorizzare e il campo di indirizzi della tabella devono essere controllati dal programma utente.

Per la gestione di diversi record dati in tabelle differenti, dovrebbe essere utilizzato un programma applicativo.

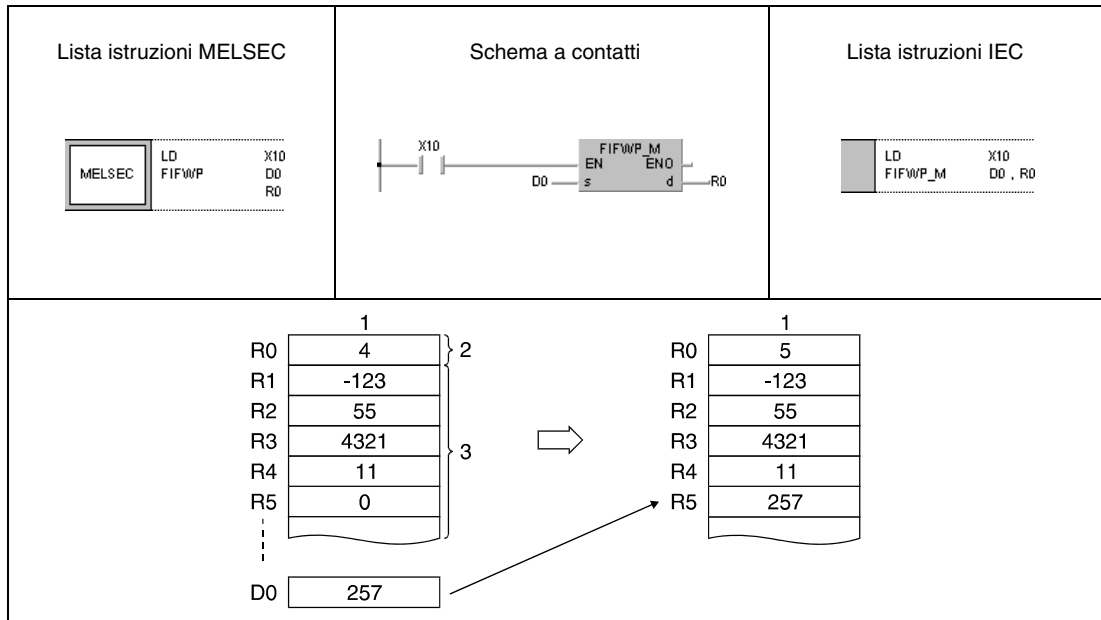
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il campo della tabella dati del FIFO supera il campo operandi ammesso quando si esegue l'istruzione FIFW (serie Q e System Q = codice di errore 4101)

Programma di esempio 1 FIFWP

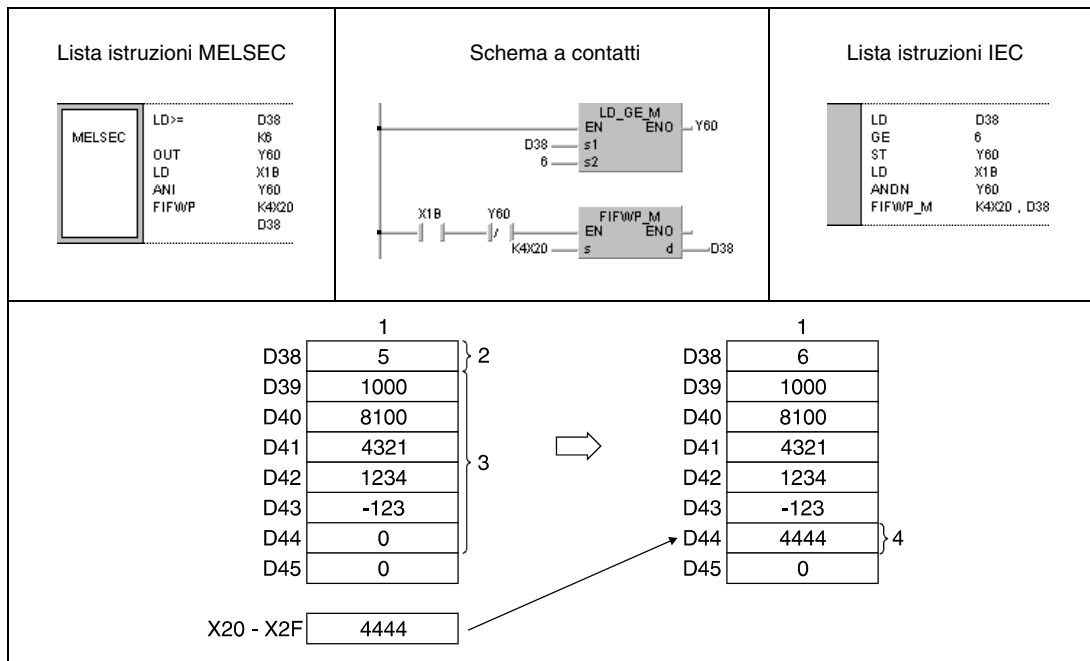
Il programma seguente specifica il campo di memorizzazione della tabella dati tramite i registri da R0 a R5. L'indirizzo di partenza del campo di memorizzazione (R0) contiene il puntatore di posizione, che indica il numero di record dati memorizzati. Con il fronte di salita di X10, il dato in D0 viene memorizzato nella prima posizione disponibile della tabella dati (R5).



- ¹ Tabella dati
- ² Puntatore posizione
- ³ Campo tabella dati

Programma di esempio 2 FIFWP

Il programma seguente specifica il campo di memorizzazione della tabella dati tramite i registri da D38 a D44. L'indirizzo di partenza del campo di memorizzazione (D38) contiene il puntatore di posizione, che indica il numero di record dati memorizzati. Con il fronte di salita di X1B, i dati degli ingressi da X20 a X2F vengono memorizzati nella prima posizione disponibile della tabella dati (D44). In questo caso la tabella dati contiene al massimo 6 record. Quindi Y60 viene programmato come limitatore per le istruzioni FIFW. L'uscita si attiva se il contenuto di D38 è superiore o uguale a 6.



- ¹ Tabella dati
- ² Puntatore posizione
- ³ Campo tabella dati
- ⁴ Ultimo indirizzo di memorizzazione disponibile

7.7.2 FIFR, FIFRP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

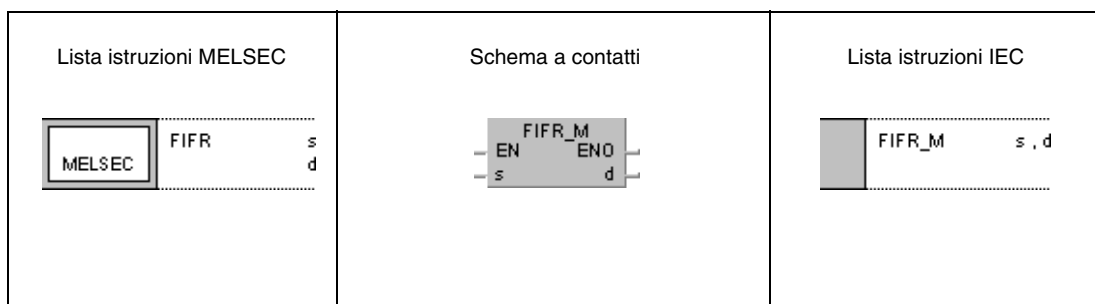
	Operandi utilizzabili															Definizione cifra Numero di passi	Indice	Riporto M9012	Errore M9010 M9011								
	Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello												
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z					V	K	H (16#)	P	I	N		
s	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●							K1 ↓ K4	7 ↓ 1	●		●	
d							●	●	●	●	●																●

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

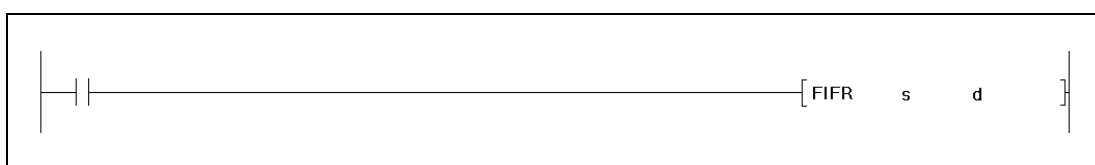
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti			Altro
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	●	—	—	SM0	3
d	—	●	●	—	—	—	—	—	—		

GX IEC Developer



GX Developer



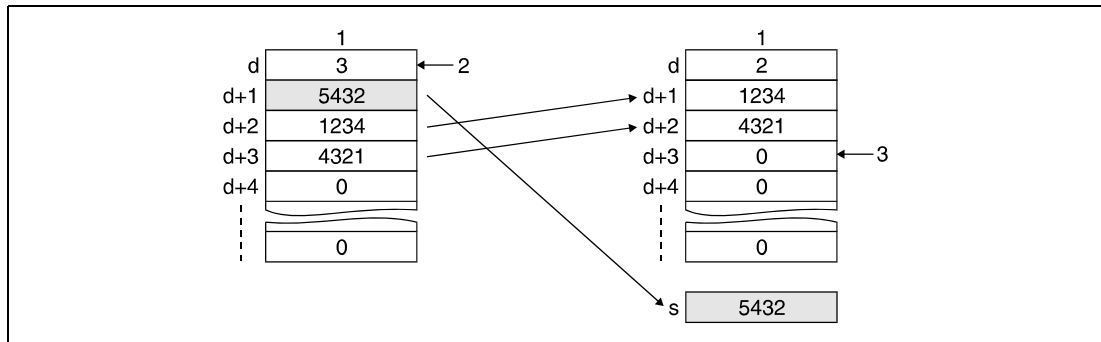
Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene i dati estratti.	BIN 16-bit
d	Primo indirizzo della tabella.	

Funzioni **Legge primo dato inserito da una tabella dati****FIFR** **Istruzione per la lettura del primo dato inserito**

L'istruzione FIFR legge dati da una tabella e li memorizza in un campo di indirizzi specificato. La lettura dei dati inizia dal primo indirizzo (d+1) dopo il puntatore di posizione. I dati vengono trasferiti nel campo di memorizzazione specificato da s.

I dati successivi della tabella vengono fatti scorrere verso l'inizio della tabella, nell'ordine del loro inserimento. Tutti i dati precedenti vengono cancellati. Dopo la lettura, il valore del puntatore di posizione (indirizzo specificato da d) viene decrementato di 1.



¹ Tabella dati

² Puntatore posizione

³ Questo registro viene azzerato.

NOTA

Accertarsi che questa istruzione non venga eseguita, fino a quando d (puntatore di posizione) contiene il valore 0.

Errori di esecuzione

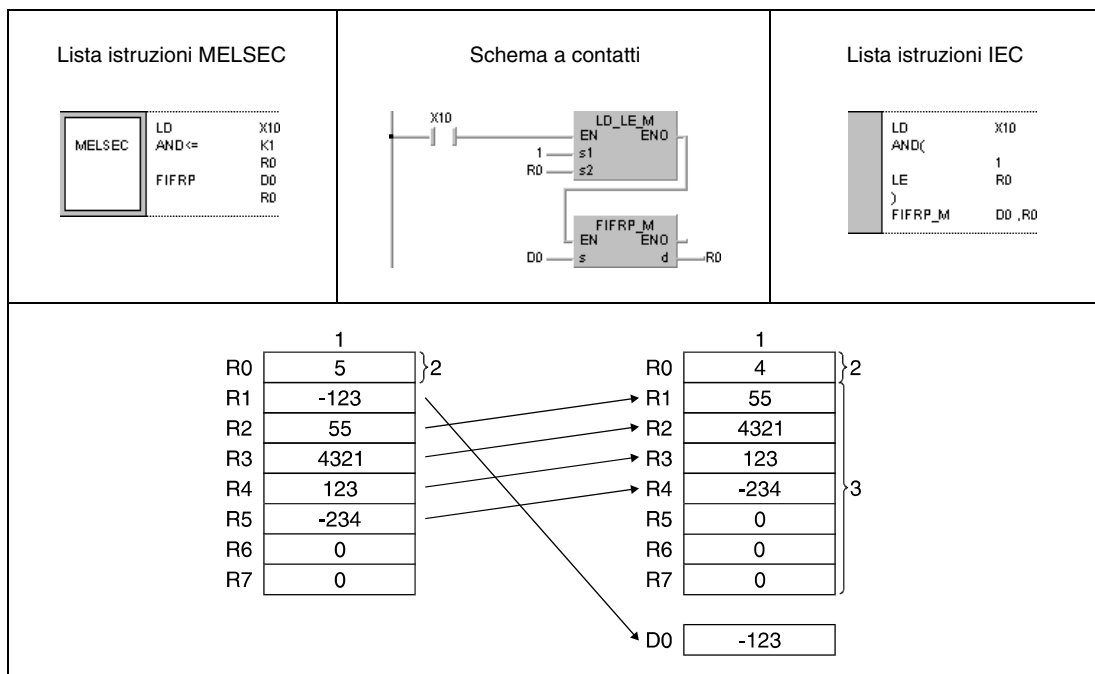
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Viene eseguita una istruzione FIFR mentre il puntatore di posizione contiene 0 (serie Q e System Q = codice di errore 4100).
- Il campo di operandi della tabella supera il campo operandi ammesso quando viene eseguita una istruzione FIFR (serie Q e System Q = codice di errore 4101).

Programma di esempio 1

FIFRP

Con il fronte di salita di X10, il programma seguente legge il valore contenuto in R1 (primo valore inserito) della tabella da R0 a R7 e memorizza il valore nel registro D0. All'inizio il valore del puntatore di posizione è 5, mentre diventa 4 dopo l'esecuzione. L'operazione di confronto precedente evita l'esecuzione dell'istruzione FIFR se il puntatore di posizione (R0) contiene il valore 0.

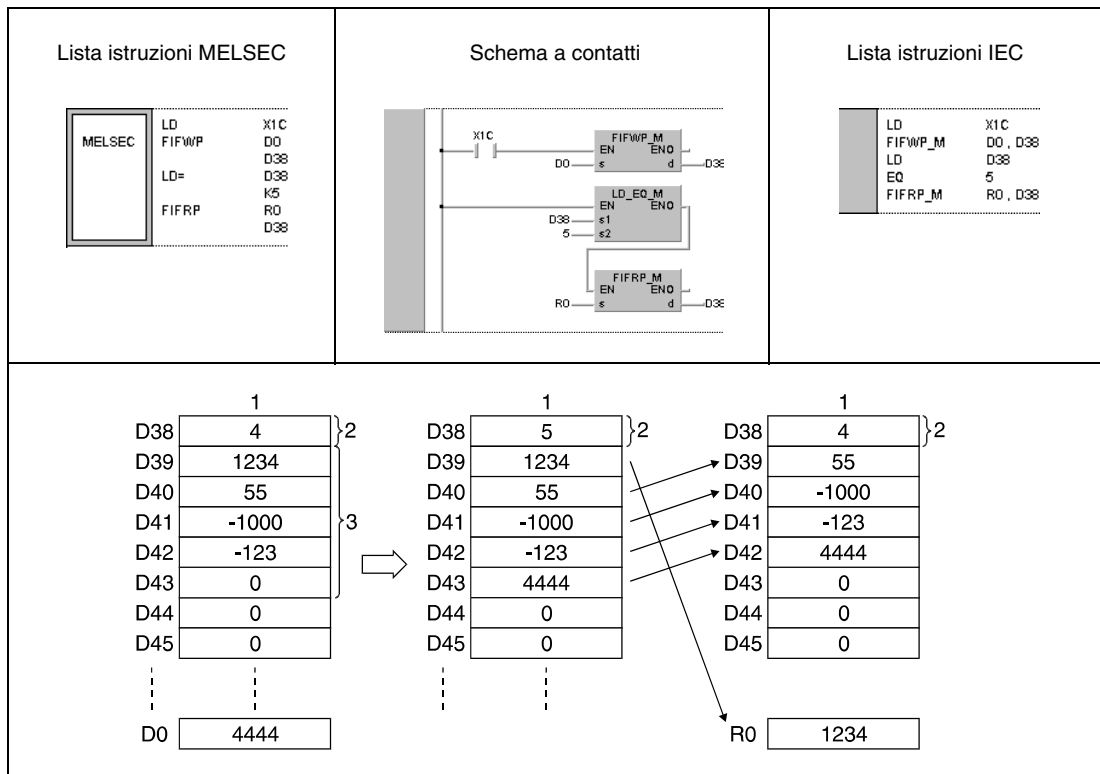


- ¹ Tabella dati
- ² Puntatore posizione
- ³ Campo tabella dati

Programma di esempio 2

FIFRP

Con il fronte positivo di X1C, il programma seguente scrive in valore in D0 nella tabella dati da D38 a D43. Se il valore del puntatore di posizione è 5, viene letto il primo valore della tabella FIFO e memorizzato in R0. Questo processo viene ripetuto ad ogni fronte di salita di X1C.



¹ Tabella dati

² Puntatore posizione

³ Campo tabella dati

7.7.3 FPOP, FPOPP

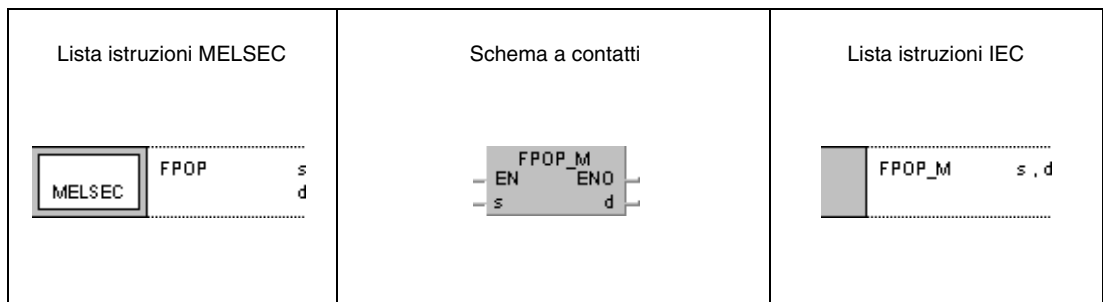
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

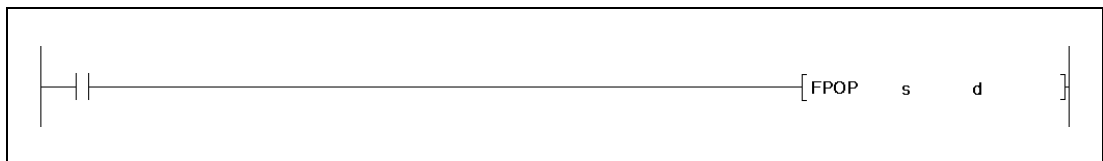
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti			Altro
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	●	—	SM0	3	
d	—	●	●	—	—	—	—	—			

GX IEC Developer



GX Developer



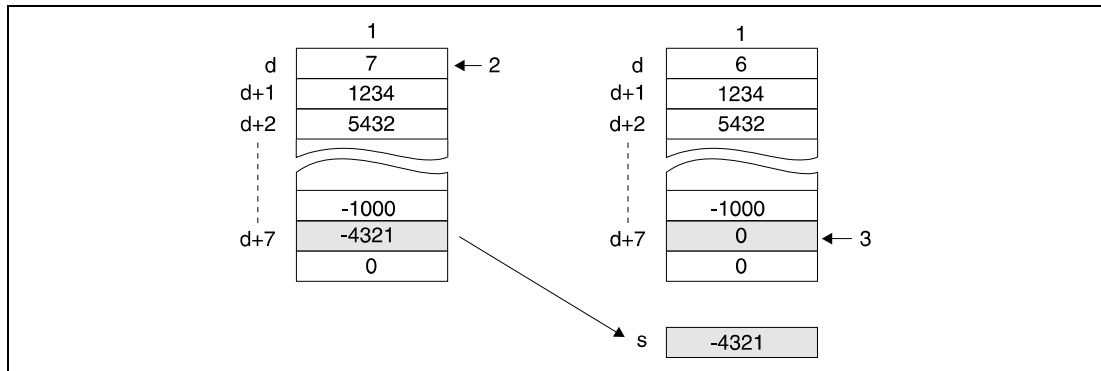
Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che memorizza il dato letto.	BIN 16-bit
d	Primo indirizzo della tabella.	

Funzioni **Letture ultimo dato inserito da una tabella dati****FPOP** **Istruzione per la lettura dell'ultimo dato inserito**

L'istruzione FPOP legge dati da una tabella e li memorizza in un campo di indirizzi specificato. La lettura dei dati inizia dall'ultimo indirizzo $d+n$ della tabella dati. I dati vengono trasferiti nel campo di memorizzazione specificato da s .

Il dato corrispondente all'indirizzo letto viene azzerato nella tabella. Dopo la lettura, il valore del puntatore di posizione (primo indirizzo di d) viene decrementato di 1.



¹ Tabella dati

² Puntatore posizione

³ Questo registro viene azzerato.

NOTA

Accertarsi che questa istruzione non venga eseguita, fino a quando d (puntatore di posizione) contiene il valore 0.

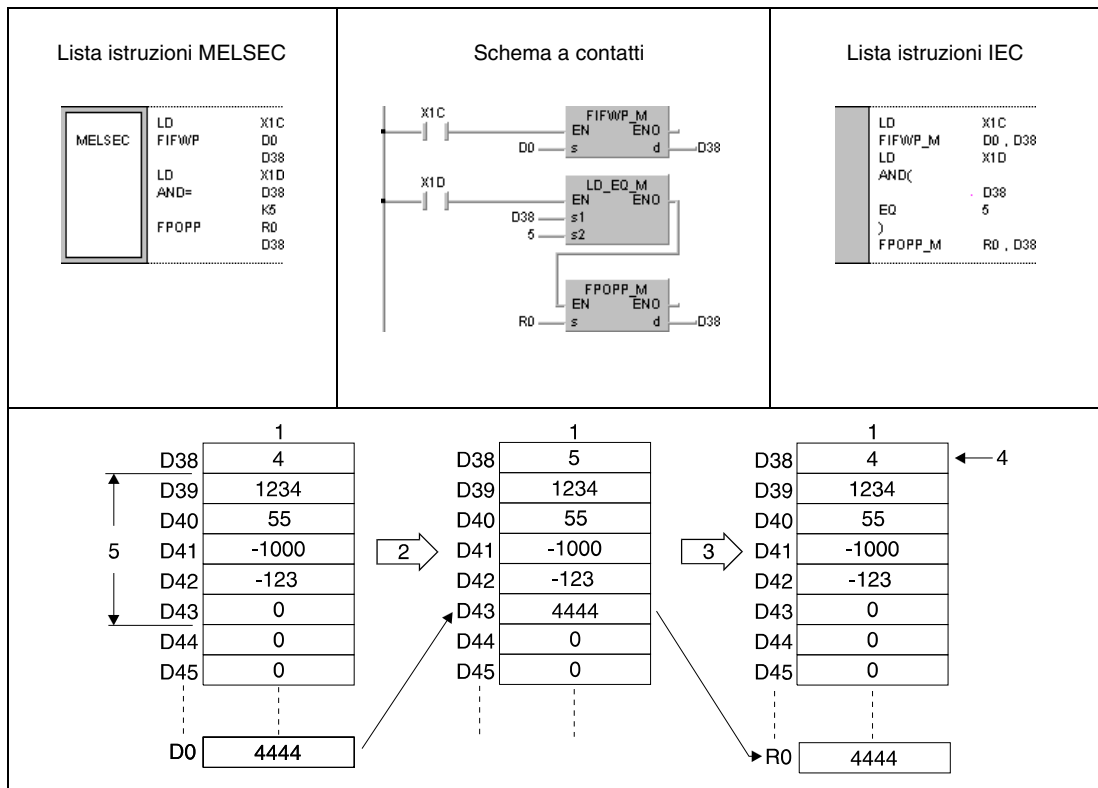
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Viene eseguita una istruzione FPOP mentre il puntatore di posizione contiene 0 (codice di errore 4100)
- Il campo della tabella dati supera il campo operandi ammesso durante l'esecuzione di una istruzione FPOP (codice di errore 4101).

**Programma
di esempio 2** FPOPP

Con il fronte positivo di X1C, il programma seguente scrive in valore in D0 nella tabella dati da D38 a D43. Se il valore del puntatore di posizione è 5, con il fronte positivo di X1D viene letto il valore del registro D43 e memorizzato in R0.



¹ Tabella dati

² Fronte positivo di X1C

³ Fronte positivo di X1D

⁴ Puntatore posizione

⁵ Campo indirizzi attuale della tabella

7.7.4 FDEL, FDELP, FINS, FINSP

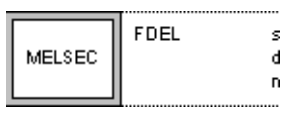
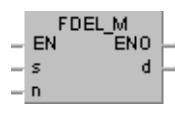
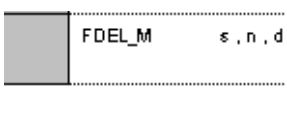
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

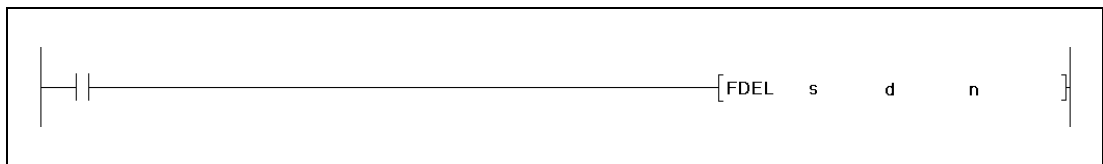
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	●	—	—	SM0	4
d	—	●	●	—	—	—	—	—	—		
n	●	●	●	●	●	●	●	●	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
--	--	---

GX Developer



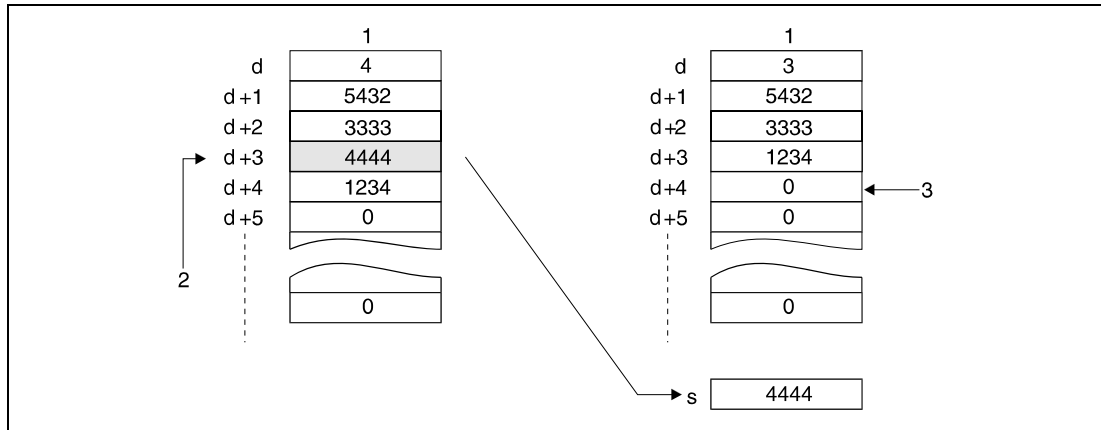
Variabili

Dati impostati	Significato	Tipo dati
s	Dato da inserire nella tabella ad un indirizzo specificato o operando che contiene questo dato. Indirizzo del primo operando che contiene i dati da cancellare dalla tabella ad un indirizzo specificato.	BIN 16-bit
d	Primo indirizzo della tabella.	
n	Indirizzo in cui inserire o cancellare i dati.	

Funzioni Cancellazione e inserimento di blocchi dati specificati in una tabella**FEDL Cancelli i blocchi dati specificati**

L'istruzione FDEL cancella il blocco dati n dopo il puntatore di posizione dalla tabella specificata da d e memorizza questo valore nell'operando indicato da s.

I dati della tabella vengono compattati dopo la cancellazione di un blocco dati. Dopo la lettura, il valore del puntatore di posizione (indirizzo specificato da d) viene decrementato di 1.



¹ Tabella dati

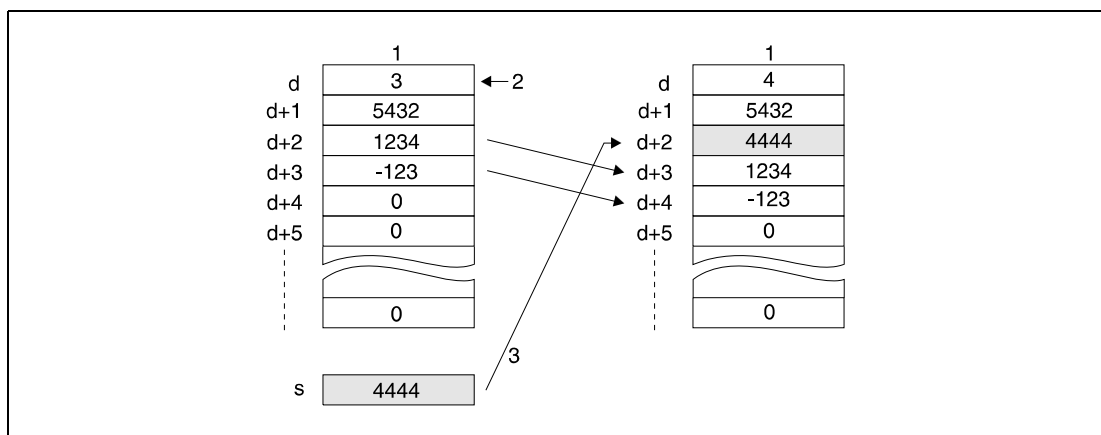
² Con n=3 viene cancellato il blocco dati d+3.

³ Questo registro viene azzerato.

FINS/FINSP Inserimento di blocchi dati specificati

L'istruzione FINS inserisce il blocco dati a 16 bit indicato da s nella posizione n dopo il puntatore di posizione, nella tabella specificata da d.

I blocchi dati successivi alla posizione di inserimento vengono fatti scorrere di un indirizzo. Dopo l'inserimento, il valore del puntatore di posizione (indirizzo specificato da d) viene incrementato di 1.



¹ Tabella dati

² Puntatore posizione

³ Con n=2 il blocco dati viene inserito in d+2

Errori di esecuzione

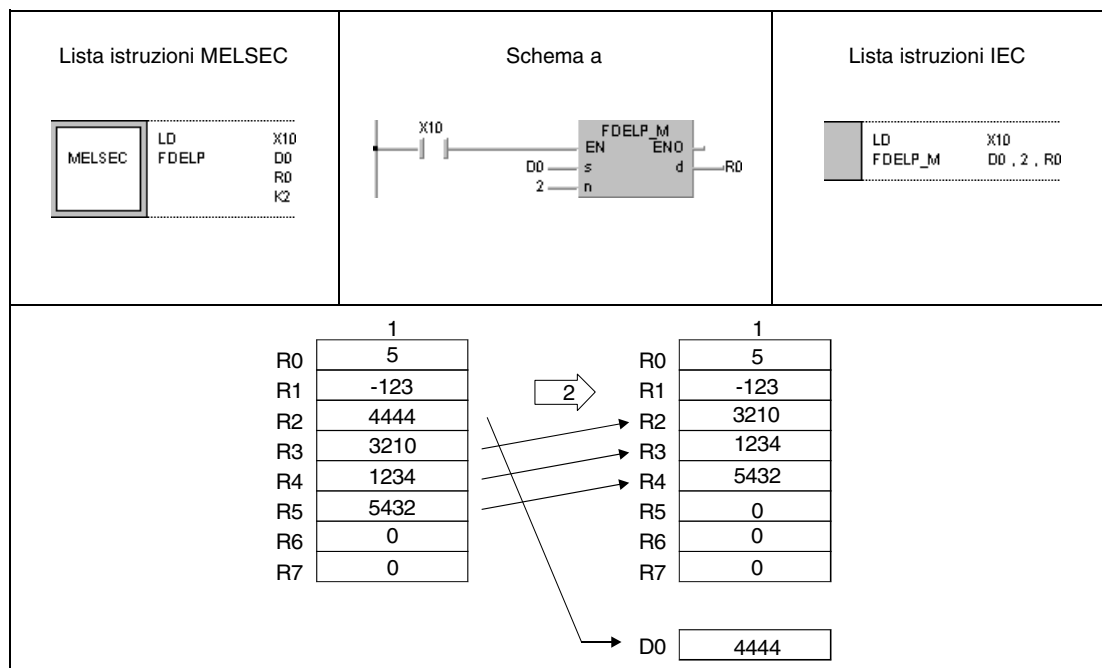
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- La posizione di inserimento in d specificata da n con l'istruzione FINS supera il campo di indirizzi del blocco dati esistente aumentato di 1 (codice di errore 4101).
- Il valore di n supera il campo di operandi della tabella d (codice di errore 4101).
- Le istruzioni FDEL o FINS sono state eseguite con n = 0 (codice di errore 4100).
- L'istruzione FDEL è stata eseguita con il valore di d = 0 (codice di errore 4100).
- Il campo della tabella dati supera il campo operandi ammesso durante l'esecuzione di una istruzione FDEL o FINS (codice di errore 4100).

Programma di esempio 1

FDELP

QUando X10 si attiva, viene cancellato il dato della seconda posizione (R2) della tabella dati da R0 a R7, e il dato viene memorizzato in D0.

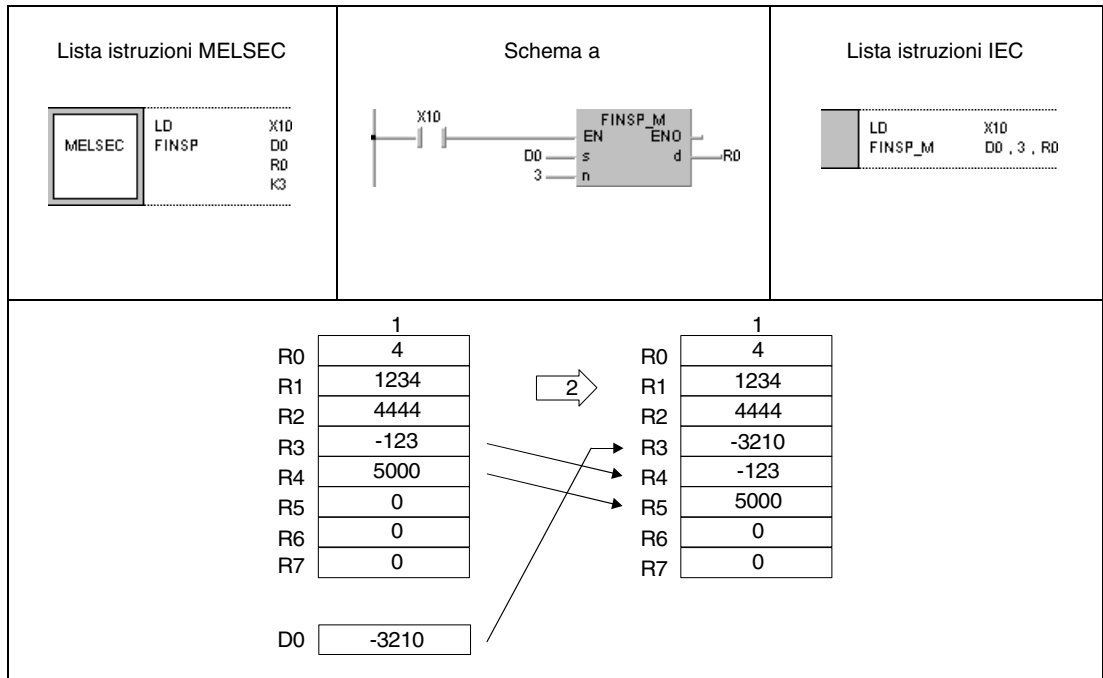


¹ Tabella dati

² Fronte positivo di X10

Programma di esempio 2 FINSP

Il programma seguente inserisce il dato in D0 nella terza posizione della tabella da R0 a R7, quando X10 diventa attivo.



¹ Tabella dati

² Fronte positivo di X10

7.8 Istruzioni di accesso a buffer di memoria

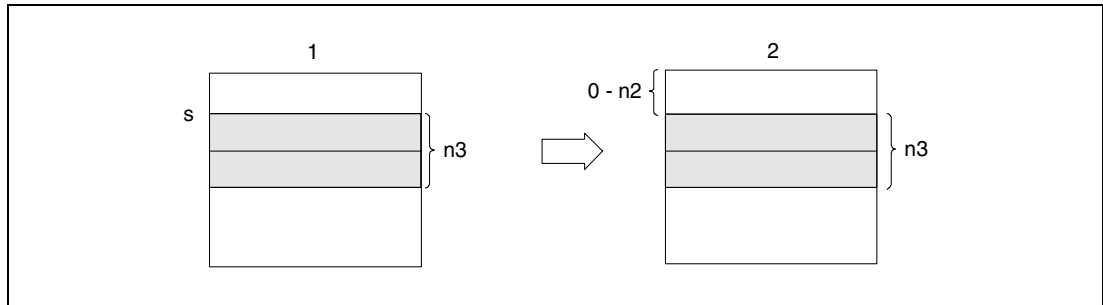
Le istruzioni seguenti consentono di accedere ai buffer di memoria dei moduli funzione speciali. Queste istruzioni consentono di scambiare dati fra la CPU e i moduli interessati. La tabella che segue offre una panoramica di queste istruzioni:

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Lettura dati da modulo funzione speciale	FROM	FROM_M
	FROMP	FROMP_M
	DFRO	DFRO_M
	DFROP	DFROP_M
Scrittura dati su modulo funzione speciale	TO	TO_M
	TOP	TOP_M
	DTO	DTO_M
	DTOP	DTOP_M

Funzioni **Letture di dati a 1 o 2 word da un modulo funzione speciale**

FROM Lettura dati a 1 word (16-bit)

L'istruzione FROM legge 1 word dal buffer di memoria di un modulo funzione speciale e lo trasferisce nell'indirizzo di memoria specificato della CPU. L'indirizzo del dato da leggere è indicato da n2, mentre il numero di word è indicato da n3 e l'indirizzo di base del modulo funzione speciale, determinato dalla posizione del modulo nel telaio base, è indicato da n1. L'indirizzo dell'area di memoria interna alla CPU in cui trasferire il dato è indicato da d.

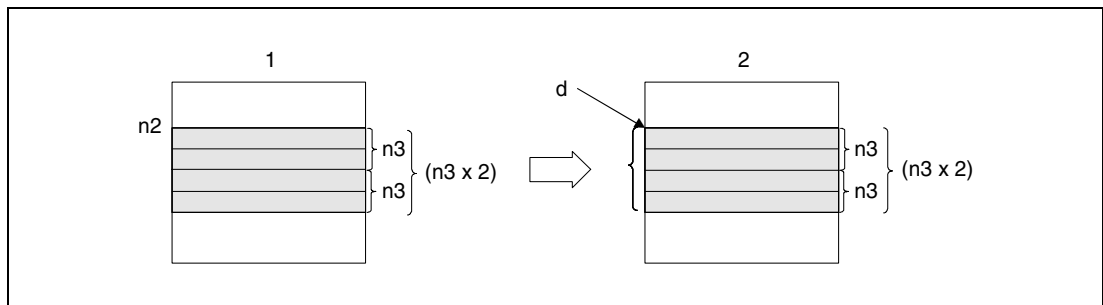


¹ Buffer di memoria del modulo funzione speciale
² Memoria della CPU

NOTA *L'istruzione FROM può anche essere usata per leggere dalla memoria condivisa di un'altra stazione, in un sistema multi CPU. Vedere il capitolo 9.6.2 per ulteriori dettagli.*

DFRO Lettura dati a 2 word (32-bit)

L'istruzione DFRO legge 2 word dati dal buffer di memoria di un modulo funzione speciale. L'indirizzo del primo dato da leggere è indicato da n2, il numero di word dati (multiplo di 2) è indicato da n3, e l'indirizzo di base del modulo funzione speciale è indicato da n1. L'indirizzo dell'area di memoria interna alla CPU in cui trasferire il dato è indicato da d.



¹ Buffer di memoria del modulo funzione speciale
² Memoria della CPU

NOTA *Una CPU della serie Q o System Q può anche accedere direttamente al buffer di memoria del modulo funzione speciale. In questo caso gli operandi sono specificati come U□\G□ (U(Indirizzo base del modulo funzione)/G(Indirizzo del buffer di memoria)).*

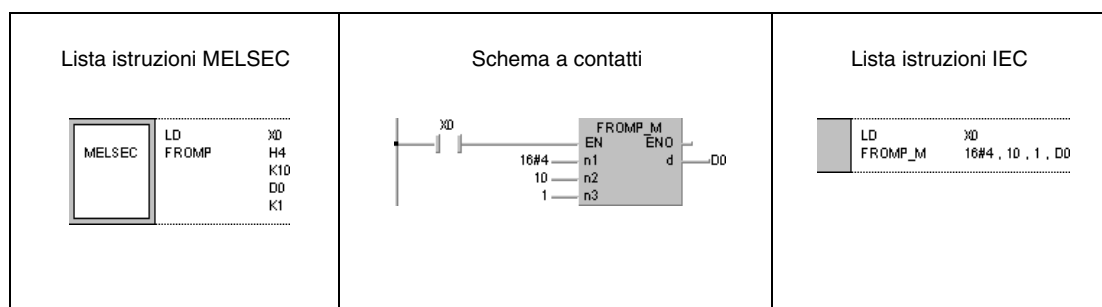
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

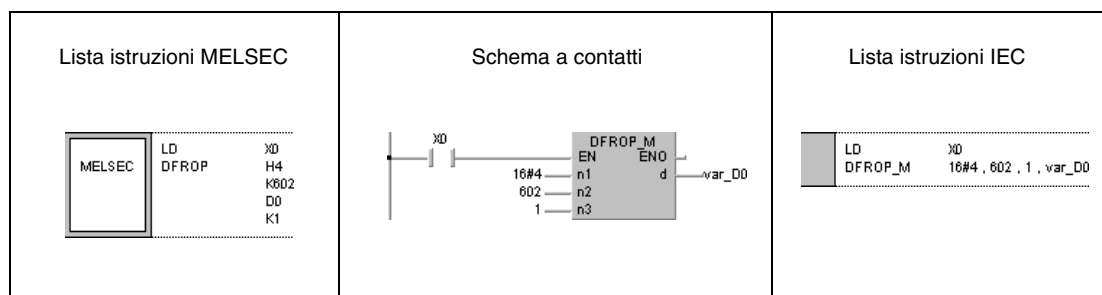
- Nessuno scambio dati con il modulo funzione speciale prima dell'esecuzione dell'istruzione (codice di errore 1412).
- Si è verificato un errore nel modulo prima dell'esecuzione dell'istruzione (codice di errore 1402).
- L'indirizzo di I/O indicato da n2 non si riferisce a un modulo funzione speciale. (serie Q e System Q = codice di errore 2110).
- Il numero di word dati indicato da n3 supera i numeri dispositivo indicato d (serie Q e System Q = codice di errore: 4101).
- L'indirizzo definito da n2 supera il campo del buffer di memoria (serie Q e System Q = codice di errore 4100).
- L'indirizzo specificato da n2 non è corretto (AJ71QC24) (serie Q e System Q = codice di errore 4100).
- L'accesso al modulo funzione speciale non è consentito.

Programma di esempio 1**FROMP**

Con il fronte positivo di X0, il programma seguente legge i valori digitali del canale CH1 dall'indirizzo 10 del buffer di memoria di un modulo A68AD. L'indirizzo dell'area di memoria del modulo è da 040 a 05F. Il dato letto è memorizzato in D0.

**Programma di esempio 2****DFROP**

Con il fronte positivo di X0, il programma seguente legge i dati dell'asse X dagli indirizzi 602 e 603 del buffer di memoria di un modulo AD71. L'indirizzo dell'area di memoria del modulo è da 040 a 05F. Il dato letto è memorizzato in D0 e D1.

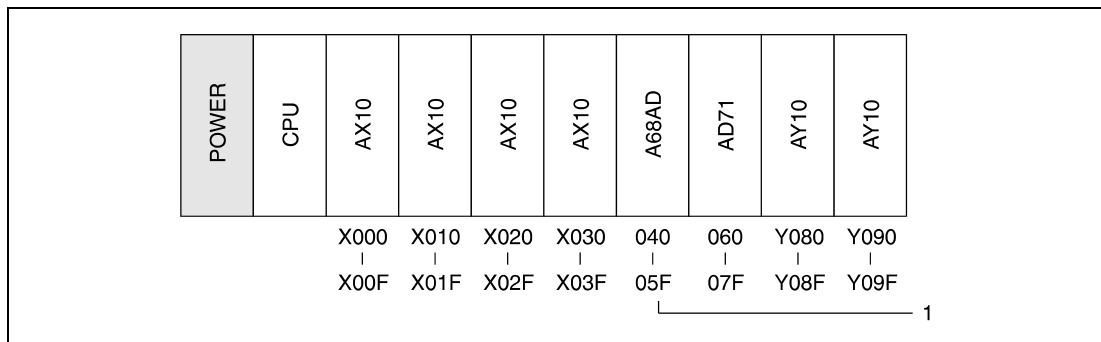


NOTA

L'indirizzo di base n1 deve essere specificato come segue:

n1 = 10 → indirizzo base = 1

n1 = 20 → indirizzo base = 2



¹ Indirizzo base del registro speciale: n1 = K4 o H4

Il programma di esempio 2 non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.8.2 TO, DTO, DTO, DTO

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

	Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Riporto errore M9012	Segnal e di errore M9010 M9011						
	Operandi a bit						Operandi a word (16-bit)						Costanti		Punta- tore							Livello					
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V							K	H (16#)	P	I	N
n1																	●	●					K1 ↓ K4				
n2																	●	●					● ²	9/11	●		
s	●	●	●	●	●	●	●	●	●	●	●						● ¹	● ¹					K1 ↓ K8	● ³			●
n3																	●	●									

¹ lo campo di definizione delle costanti è: da H0 a HFFFF, da K-32768 a K32767.

² La lunghezza del blocco può essere specificata da K1 a K4 con una istruzione TO(P) e da K1 a K8 con una istruzione DTO(P).

³ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

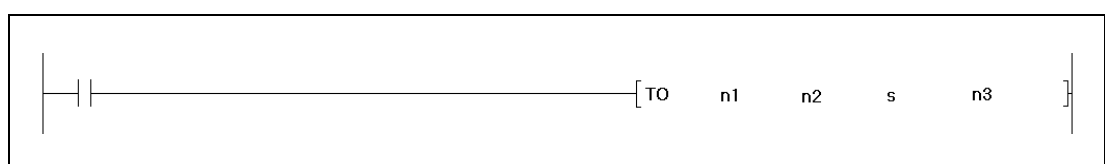
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro U
	Bit	Word		Bit	Word						
n1	●	●	●	●	●	●	●	●	●	SM0	5
n2	●	●	●	●	●	●	●	●			
s	●	●	●	—	—	—	—	●			
n3	●	●	●	●	●	●	●	●			

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
<div style="border: 1px solid black; padding: 5px; display: inline-block;">MELSEC</div> TO n1 n2 s n3		TO_M s , n1 , n2 , n3

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
n1	Indirizzo iniziale del modulo funzione speciale sull'unità base.	BIN 16-bit
n2	Primo indirizzo dell'area di memoria in cui scrivere.	

Variabili

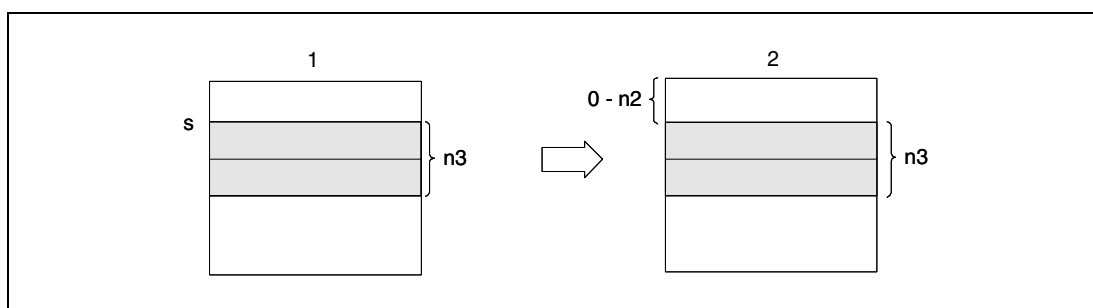
Dati impostati	Significato	Tipo dati
s	Dato da scrivere o primo indirizzo dell'area di memoria della CPU contenente i dati da scrivere.	BIN 16/32-bit
n3	Numero di word dati da scrivere.	BIN 16-bit

Funzioni

Scrittura di dati a 1 o 2 word nel buffer di memoria di un modulo funzione speciale

TO Scrittura dati a 1 word (16-bit)

L'istruzione TO scrive dati da una word dalla memoria della CPU al buffer di memoria di un modulo funzione speciale. Il primo indirizzo della zona di memoria in cui scrivere è indicato da n2, mentre il numero di word è indicato da n3 e l'indirizzo di base del modulo funzione speciale, determinato dalla posizione del modulo nel telaio base, è indicato da n1. Il primo indirizzo dell'area di memoria contenente i dati da scrivere è indicato da s.

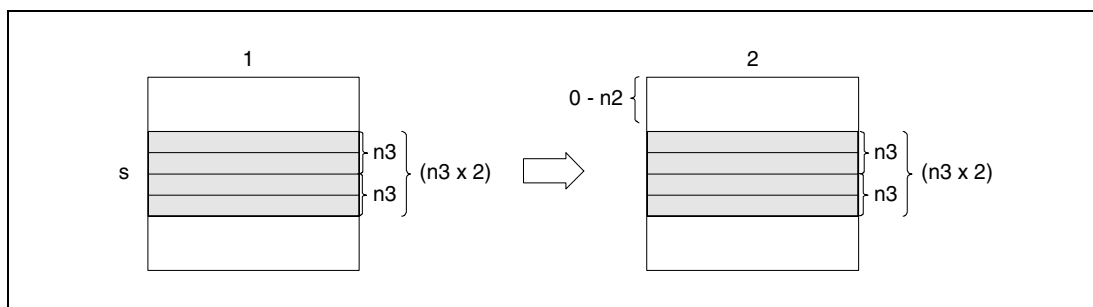


¹ Memoria della CPU

² Buffer di memoria del modulo funzione speciale

DTO Scrittura dati a 2 word (32-bit)

L'istruzione DTO scrive dati da 2 word dalla memoria della CPU al buffer di memoria di un modulo funzione speciale. Il primo indirizzo dell'area di memoria in cui scrivere è indicato da n2, il numero di word dati (multiplo di 2) è indicato da n3, e l'indirizzo di base del modulo funzione speciale è indicato da n1. Il primo indirizzo dell'area di memoria contenente i dati da scrivere è indicato da s.



¹ Memoria della CPU

² Buffer di memoria del modulo funzione speciale

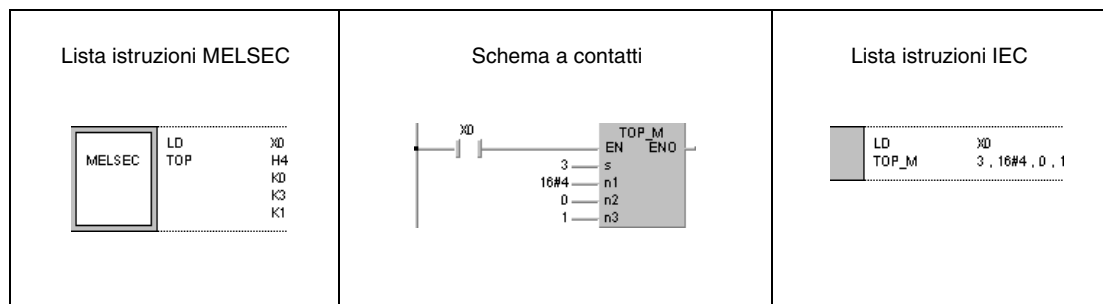
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

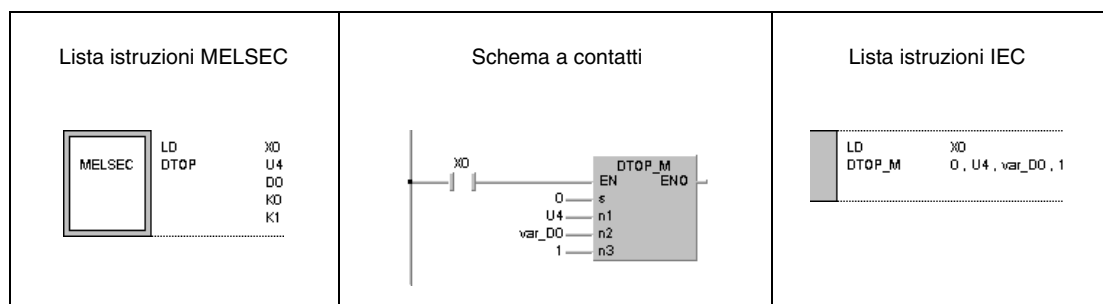
- Nessuno scambio dati con il modulo funzione speciale prima dell'esecuzione dell'istruzione (codice di errore 1412).
- Si è verificato un errore nel modulo prima dell'esecuzione dell'istruzione (codice di errore 1402).
- L'indirizzo di I/O indicato da n2 non si riferisce a un modulo funzione speciale. (serie Q e System Q = codice di errore 2110).
- Il numero di word dati indicato da n3 supera il campo ammesso per l'operando indicato da d (serie Q e System Q = codice di errore: 4101).
- L'indirizzo definito da n2 supera il campo del buffer di memoria (serie Q e System Q = codice di errore 4100).
- L'indirizzo specificato da n2 non è corretto (AJ71QC24) (serie Q e System Q = codice di errore 4100).
- L'accesso al modulo funzione speciale non è consentito.

Programma di esempio 1**TOP**

Con il fronte positivo di X0, il programma seguente imposta i canali CH1 e CH2 di un modulo A68AD per eseguire una conversione A/D. Il modulo funzione speciale si trova agli indirizzi da 040 a 05F. Il valore 3 viene scritto all'indirizzo 0 del buffer di memoria.

**Programma di esempio 2****DTOP**

Con il fronte di salita di X0, il programma che segue azzerava i valori dei dati x negli indirizzi 41 e 42 del buffer di memoria di un modulo AD71. Il modulo funzione speciale si trova agli indirizzi da 040 a 05F.

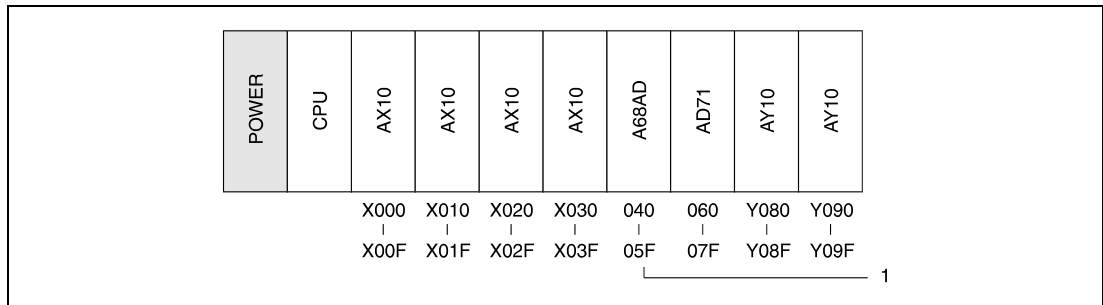


NOTA

L'indirizzo di base n1 deve essere specificato come segue:

n1 = 10 → indirizzo base = 1

n1 = 20 → indirizzo base = 2



¹ Indirizzo base del registro speciale: n1 = K4 o H4

Il programma di esempio 2 non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.9 Istruzioni di visualizzazione

Le serie MELSEC A e Q ed il System Q possiedono diverse istruzioni che inviano caratteri ASCII su un modulo di uscita o sul display a LED del pannello frontale dei moduli CPU che lo comprendono. Sono disponibili in totale 7 istruzioni di visualizzazione.

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Emissione caratteri ASCII	PR	PR_M
	PRC	PRC_M
Visualizzazione di caratteri ASCII e commenti	LED	LED_M
	LEDC	LEDC_M
	LEDA	LEDA_M
	LEDB	LEDB_M
Cancella display	LEDR	LEDR_M

NOTA

Su una CPU A3A, le istruzioni LEDA e LEDB non possono essere utilizzate direttamente come istruzioni di visualizzazione. In questo caso le istruzioni servono come comandi di avvio per le istruzioni applicative dedicate.

Per poter usare le istruzioni LEDA e LEDB con una CPU A3A, si deve modificare la sequenza dei caratteri della stringa, tramite le istruzioni applicative dedicate delle CPU AnA o AnAS. Per i dettagli, fare riferimento al manuale di programmazione separato per la serie AnA e AnAS (Istruzioni dedicate).

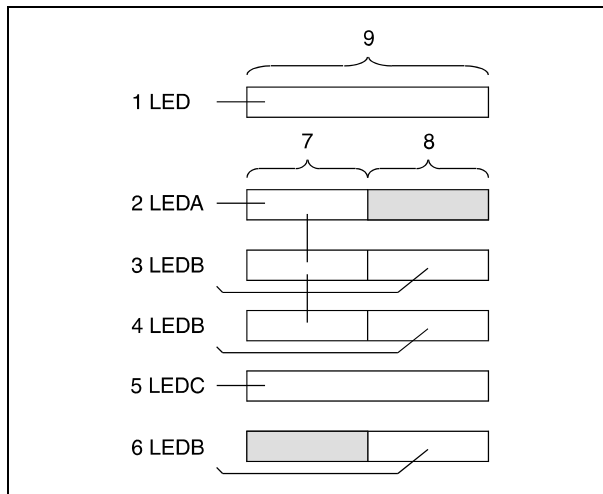
Il display a LED obbedisce alle seguenti priorità:

1. Visualizzazione errori di autodiagnosi
2. Visualizzazione istruzione CHK
3. Visualizzazione numero annunciatore F
4. Visualizzazione carattere ASCII tramite istruzione LED (A, B, C)
5. ERRORE BATTERIA

Su una CPU A3A la priorità può essere modificata a piacimento. Per i dettagli, fare riferimento ai manuali della serie AnA.

Se è visualizzata una indicazione delle prime tre priorità, l'esecuzione di una istruzione di visualizzazione non modifica l'indicazione corrente. Se è visualizzato "BATTERY ERROR" (errore batteria), l'indicazione sul display cambia se viene eseguita una istruzione LED (A, B, C).

La figura seguente mostra il display a LED dopo l'esecuzione di una istruzione LED (A, B, C).



Con una istruzione LED (1) vengono visualizzati fino a 16 caratteri (9). Con una istruzione LEDA (2) vengono visualizzati i primi 8 caratteri (7); gli ultimi 8 caratteri (8) rimangono blank. Se viene eseguita successivamente una istruzione LEDB, i dati vengono visualizzati sugli 8 ultimi caratteri. Se viene nuovamente eseguita una istruzione LEDB (4), i dati visualizzati sugli ultimi 8 caratteri vengono sovrascritti, mentre i dati dei primi 8 caratteri rimangono invariati. L'esecuzione di una istruzione LEDC (5) consente la visualizzazione di un commento predefinito (15 caratteri). La successiva esecuzione di una istruzione LEDB (6) sovrascrive i dati originali, mentre i primi 8 caratteri rimangono blank.

Sul display a LED delle CPU adatte, possono essere visualizzati i seguenti elementi:

Cifre: da 0 a 9

Lettere: da A a Z (maiuscole)

Simboli speciali: < > = * / ' + -

Funzioni

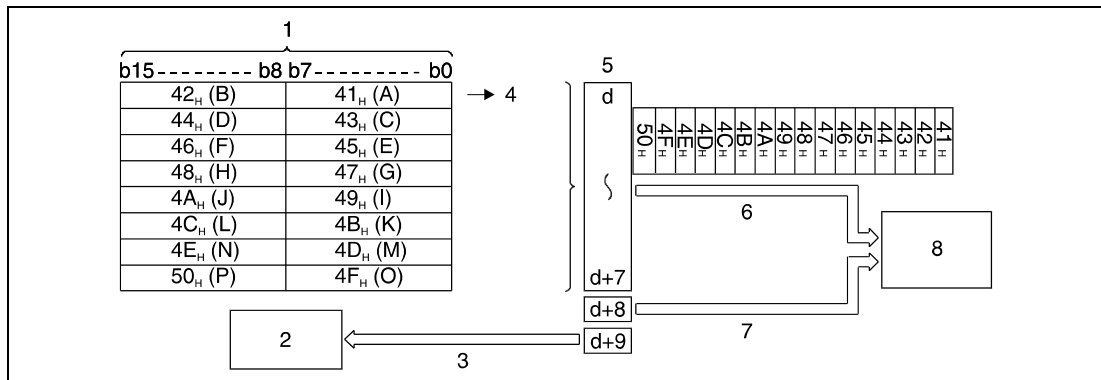
Uscita su dispositivo periferico

PR Invio di una stringa di caratteri ASCII

L'istruzione PR svolge due funzioni. La funzione svolta dipende dallo stato del relé speciale M9049 (serie A) o SM702 (serie Q e System Q):

M9049/SM701 attivo (1) (funzione 1):

Emette una stringa da 16 caratteri ASCII verso un modulo di uscita. La stringa di caratteri, divisa in due metà da 8 caratteri, viene letta dall'area di indirizzi s ed emessa sulle uscite definite da d.



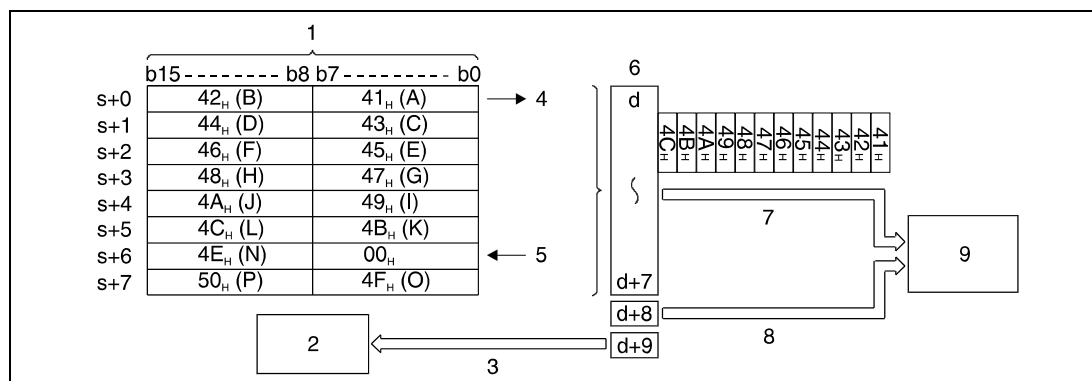
- 1 Operando contenente il codice ASCII
- 2 Programma sequenza
- 3 Segnale che indica l'esecuzione dell'istruzione PR (usato come interblocco)
- 4 Avvio emissione
- 5 Uscite (Y)
- 6 Emissione codice ASCII
- 7 Emissione segnale di strobe
- 8 Stampante o dispositivo di visualizzazione

L'istruzione PR può solo accedere a dati ASCII già memorizzati. Se i dati memorizzati cambiano, viene emesso lo stato attuale. Per la conversione da dati alfanumerici a codice ASCII, usare l'istruzione ASC.

Durante l'emissione dei 16 caratteri del codice ASCII, il flag di esecuzione dell'istruzione PR ($d+9$) è attivo (ON). Quindi l'uscita Y corrispondente all'indirizzo $d+9$ è attiva per tutta la durata dell'istruzione PR.

M9049/SM701 non attivo (0) (funzione 2):

Emette i dati della stringa di caratteri ASCII fino al codice carattere "00H" in formato esadecimale leggendo dall'area specificata da s verso le uscite indicate da d.



- 1 Operando contenente il codice ASCII
- 2 Programma sequenza
- 3 Segnale che indica l'esecuzione dell'istruzione PR (usato come interblocco)
- 4 Avvio emissione
- 5 Fine della stringa di caratteri (fine trasmissione)
- 6 Uscite (Y)
- 7 Emissione codice ASCII
- 8 Emissione segnale di strobe
- 9 Stampante o dispositivo di visualizzazione

Se il contenuto degli operandi che contengono il codice ASCII viene sovrascritto durante l'emissione, viene emesso lo stato attuale.

La fine della stringa di caratteri ASCII è indicata dal codice carattere "00H".

Se il codice esadecimale "00H" non esiste nell'operando specificato, l'esecuzione viene arrestata e viene attivata una segnalazione di errore.

Durante l'emissione dei caratteri del codice ASCII, il flag di esecuzione dell'istruzione PR (d+9) è attivo (ON).

NOTA

Le CPU della serie A possono eseguire solo la funzione 1.

Per l'esecuzione di una istruzione PRC è necessario un modulo di uscita con 10 uscite binarie consecutive. La zona di memoria inizia dall'indirizzo specificato da d.

I segnali emessi dal modulo di uscita sono trasmessi con una cadenza di 30ms per carattere. Quindi, l'invio di n caratteri richiede n x 30ms. La trasmissione di uscita viene controllata con una interruzione a 10 ms, per cui il programma di sequenza viene processato continuamente.

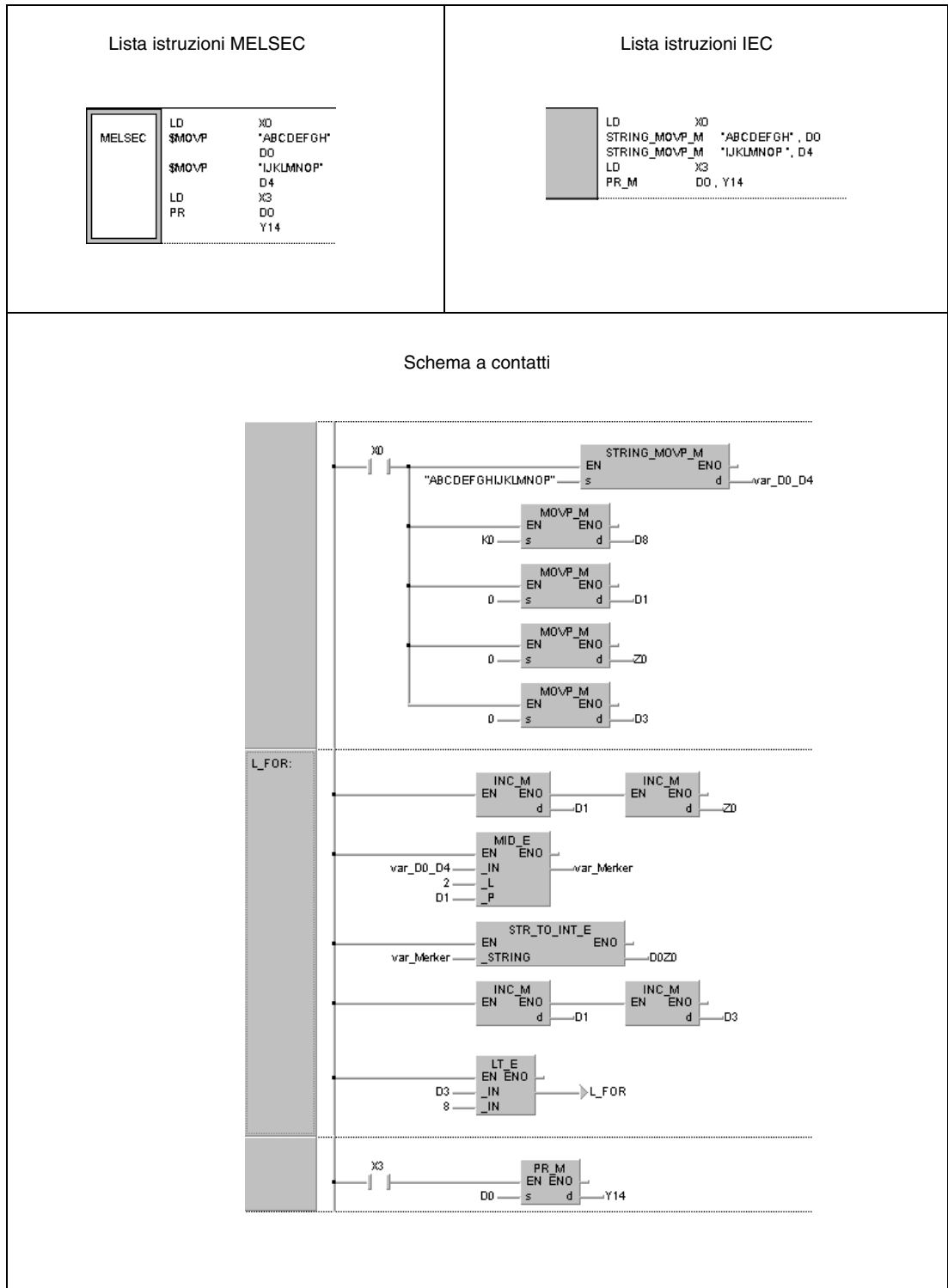
I 10 indirizzi del modulo di uscita vengono processati indipendentemente dal rinfresco degli I/O dopo l'istruzione END del programma di sequenza.

Oltre al codice ASCII, un segnale di strobe (ON ì 10 ms; OFF = 20 ms) viene inviato sull'uscita Y= d+8.

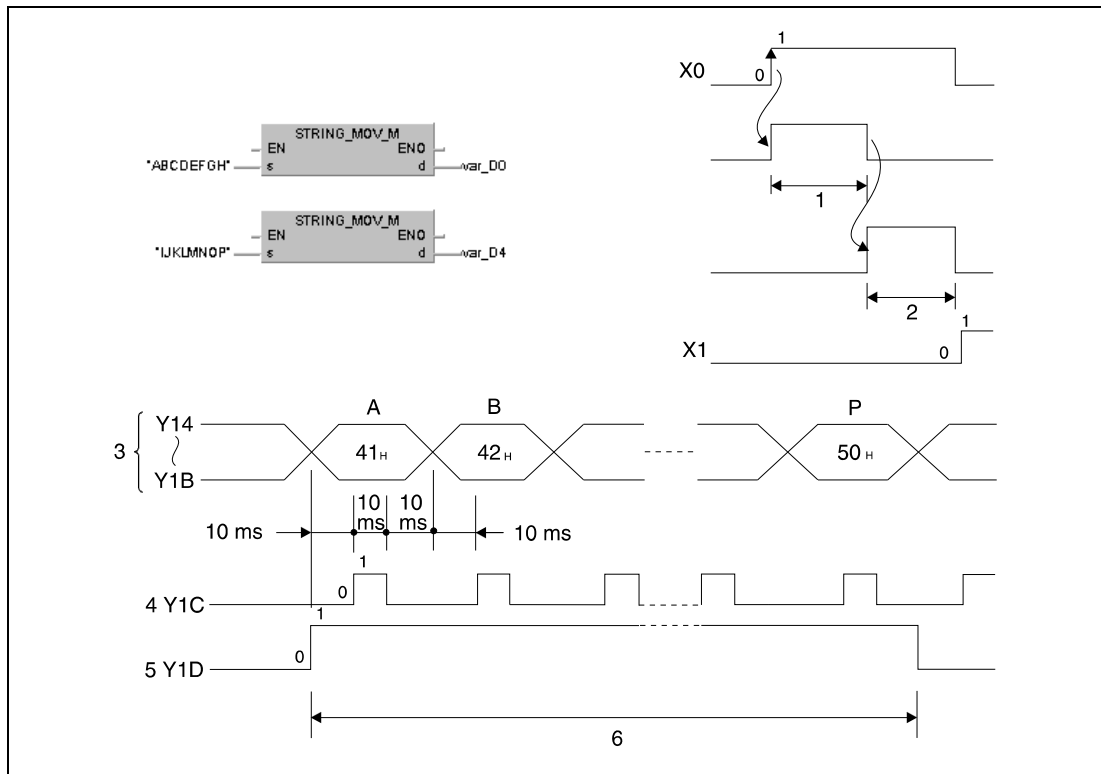
Le istruzioni PR e PRC possono essere eseguite più volte. Si deve quindi prevedere un interblocco tramite il flag di esecuzione dell'istruzione PR (operando di uscita Y= d+9) in modo che le istruzioni PR e PRC non vengano eseguite contemporaneamente.

Esempio di programma PR

Con il fronte positivo di X0, il programma seguente converte la stringa di caratteri "ABCDEFGHJKLMNOP" in codice ASCII e la memorizza nei registri da D0 a D7. Quando X3 viene attivato, il codice ASCII contenuto da D0 a D7 viene inviato sulle uscite da Y14 a Y1D.



Lo schema di temporizzazione seguente mostra l'elaborazione del programma:



¹ Memorizzazione della stringa di caratteri "ABCDEFGH" da D0 a D3

² Memorizzazione della stringa di caratteri "IJKLMNOP" da D4 a D7

³ Codice ASCII

⁴ Segnale di strobe

⁵ Flag di esecuzione istruzione PR

⁶ Elaborazione istruzione PR (periodo = 480 ms)

NOTA

Se non viene utilizzata una CPU della serie A e SM701 non è attivo, si deve scrivere il codice "00H" nel registro D8. Senza questo carattere, il programma dell'esempio precedente genera un errore.

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.9.2 PRC

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC A

	Operandi utilizzabili														Definizione cifra	Numero di passi	Indice	Rapporto errore	Segnal e di errore			
	Operandi a bit						Operandi a word (16-bit)						Costanti							Punta- tore		Livello
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1						Z	V	
s	●	●	●	●	●	●	●	●	●	●	●								●	●		
d		●																				

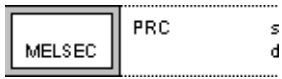
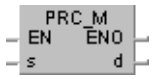
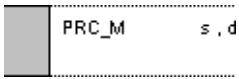
¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

Operandi MELSEC Q

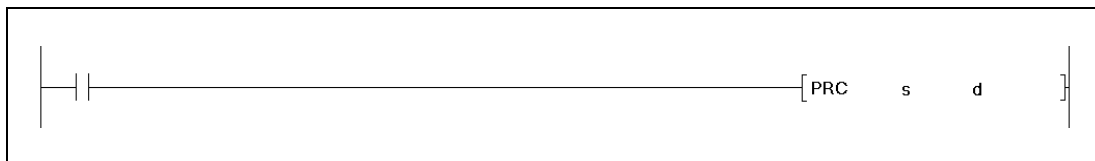
	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti \$			Altro P, I, J, U
	Bit	Word		Bit	Word						
s	●	●	●	●	●	—	—	●	—		
d	● ¹	—	—	—	—	—	—	—	—		

¹ Solo Y

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
--	--	---

GX Developer



Variabili

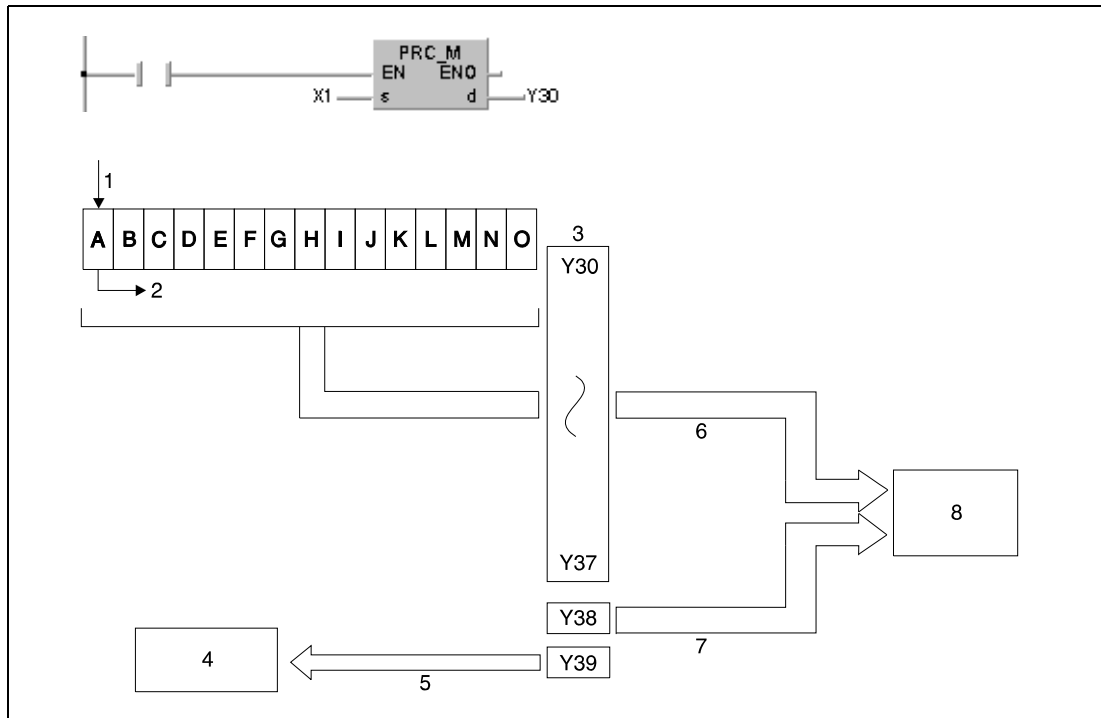
Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene il commento da emettere.	BIN 16-bit
d	Indirizzo base del modulo di uscita per il commento	Bit

Funzioni Uscita su dispositivo periferico**PRC Emissione di un commento**

L'istruzione PRC emette il commento di un operando (in codice ASCII) verso un modulo di uscita.

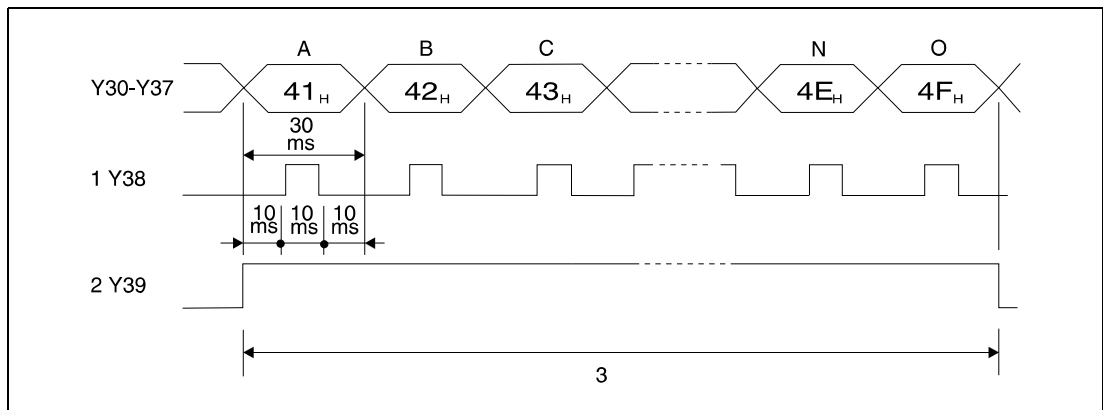
Le CPU MELSEC della serie A leggono la stringa di caratteri divisa in due metà da 8 caratteri dall'area di indirizzi s e la emettono sulle uscite specificate da d.

Con le CPU MELSEC serie Q e System Q si possono emettere 16 o 32 caratteri. La scelta avviene con il relé speciale SM701. Se SM701 è attivo (1), vengono emessi 16 caratteri; se SM701 non è attivo (0), vengono emessi 32 caratteri.



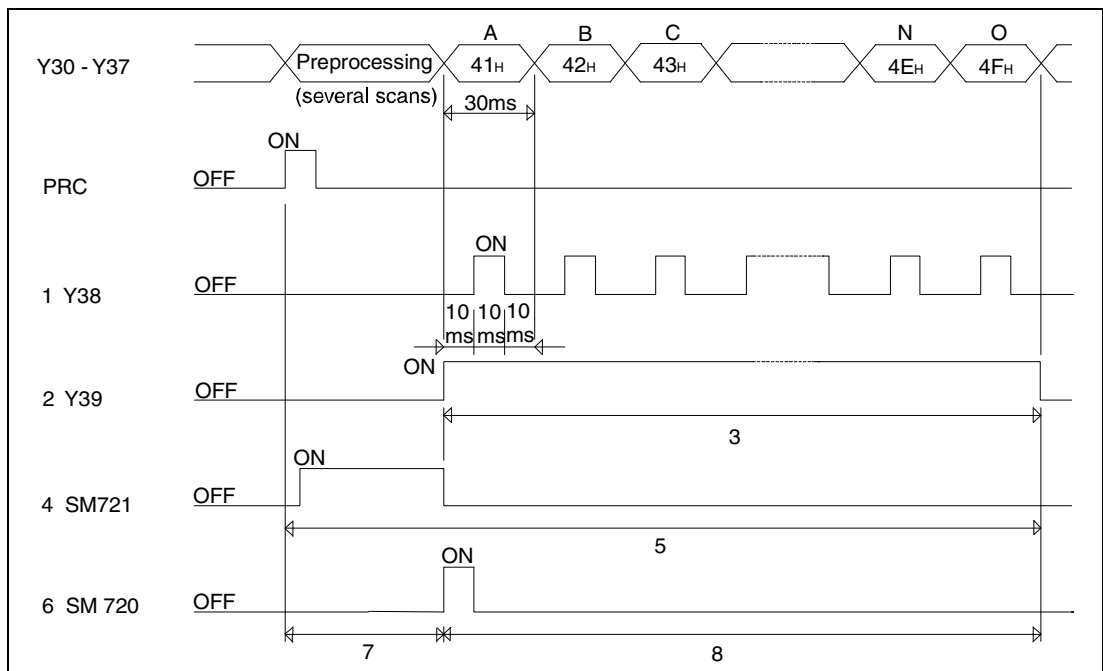
- ¹ Commento (codice ASCII) a partire da X1
- ² Avvio emissione
- ³ Uscite (Y)
- ⁴ Programma sequenza
- ⁵ Flag di esecuzione istruzione PR (usato come interblocco)
- ⁶ Emissione codice ASCII
- ⁷ Emissione segnale di strobe
- ⁸ Stampante o dispositivo di visualizzazione

Il diagramma di temporizzazione seguente illustra l'elaborazione di una istruzione PRC su una CPU QnA:



- 1 Segnale di strobe
- 2 Flag di esecuzione istruzione PRC
- 3 Elaborazione istruzione PR (periodo = 480 ms)

L'elaborazione dell'istruzione PRC in una CPU multi processore del System Q è mostrata dal diagramma di temporizzazione seguente:



- 1 Segnale di strobe
- 2 Flag di esecuzione istruzione PRC
- 3 Tempo di elaborazione (16 x 30 ms = 480 ms) dell'istruzione PRC
- 4 Flag di accesso file in corso
- 5 L'istruzione PRC non può essere eseguita di nuovo
- 6 Flag accesso file terminato
- 7 Nessun'altra istruzione può essere eseguita
- 8 Si possono eseguire istruzioni diverse da PRC, S.FREAD, S.FWRITE, PLOAD, PUNLOAD e PSWAPP

Nel modulo di uscite digitali assegnato ci sono 10 uscite binarie. La zona di memoria inizia dall'uscita Y specificata da d.

I segnali emessi dal modulo di uscita sono trasmessi con una cadenza di 30ms per carattere. Quindi, l'invio di n caratteri richiede n x 30ms. La trasmissione di uscita viene controllata con una interruzione a 10 ms, per cui il programma di sequenza viene processato continuamente.

Oltre al codice ASCII, un segnale di strobe (ON = 10 ms; OFF = 20 ms) viene inviato sull'uscita Y= d+8.

Durante l'emissione dei 16 caratteri del codice ASCII, il flag di esecuzione dell'istruzione PRC (d+9) è attivo (ON). Quindi l'uscita Y corrispondente all'indirizzo d+9 è attiva per tutta la durata dell'istruzione PRC. Le istruzioni PR e PRC possono essere eseguite più volte. Si deve quindi prevedere un interblocco tramite il flag di esecuzione dell'istruzione PRC (operando di uscita Y= d+9) in modo che le istruzioni PR e PRC non vengano eseguite contemporaneamente.

Se l'area indirizzata da s non contiene dati, l'istruzione non viene eseguita.

L'istruzione PRC può accedere solo a commenti già memorizzati nel PLC. Per la conversione da dati alfanumerici a codice ASCII, usare l'istruzione ASC.

Al termine dell'esecuzione dell'istruzione PRC, SM720 si attiva (ON) per una scansione. SM721 è attivo (ON) durante l'esecuzione dell'istruzione PRC. L'istruzione PRC non può essere eseguita se SM721 è già ON. Se si cerca di farlo, l'elaborazione non viene eseguita.

NOTA

L'istruzione PRC può accedere solo a commenti memorizzati in una memory card. L'istruzione PRC non può accedere a commenti memorizzati nella memoria interna.

Il file di commento utilizzato dall'istruzione PRC è impostato nel parametro "PC File Setting" in modo parametrizzazione.

Non è possibile l'emissione di un file di commento con l'istruzione PRC se il file di commento non è stato impostato.

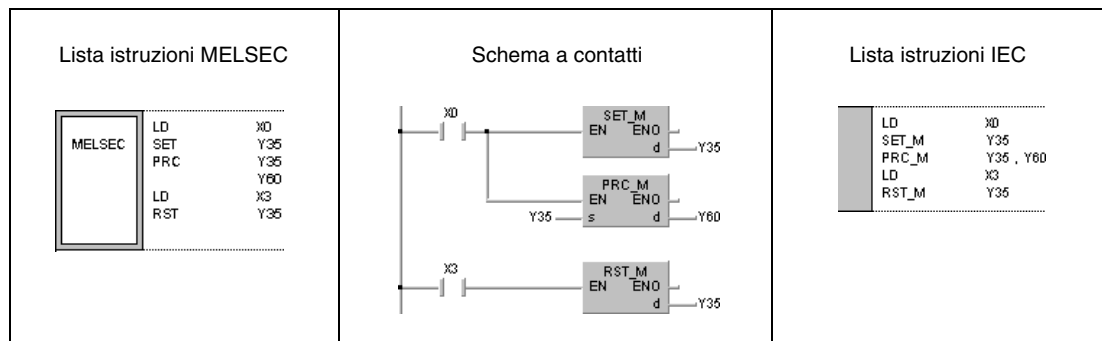
Non eseguire l'istruzione PRC da un programma di interruzione. Vengono provocati malfunzionamenti.

Gli operandi di commento per l'istruzione PRC sono memorizzati su una memory card IC. La memoria interna della CPU non può contenere commenti (solo serie Q e System Q).

Esempio di programma

PRC

Se X0 è attivo (ON), il programma seguente attiva l'uscita Y35 ed emette contemporaneamente il commento di Y35 in codice ASCII sulle uscite da Y60 a Y69. Quando X3 viene attivato, Y35 viene resettato (OFF).



7.9.3 LED

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
	● ¹	● ²	●	● ³	

¹ Solo CPU A3N

² Solo CPU A3A

³ Tranne CPU Q2A (S1)

Operandi MELSEC A

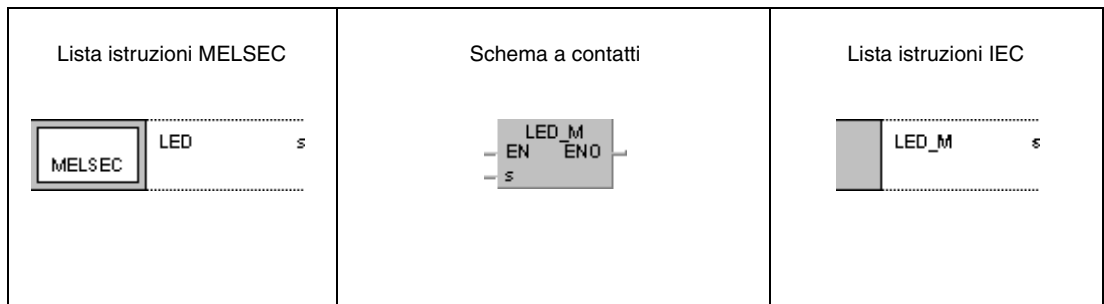
	Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Rapporto errore	Segnal e di errore								
	Operandi a bit						Operandi a word (16-bit)						Costanti		Punta-tore							Livello							
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V				K	H (16#)	P	I	N	M9012	M9010 M9011			
s							●	●	●	●	●															3	●		●

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

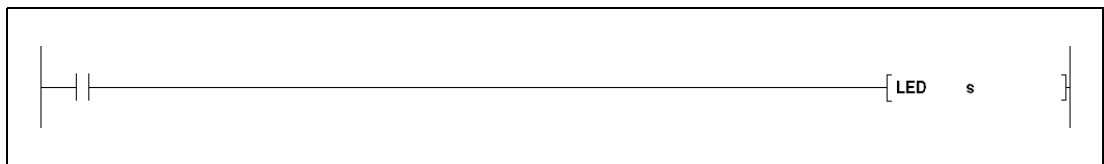
Operandi MELSEC Q

	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti \$			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	●	—	—	2

GX IEC Developer



GX Developer

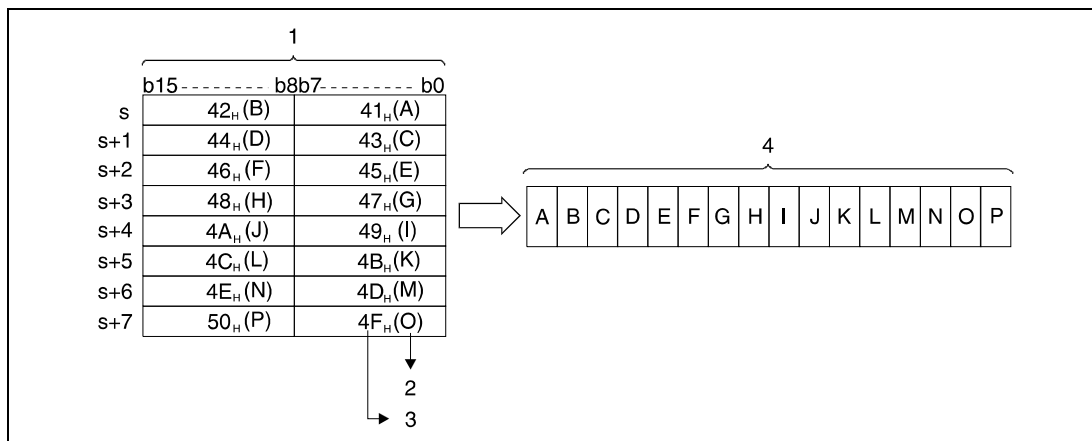


Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene i dati ASCII da visualizzare.	Stringa caratteri

Funzioni **Invio su un display a LED****LED** **Visualizza dati ASCII sul display a LED della CPU**

L'istruzione LED legge dati ASCII (16 caratteri) da una area specificata e li visualizza su un display CPU adeguato. L'indirizzo del primo operando che contiene il codice ASCII in 8 indirizzi è specificato da s (vedi figura seguente).



¹ Dati da visualizzare

² Carattere ASCII

³ Codice ASCII (esadecimale)

⁴ Display a LED sul pannello frontale della CPU

Se nell'area specificata non sono contenuti dati ASCII, la visualizzazione di timer, contatori e registri dati e di collegamento viene cancellata. Per il file registri R, la visualizzazione viene cancellata se i file registri corrispondenti sono già cancellati.

Sul display a LED delle CPU adatte, possono essere visualizzati i seguenti elementi:

Cifre: da 0 a 9

Lettere: da A a Z (maiuscole)

Simboli speciali: < > = * / ^ + -

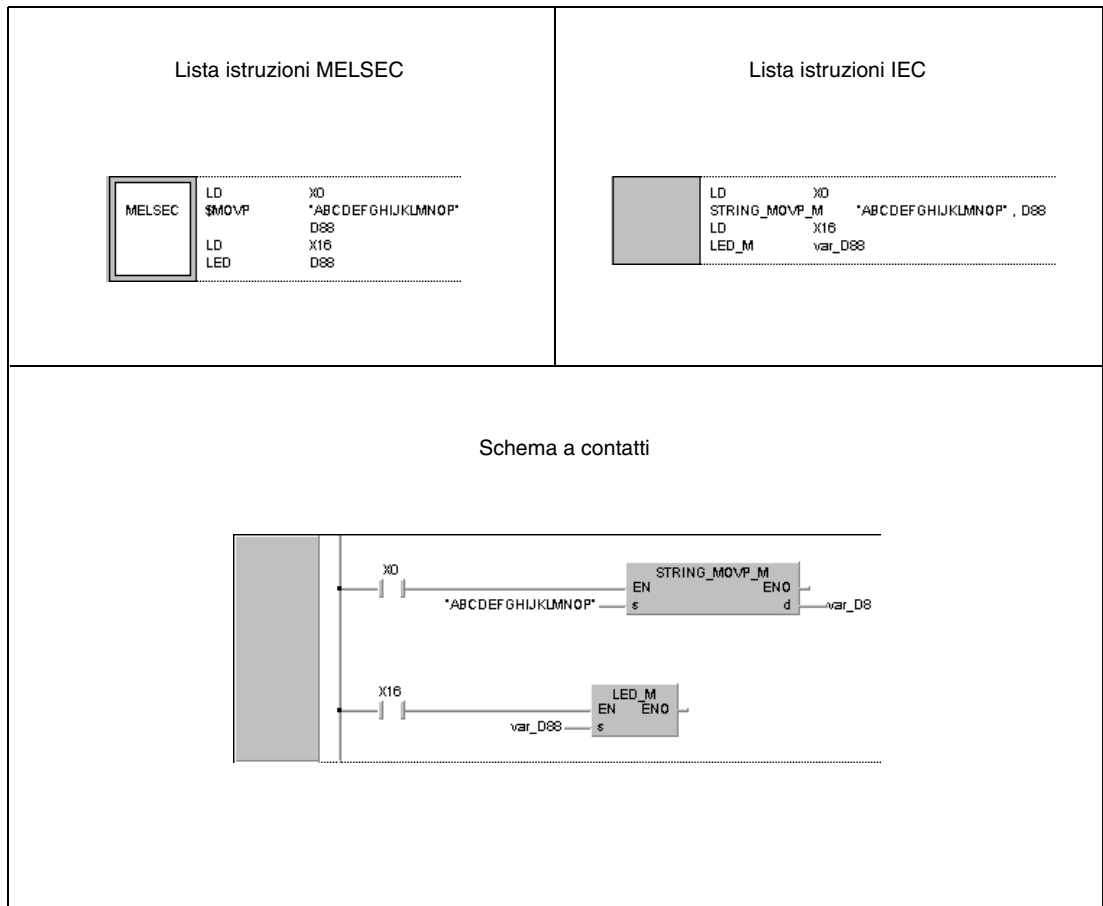
L'istruzione LED può solo accedere a dati ASCII già memorizzati. Per la conversione da dati alfanumerici a codice ASCII, usare l'istruzione \$MOV o ASC.

NOTA

L'istruzione LED può essere usata solo con le CPU A3N, A3A, Q3A, Q4A o Q4AR. Se l'istruzione viene eseguita su una CPU senza display a LED, non viene eseguita nessuna elaborazione.

Esempio di programma LED

Il programma seguente converte una stringa di caratteri in codice ASCII, la memorizza nei registri specificati ed invia il contenuto dei registri sul display a LED. In un primo passo, dopo l'attivazione di X0, il programma seguente converte la stringa di caratteri "ABCDEFGHJKLMNOP" in codice ASCII e la memorizza nei registri dati da D88 a D95. Dopo l'attivazione di X16, i dati ASCII memorizzati da D88 a D95 vengono visualizzati sul display della CPU.



7.9.4 LEDC

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
	● ¹	● ²	●	● ³	

¹ Solo CPU A3N

² Solo CPU A3A

³ Tranne CPU Q2A (S1)

Operandi MELSEC A

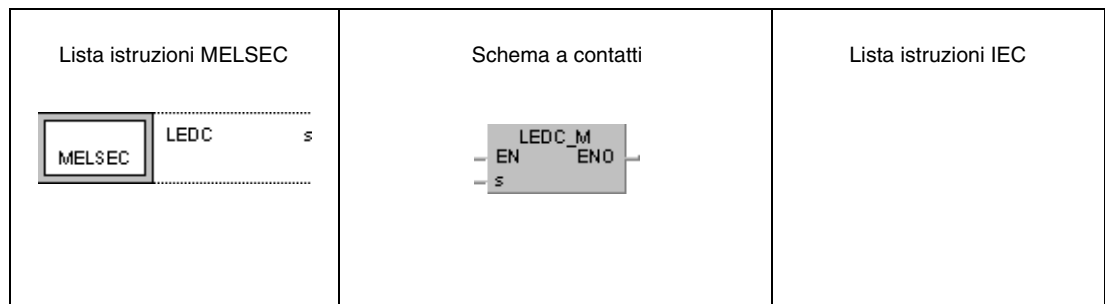
Operandi utilizzabili																	Definizione cifra	Numero di passi	Indice	Rapporto errore	Segnal e di errore			
Operandi a bit							Operandi a word (16-bit)							Costanti		Punta-tore						Livello		
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K				H (16#)	P	I	N	M9012
●	●	●	●	●	●	●	●	●	●	●	●							●	●		3	● ¹	●	●

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

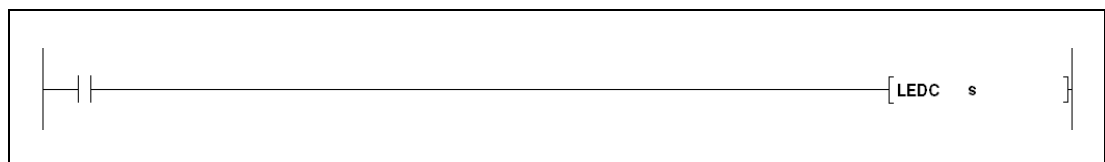
Operandi MELSEC Q

s	Operandi utilizzabili								Segnale di errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti \$			Altro
	Bit	Word		Bit	Word						
●	●	●	●	●	●	—	—	●	—	2	

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene il commento da visualizzare.	Nome operando

Funzioni **Invio su un display a LED****LEDC** **Visualizza commento sul display a LED della CPU**

L'istruzione LEDC legge commenti (16 caratteri) da una area specificata e li visualizza su un display CPU adeguato. La visualizzazione viene comunque limitata ai primi 16 caratteri da visualizzare. L'indirizzo del primo operando che contiene il commento è specificato da s.

Se nell'operando specificato non è memorizzato nessun commento, il display sul frontale della CPU rimane spento. Se i dati risiedono oltre il campo del commento, l'istruzione LEDC non viene elaborata e la visualizzazione sul display resta invariata.

Se un commento contiene dei caratteri non visualizzabili, la visualizzazione può essere non corretta. Sul display a LED delle CPU adatte, possono essere visualizzati i seguenti elementi:

Cifre: da 0 a 9

Lettere: da A a Z (maiuscole)

Simboli speciali: < > = * / ^ + -

Una Q2ACPU(S1) non supporta l'istruzione LED. L'elaborazione dell'istruzione non produce alcun effetto.

NOTA

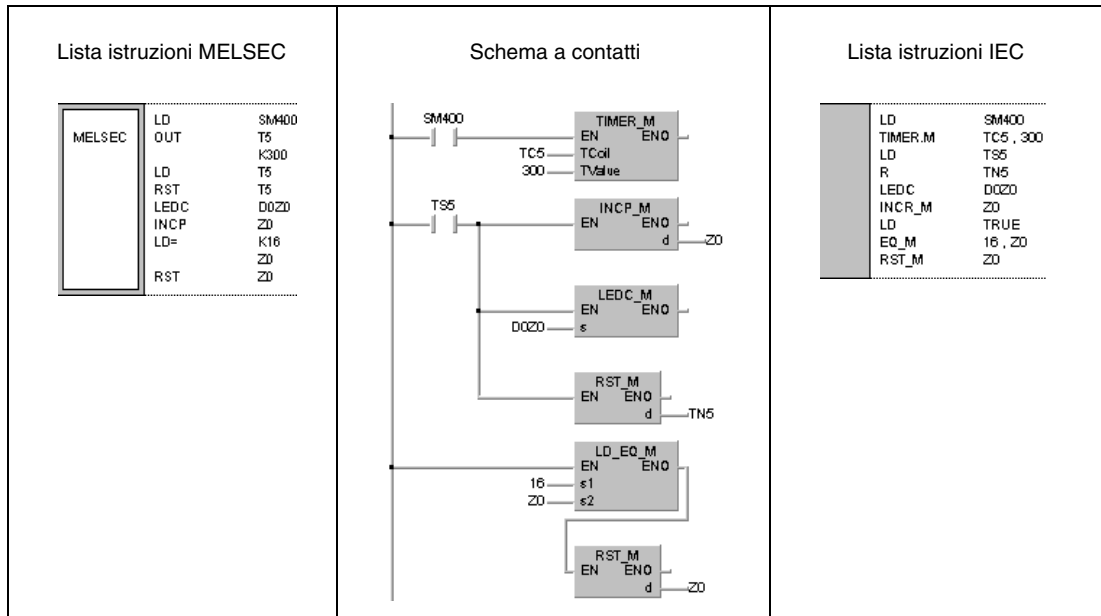
L'istruzione LEDC delle istruzioni dedicate delle CPU AnA imposta gli operandi.

Fare riferimento al manuale di programmazione separato per le CPU AnA, per i dettagli sulla programmazione dell'istruzione LEDC con una CPU A3A (Istruzioni dedicate) (solo serie A).

Esempio di programma

LEDC

Il programma seguente visualizza i commenti da D0 a D15 in intervalli di 30 secondi. Il timer T5 attiva la condizione di ingresso dell'istruzione LEDC ogni 30 secondi. Quando il timer commuta ON, il commento nei registri dati D(0+Z) viene visualizzato ed il valore di Z viene incrementato di 1. Quando Z diventa uguale a 16, il valore di Z viene azzerato.



7.9.5 LEDA, LEDB

CPU

AnS	AnN	An(S)	AnU	QnA(S), Q4AR	System Q
	● ¹				

¹ Solo CPU A3N

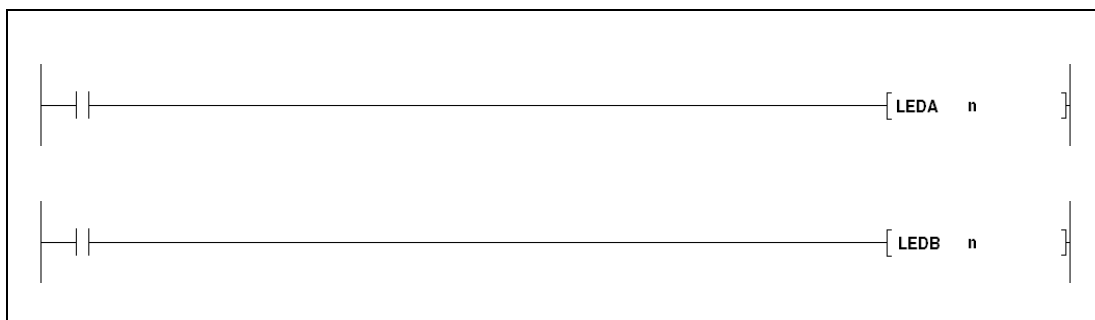
Operandi MELSEC A

Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Segnal e di riporto	Segnal e di errore						
Operandi a bit						Operandi a word (16-bit)						Costanti		Punta- tore							Livello					
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H (16#)	P	I	N				M9012	M9010 M9011	
n																							13			

GX IEC Developer

<p>Lista istruzioni MELSEC</p> <table border="1" style="margin-left: 20px;"> <tr> <td style="border: 1px solid black; padding: 2px;">MELSEC</td> <td>LEDA</td> <td>n</td> </tr> <tr> <td></td> <td>LEDB</td> <td>n</td> </tr> </table>	MELSEC	LEDA	n		LEDB	n	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p> <table border="1" style="margin-left: 20px;"> <tr> <td>LEDA_M</td> <td>n</td> </tr> <tr> <td>LEDB_M</td> <td>n</td> </tr> </table>	LEDA_M	n	LEDB_M	n
MELSEC	LEDA	n										
	LEDB	n										
LEDA_M	n											
LEDB_M	n											

GX Developer



Variabili

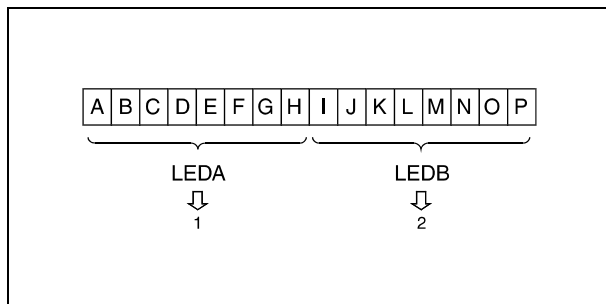
Dati impostati	Significato	Tipo dati
n	Dati ASCII	Stringa caratteri

Funzioni Invio su un display a LED

LEDA, LEDB Visualizza stringa di caratteri ASCII sul display a LED della CPU

Queste istruzioni visualizzano una stringa di caratteri ASCII sul display a LED di una CPU appropriata. La stringa ASCII è composta da 8 caratteri per ogni istruzione e viene specificata con l'istruzione LEDA o LEDB.

In totale, è possibile visualizzare fino a 16 caratteri con entrambe le istruzioni. L'istruzione LEDA definisce i primi 8 caratteri (metà sinistra), mentre l'istruzione LEDB definisce gli ultimi 8 caratteri (metà destra) del display a LED.



- ¹ Definizione dei primi 8 caratteri
- ² Definizione degli ultimi 8 caratteri

Sul display a LED delle CPU adatte, possono essere visualizzati i seguenti elementi:

- Cifre: da 0 a 9
- Lettere: da A a Z (maiuscole)
- Simboli speciali: < > = * / ' + -

NOTA *L'uso delle istruzioni LEDA / LEDB su una CPU AnA o AnU indica l'inizio delle istruzioni dedicate. Fare riferimento al manuale di programmazione separato delle CPU AnA o AnU (Istruzioni dedicate) per i dettagli sulla programmazione delle istruzioni LEDA / LEDB.*

Esempio di programma LEDA, LEDB
 Se XC è attivo, il programma seguente invia la stringa di caratteri "ABCDEFGH IJKLMNOP" al display a LED della CPU.

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC									
<table border="1" style="border-collapse: collapse; margin-left: auto; margin-right: auto;"> <tr> <td style="padding: 2px;">MELSEC</td> <td style="padding: 2px;">LD</td> <td style="padding: 2px;">XC</td> </tr> <tr> <td style="padding: 2px;"></td> <td style="padding: 2px;">LEDA</td> <td style="padding: 2px;">"ABCDEFGH"</td> </tr> <tr> <td style="padding: 2px;"></td> <td style="padding: 2px;">LEDB</td> <td style="padding: 2px;">"JKLMNOP"</td> </tr> </table>	MELSEC	LD	XC		LEDA	"ABCDEFGH"		LEDB	"JKLMNOP"		
MELSEC	LD	XC									
	LEDA	"ABCDEFGH"									
	LEDB	"JKLMNOP"									

NOTA *L'ultima metà della stringa di caratteri visualizzata con l'istruzione LED viene cancellata, se i primi 8 caratteri vengono sovrascritti dall'istruzione LEDA. Viceversa, la prima metà di una stringa di caratteri viene cancellata se gli ultimi 8 caratteri vengono sovrascritti con una istruzione LEDB.*

7.9.6 LEDR

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

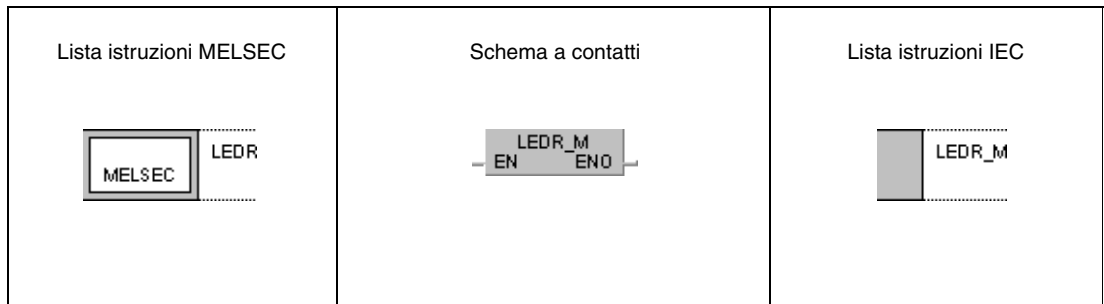
Operandi MELSEC A

Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Riporto	Errore				
Operandi a bit						Operandi a word (16-bit)						Costanti	Punta-tore	Livello										
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V				K	H (16#)	P	I	N	M9012
																						1		

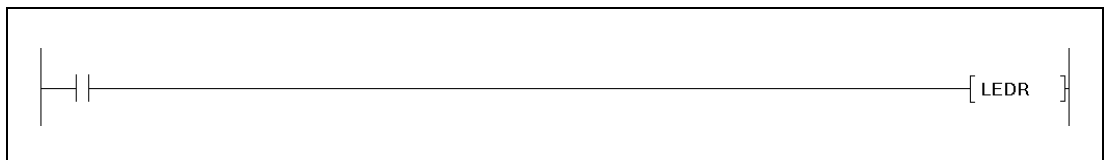
Operandi MELSEC Q

Operandi utilizzabili										Errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti	Altro			
Bit	Word		Bit	Word							
—	—	—	—	—	—	—	—	—	—	—	1

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
—	—	—

Funzioni Cancella spie e segnalazioni di errore**LEDR Istruzione di cancellazione**

L'istruzione LEDR cancella le spie di errore impostate automaticamente in caso di errore di esecuzione. L'istruzione LEDR produce lo stesso effetto della premuta del pulsante INDICATOR RESET sui moduli CPU con display a LED (solo serie A).

Funzionamento dell'istruzione LEDR con spie di errore attivate durante l'autodiagnosi (Solo serie Q e System Q):

Se si verifica un errore durante l'autodiagnosi che non compromette il corretto funzionamento della CPU, l'esecuzione di una istruzione LEDR spegne il LED "ERROR" e la visualizzazione dell'errore sulla CPU.

Inoltre SM1 e SD0 devono essere cancellati dal programma utente, dato che non vengono azzerati automaticamente dall'istruzione LEDR. Non vengono eseguiti neppure i passi successivi per la cancellazione delle spie di errore.

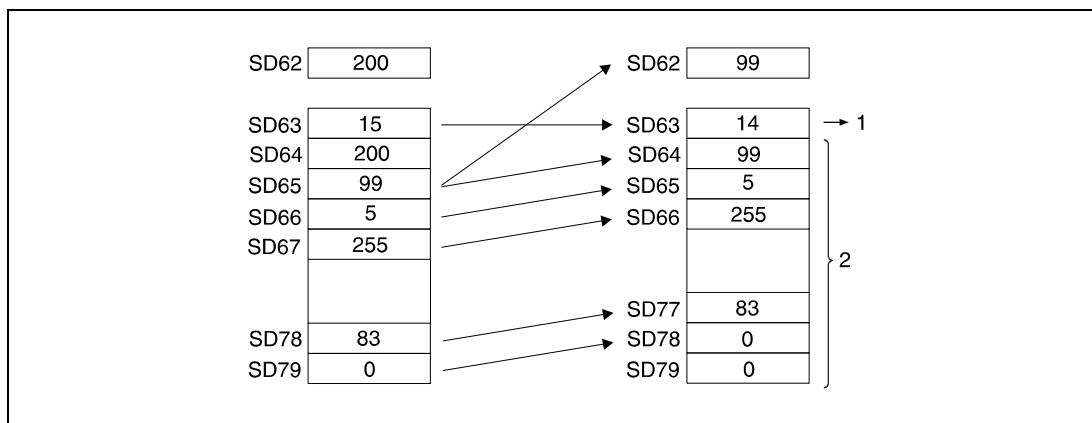
Funzionamento dell'istruzione LEDR in caso di errore di batteria (solo serie Q):

Se l'istruzione LEDR viene eseguita dopo la sostituzione della batteria, il LED "BAT. ARM" sul pannello frontale della CPU e la visualizzazione dell'errore sulla CPU vengono cancellati. Contemporaneamente, anche SM52 viene automaticamente azzerato.

Funzionamento dell'istruzione LEDR con una spia di errore F attiva su una CPU senza display a LED:

A seguito dell'esecuzione dell'istruzione LEDR, vengono eseguite le seguenti operazioni:

- Il LED ERROR sul pannello frontale della CPU lampeggia e poi si spegne.
- La spia di errore F memorizzata in D9009 (serie A) o SD62 (serie Q e System Q) viene cancellata.
- I registri D9009 e D9125 (serie A) o SD62 e SD64 (serie Q/System Q) vengono azzerati e le spie di errore memorizzate da D9126 a D9131 (serie A) o da SD65 a SD79 (serie Q/System Q) vengono fatte scorrere per ulteriore elaborazione.
- Il numero della nuova spia di errore F fatta scorrere in D9125 (serie A) o SD62 (serie Q/System Q) viene scritto in D9009 (serie A) o SD62 (serie Q/System Q).
- L'accumulatore della spia di errore in D9124 (serie A) o SD63 (serie Q/System Q) viene decrementato di 1. Se D9124 (serie A) o SD63 (serie Q/System Q) sono già a 0, questo valore rimane invariato.



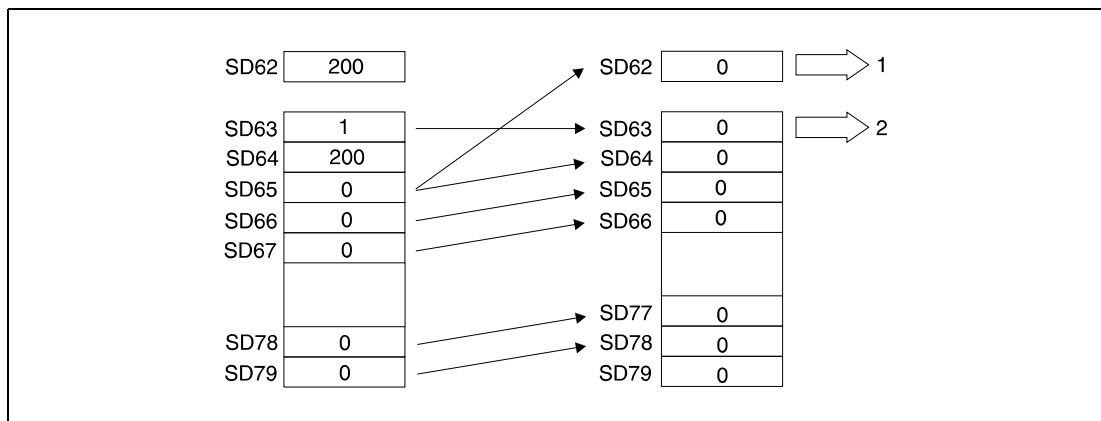
¹ Numero di spie di errore memorizzate

² Area memorizzazione numeri F

Funzionamento dell'istruzione LEDR con una spia di errore F attiva su una CPU con display a LED:

A seguito dell'esecuzione dell'istruzione LEDR, vengono eseguite le seguenti operazioni:

- La spia di errore visualizzata sul display a LED viene cancellata.
- La spia di errore F memorizzata in D9009 (serie A) o SD62 (serie Q e System Q) viene cancellata.
- I registri D9009 e D9125 (serie A) o SD62 e SD64 (serie Q/System Q) vengono azzerati e le spie di errore memorizzate da D9126 a D9131 (serie A) o da SD65 a SD79 (serie Q/System Q) vengono fatte scorrere per ulteriore elaborazione.
- Il numero della nuova spia di errore F fatta scorrere in D9125 (serie A) o SD62 (serie Q/System Q) viene scritto in D9009 (serie A) o SD62 (serie Q/System Q).
- L'accumulatore della spia di errore in D9124 (serie A) o SD63 (serie Q/System Q) viene decrementato di 1. Se D9124 (serie A) o SD63 (serie Q/System Q) sono già a 0, questo valore rimane invariato.
- Il numero attuale della spia di errore memorizzata in D9009 (serie A) o SD62 (serie Q e System Q) viene visualizzato. Se D9124 (serie A) o SD63 (serie Q/System Q) sono già a 0, la visualizzazione non avviene.



¹ Dato che SD63 contiene 0, nessuna spia di errore viene visualizzata sul display a LED.

² Numero di spie di errore memorizzate

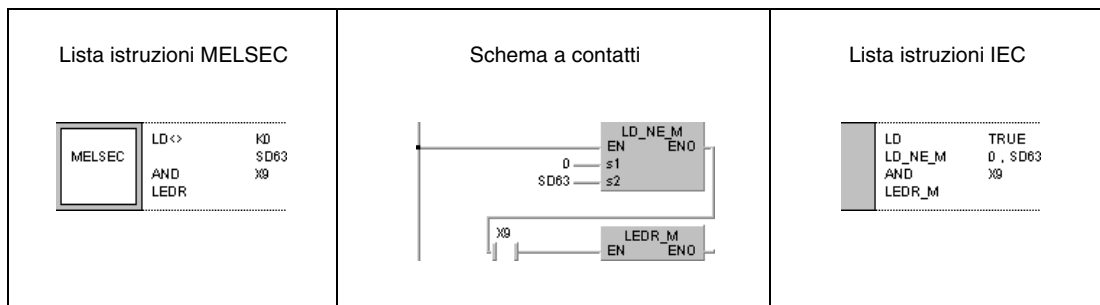
NOTA

Solo serie A:

L'uso dell'istruzione LEDR su una CPU AnA o AnU indica la fine delle istruzioni dedicate. Fare riferimento al manuale di programmazione separato delle CPU AnA o AnU (Istruzioni dedicate) per i dettagli sulla programmazione delle istruzioni LEDR.

Esempio di programma**LEDR**

Se X9 è attivo e il valore nel registro SD63 è diverso da 0, il programma seguente esegue una istruzione LEDR.

**NOTA**

La tabella seguente mostra i valori di default per i numeri di errore impostati nei registri speciali da SD207 a SD209 ed il loro ordine di priorità.

Ordine di priorità	Numero di errore (Esadecimale)	Descrizione	Nota
1	1	AC DOWN	Interruzione alimentazione
2	2	UNIT VERIFY ERR. FUSE BREAK OFF P. UNIT ERROR	Errore verifica modulo di I/O Fusibile interrotto Errore verifica modulo funzione speciale
3	3	OPERATIN ERROR LINK PARA ERROR SFCP OPE. ERROR SFCP EXE. ERROR	Errore operazione Errore parametro di collegamento Errore di esecuzione istruzione SFC Errore esecuzione programma SFC
4	4	ICM.OPE ERROR FILE OPE ERROR EXTEND INST. ERROR	Errore funzionamento memory card Errore accesso file Errore istruzione estesa
5	5	PRG.TIME OVER	Superamento tempo di ciclo costante o watchdog Superamento tempo sorveglianza esecuzione a bassa velocità
6	6	Istruzione CHK	L'errore viene rilevato dall'istruzione CHK
7	7	Spia di errore	
8	8	Istruzione LED	
9	9	BATTERY ERR.	
10	A	Dati orologio	

7.10 Diagnostica guasti e debug

Le istruzioni per ricerca guasti e debug, supportano il controllo guasti, attivazione e disattivazione della memoria di stato, traccia di campionamento e traccia di programma. La tabella che segue offre una panoramica di queste istruzioni:

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Controllo anomalie	CHKST	CHKST_M
	CHK	CHK_M
	CHKCIR	CHKCIR_M
	CHKEND	CHKEND_MD
Set / reset memoria di stato	SLT	SLT_M
	SLTR	SLTR_M
Set / reset traccia campionamento	STRA	STRA_M
	STRAR	STRAR_M
Esegue / attiva / disattiva traccia programma	PTRA	PTRA_M
	PTRAR	PTRAR_M
	PTRAEXE	PTRAEXE_M
	PTRAEXEP	PTRAEXEP_M

NOTA

Controllare se queste funzioni sono disponibili e supportate dalla versione di GX IEC Developer utilizzata.

7.10.1 CHKST, CHK (Solo serie Q e System Q):

CPU

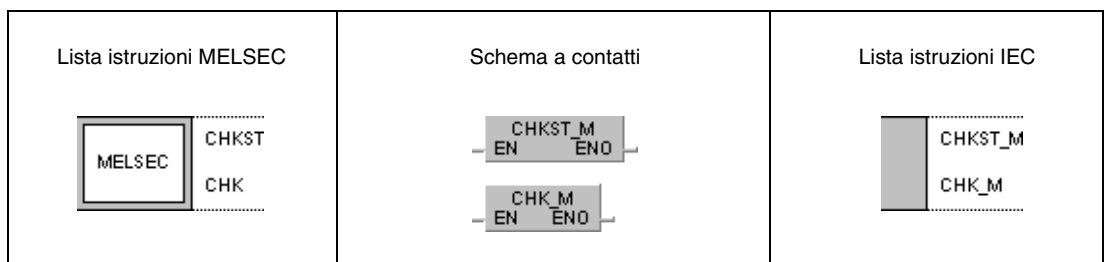
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

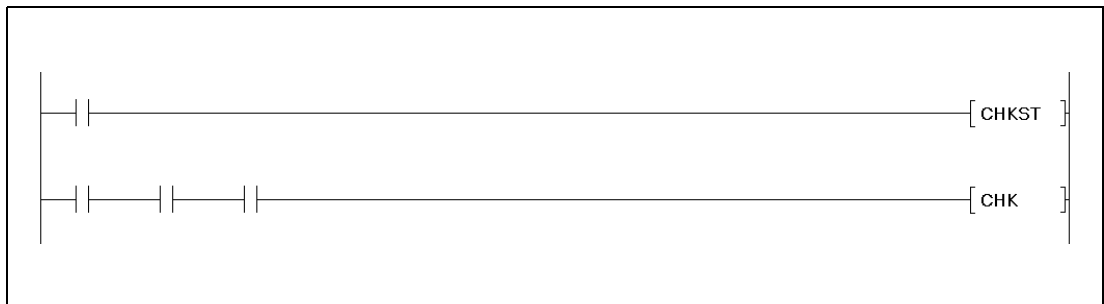
Operandi MELSEC Q

Operandi utilizzabili										Errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word				DY			
—	—	—	—	—	—	—	—	—	—	SM0	1

GX IEC Developer



GX Developer



Variabili

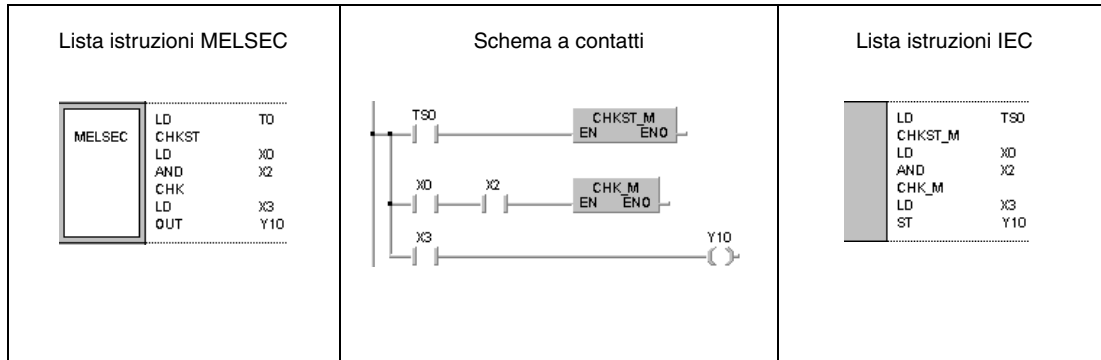
Dati impostati	Significato	Tipo dati
—	—	—

Funzioni

Controllo anomalie per operazioni bidirezionali (solo serie Q e System Q)

CHKST Istruzione di avvio per l'istruzione CHK

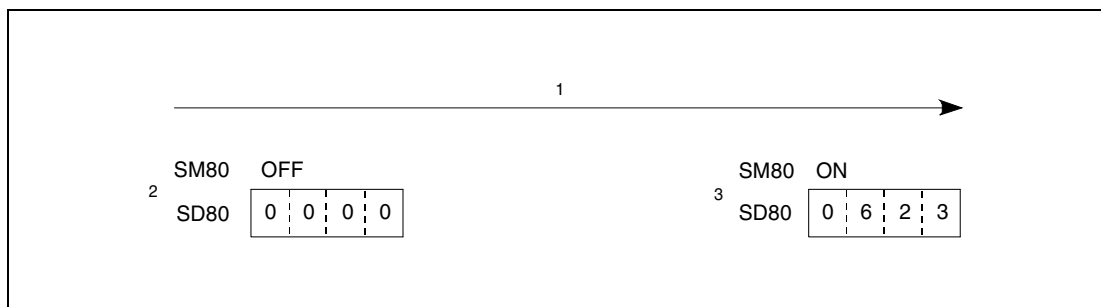
L'istruzione CHKST avvia l'esecuzione dell'istruzione CHK. Se la condizione di esecuzione per l'istruzione CHKST non è attiva (0), viene eseguito il passo di programma che segue l'istruzione CHK. Quando la condizione di esecuzione dell'istruzione CHKST è attiva (1), l'istruzione CHK viene eseguita. Queste istruzioni sono programmate nello schema a contatti sottostante.



CHK Istruzione controllo anomalie

Con alcuni tipi di CPU (e in funzione del modo di controllo) l'istruzione CHK consente il controllo di anomalie sui circuiti a contatti con finecorsa che sorvegliano i movimenti bidirezionali. Se si verifica un errore in questi circuiti, il relé speciale SM80 si attiva e il codice di errore corrispondente viene memorizzato in un registro speciale.

Le CPU della serie Q e del System Q memorizzano il codice di errore come numero BCD a 4 cifre nel registro speciale SD80. Le tre cifre più pesanti indicano il numero del contatto (in questo caso contatto 62), mentre la cifra più leggera contiene il numero del circuito di controllo anomalia (numero bobina 1 - 6; in questo caso bobina 3).



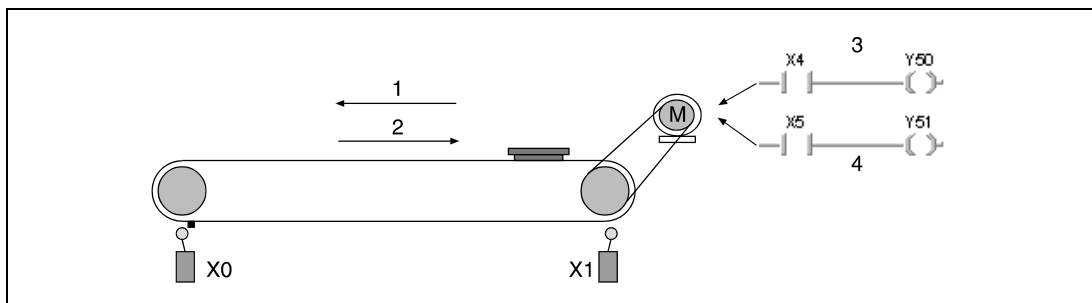
¹ Contatto 62; bobina 3 (durante controllo anomalia)

² Prima del controllo anomalia

³ Dopo il controllo anomalia

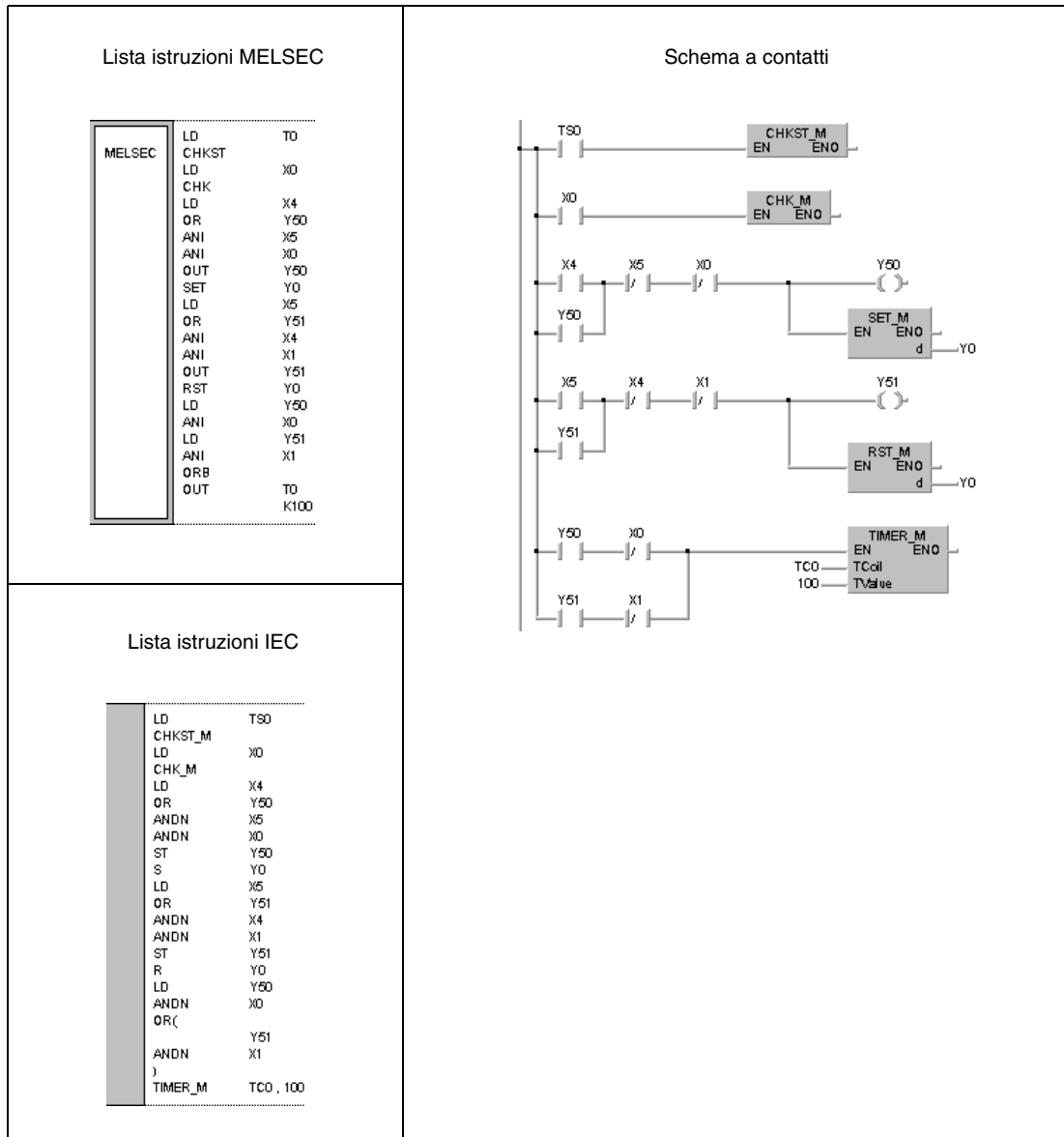
I contatti di ingresso programmati prima dell'istruzione CHK non sono utilizzati come condizione di esecuzione dell'istruzione CHK, ma come specifica delle condizioni di controllo.

Di seguito, la programmazione del controllo anomalie tramite l'istruzione CHK viene illustrata con un esempio concreto. La figura seguente mostra un nastro trasportatore che si muove da sinistra, fino al finecorsa destro. I finecorsa relativi vengono rilevati tramite interruttori di finecorsa. Il contatto di avviamento per il movimento di avanzamento è X4 e quello per il movimento di arretramento è X5.



- ¹ Movimento di avanzamento
- ² Movimento di arretramento
- ³ Comando avanzamento
- ⁴ Comando arretramento

La figura seguente mostra un programma di esempio per il funzionamento e il controllo errori del convogliatore mostrato in precedenza, usando una CPU serie Q. In assenza di errori il programma salta al passo di programma dopo l'istruzione CHK. Con il fronte di salite di X4, il nastro convogliatore viene fatto avanzare e Y0 viene attivato per il controllo errori. Con il fronte di salita di X5, il nastro convogliatore viene fatto arretrare e Y0 viene disattivato. Il timer T0 controlla il tempo di attivazione. Se il tempo di attivazione impostato viene superato, viene attivata l'istruzione CHKST tramite il contatto TS0. Nel passo di programma successivo, viene eseguita l'istruzione CHK, ed il codice di errore viene memorizzato nel registro speciale SD80.



Il funzionamento dell'istruzione CHK può essere illustrato con i seguenti schemi a contatti, le cui funzioni sono simili a quelle dell'istruzione CHK. Gli indirizzi dei contatti per i finecorsa per i movimenti di avanzamento X_n ed arretramento X_{n+1} sono definiti consecutivamente. L'indirizzo del finecorsa di avanzamento X_n deve essere inferiore all'indirizzo del finecorsa di arretramento X_{n+1} . L'indirizzo del finecorsa di avanzamento viene assegnato ad una uscita Y_n avente lo stesso indirizzo. Secondo il programma di esempio, questa uscita viene attivata durante il movimento di avanzamento e disattivata durante il movimento di arretramento.

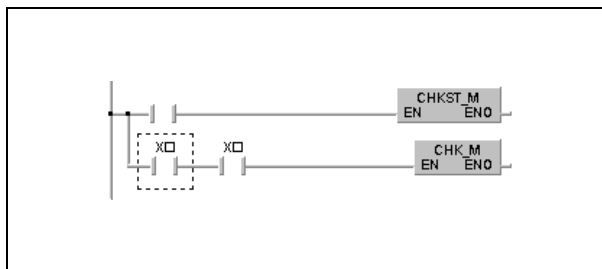
Per una migliore comprensione del programma di esempio precedente, i contatti X_0 (X_n), X_1 (X_{n+1}) e Y_0 (Y_n) sono usati direttamente specificando il numero di bobina. A seconda del programma, possono essere sostituiti con qualsiasi altro numero.

NOTA

Le uscite Y_n sono trattate come relé interni e non possono essere inviate a dispositivi esterni.

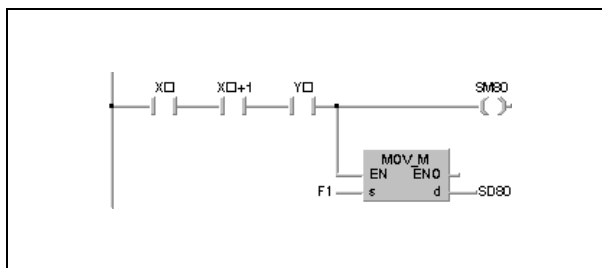
Gli schemi seguenti, relativi alle istruzioni CHK ed ai 6 circuiti di controllo errore generati (condizioni di errore) sono disposti a coppie.

Di seguito, vengono illustrate le istruzioni CHK. Il contatto indicato come X_n può indicare un massimo di 150 contatti (150 nastri convogliatori o applicazioni similari).



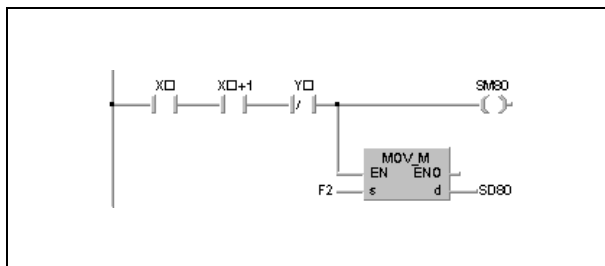
Circuito controllo errori 1 (bobina numero 1):

Entrambi i finecorsa rispondono al movimento di avanzamento del convogliatore.



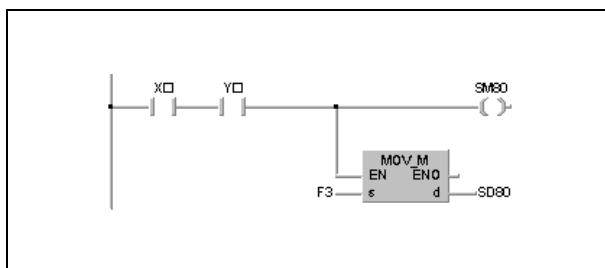
Circuito controllo errori 2 (bobina numero 2):

Entrambi i finecorsa rispondono al movimento di arretramento del convogliatore.



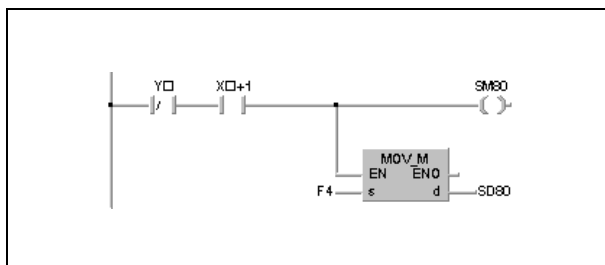
Circuito controllo errori 3 (bobina numero 3):

Comando di avanzamento con nastro su finecorsa avanti.



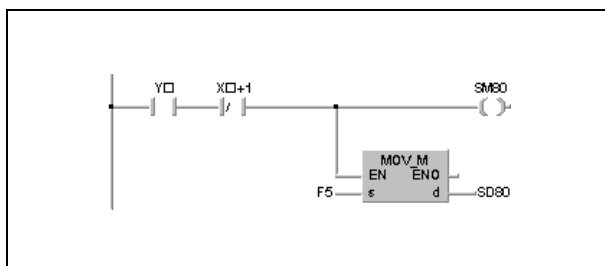
Circuito controllo errori 4 (bobina numero 4):

Comando di arretramento con nastro su finecorsa indietro.



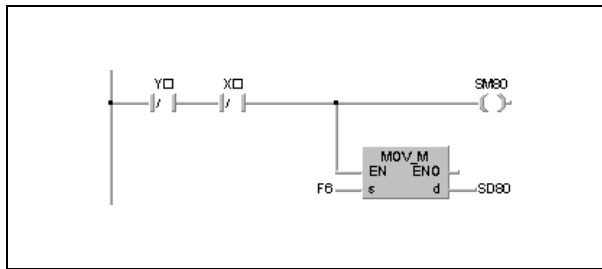
Circuito controllo errori 5 (bobina numero 5):

Comando di avanzamento con finecorsa indietro disattivato.

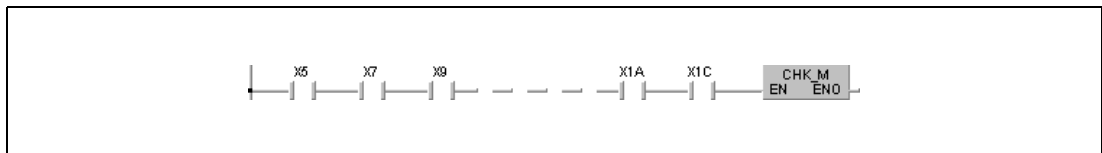


Circuito controllo errori 6 (bobina numero 6):

Comando di arretramento con finecorsa avanti disattivato.



L'istruzione CHK può specificare un massimo di 150 indirizzi di contatti per i finecorsa avanti. Nell'indicazione degli indirizzi dei contatti, vengono scavalcati i numeri relativi ai finecorsa indietro.

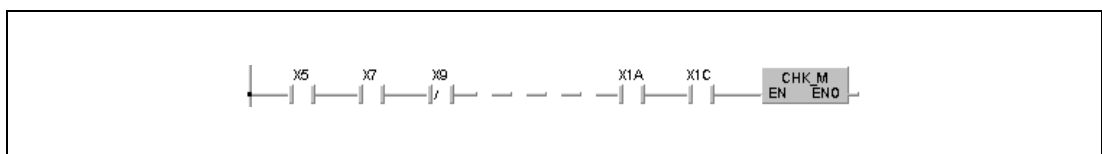


Il relé SM80 ed il registro speciale SD80 devono essere azzerati dopo l'esecuzione dell'istruzione CHK, per non mantenere il loro stato precedente. Se non vengono azzerati prima di un'altra istruzione CHK, l'istruzione non può essere eseguita.

L'istruzione CHKST deve essere programmata prima dell'istruzione CHK.

L'istruzione CHK può essere programmata in qualsiasi passo della sequenza di programma. L'istruzione CHK può essere richiamata per un massimo di due volte in una unità organizzativa di programma (POU).

Gli indirizzi delle bobine devono essere programmate tramite una istruzione LD o AND prima dell'istruzione CHK. Non sono supportate altre istruzioni d'ingresso. Se viene programmata una istruzione LDI o ANI, il controllo di errore dell'istruzione CHK non può essere eseguito. Gli indirizzi dei contatti designati per il controllo di errore possono però essere specificati con istruzioni LDI o ANI. Nello schema seguente il contatto corrispondente a X9 viene ignorato perché è un contatto NC (normalmente chiuso).



Usando una CPU serie o System Q, il metodo di rilevamento degli errori dipende dallo stato del relé speciale SM710 con la logica seguente.

SM710 disattivato (0):

Il controllo errori viene eseguito in sequenza di numero di bobina (circuito controllo errore) dal contatto 1 (finecorsa) al contatto n (finecorsa).

Il primo contatto viene controllato in base ai circuiti di controllo errore da 1 a 6. Poi il contatto successivo viene controllato in base agli stessi circuiti di controllo errore e così via. Il controllo termina quando l'ennesimo contatto è stato controllato in base a tutti i 6 circuiti di controllo errore.

SM710 attivato (1):

Il controllo errori viene eseguito nella sequenza degli indirizzi dei contatti (finecorsa) dal circuito di controllo 1 al circuito 6.

Il primo circuito di controllo, sorveglia i contatti da 1 a n, Poi il circuito di controllo successivo sorveglia dal contatto 1 al contatto n. L'operazione viene completata dopo che il sesto circuito di controllo ha eseguito il test dei contatti da 1 a n.

Se viene rilevato più di un errore, viene memorizzato il numero del primo errore rilevato. Gli errori successivi vengono ignorati.

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Due contatti di controllo errori sono collegati in parallelo nello stesso circuito di controllo (codice di errore 4235).
- Superato il massimo numero di 150 operandi di ingresso (codice di errore 4235).
- Una istruzione CHKST non è seguita da una istruzione CHK (codice di errore 4235).
- Una istruzione CHK viene eseguita prima di una istruzione CHKST (codice di errore 4235).

7.10.2 CHK (solo serie A)

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
● ¹	● ¹	●	●		

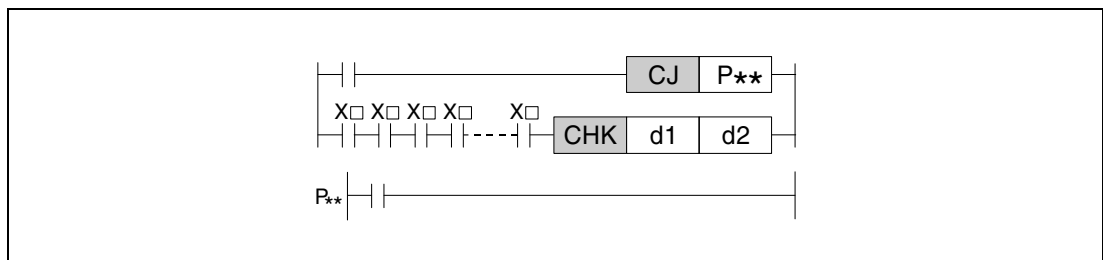
¹ Solo in modo diretto

Operandi MELSEC A

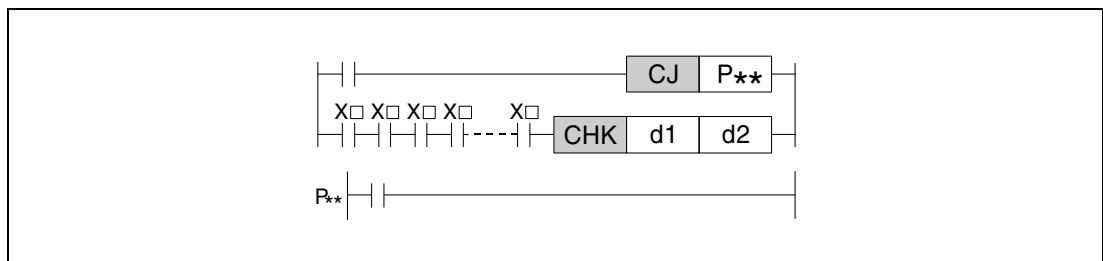
Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Rapporto	Errore			
Operandi a bit					Operandi a word (16-bit)						Costanti		Puntatore		Livello								
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H (16#)	P	I	N	M9012	M9010 M9011	
d1	●	●	●	●	●	●																	
d2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●							● ⁵	

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
d1	Operando da attivare durante il controllo errori.	Bit
d2	Operando contenente il codice di errore	BIN 16-bit

Funzioni Controllo errori per operazioni bidirezionali (solo serie A)**CHK Istruzione controllo anomalie**

Il funzionamento dell'istruzione CHK dipende dalla modalità di controllo I/O selezionata. Usando una CPU A1S e AnN in modalità rinfresco, l'istruzione di controllo genera un flip-flop.

In modalità di controllo I/O diretto (tranne CPU AnA, AnAS, AnU e A2C) l'istruzione di controllo rileva le anomalie nelle operazioni bidirezionali.

A causa del puntatore 254, l'istruzione CHK può essere programmata solo con lista di istruzioni.


In combinazione con alcuni tipi di CPU (ed a seconda della modalità di controllo), l'istruzione CHK supporta il controllo anomalie di circuiti a contatti con finecorsa, per il rilevamento di anomalie in operazioni di movimenti bidirezionali. Se si verifica un errore in questi circuiti, l'operando indicato da d1 si attiva e il codice di errore corrispondente viene memorizzato in d2.

I contatti di ingresso programmati prima dell'istruzione CHK non sono utilizzati come condizione di esecuzione dell'istruzione CHK, ma come specifica delle condizioni di controllo.

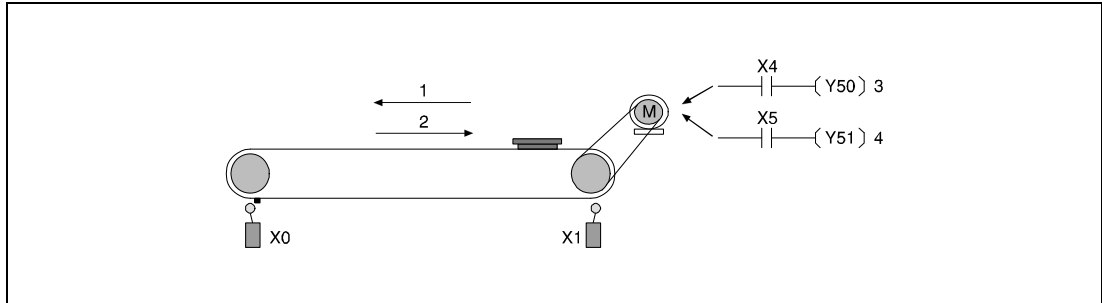
Lo scopo dell'istruzione CHK è quello di rilevare l'occorrenza e la causa di anomalie nell'esecuzione del programma, ad es. se viene superato il tempo di esecuzione impostato. Se non si riscontrano errori durante l'esecuzione del programma, la parte di programma che contiene l'istruzione CHK può essere saltata tramite le istruzioni CJ, SCJ o JMP.

L'istruzione CHK viene eseguita a ciascuna scansione di programma ed è indipendente dallo stato degli operandi di ingresso programmati prima dell'istruzione CHK, come specifica delle condizioni di controllo.

Il programma seguente attiva l'uscita Y60 ed esegue l'istruzione di controllo, se il tempo di esecuzione previsto viene superato. Se l'istruzione CHK rileva una anomalia, M0 viene attivato, ed il programma salta alla destinazione P31 (non indicata). La destinazione P31 (non indicata) può contenere ad esempio una sezione di programma per l'elaborazione dell'errore. Se il tempo di esecuzione non viene superato, la sezione di programma relativa al controllo errori viene saltata, e viene eseguito il passo 18 alla destinazione P30. A causa del puntatore 254, questo programma può essere definito solo con lista di istruzioni.

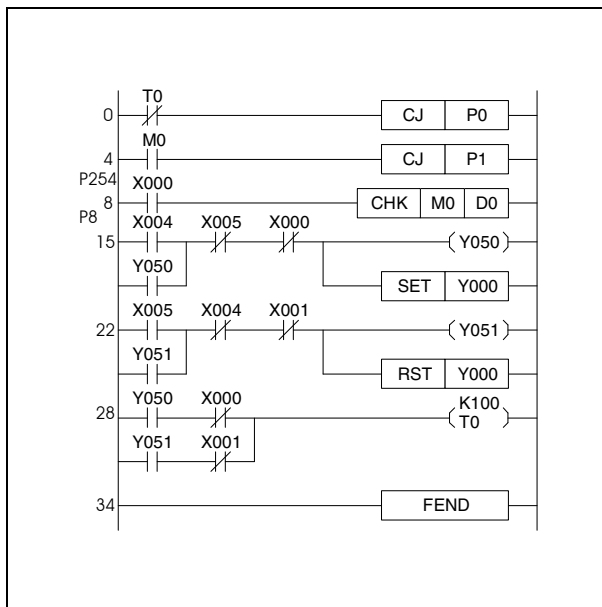
100	LDI	Y060	
101	CJ	P30	
104	LD	M0	
105	CJ	P30	
108	P254		
109	LD	X010	
110	AND	X015	
111	AND	X008	
112	AND	X01A	
	113	CHK	M0 D0
	118	P30	
	118	LD	M10
	119	OUT	Y040
	121	END	

Di seguito, la programmazione del controllo anomalie tramite l'istruzione CHK viene illustrata con un esempio concreto. La figura seguente mostra un nastro trasportatore che si muove da sinistra, fino al finecorsa destro. I finecorsa relativi vengono rilevati tramite interruttori di finecorsa (X0 e X1). Il contatto di avviamento per il movimento di avanzamento è X4 e quello per il movimento di arretramento è X5.

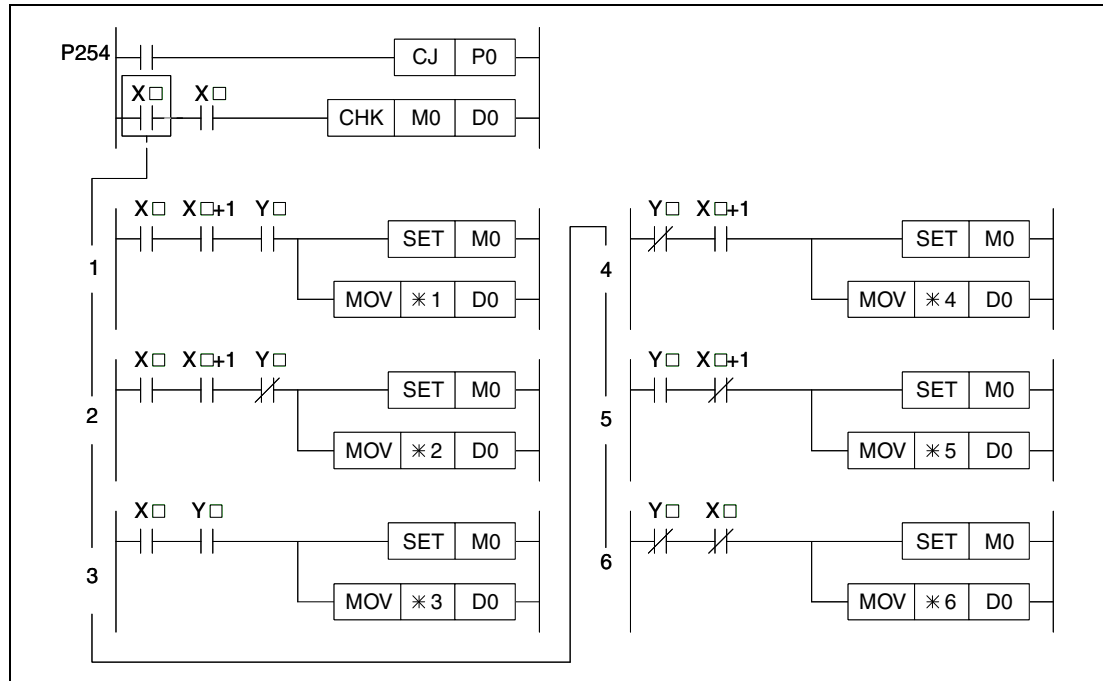


- 1 Movimento di avanzamento
- 2 Movimento di arretramento
- 3 Comando avanzamento
- 4 Comando arretramento

La figura seguente mostra un programma di esempio per il funzionamento e il controllo errori del convogliatore mostrato in precedenza. A causa del puntatore 254, questo programma può essere definito solo con lista di istruzioni o con l'editor a contatti di GX Developer. Nel funzionamento senza errori, il programma salta alla destinazione P0. Se X4 è attivo, il nastro convogliatore avanza e Y0 viene attivato per il controllo anomalie. Se X5 è attivo, il nastro convogliatore viene fatto arretrare e Y0 viene disattivato. Il timer T0 controlla il tempo di attivazione. Se viene superato il tempo impostato, M0 viene attivato tramite l'istruzione CHK ed il codice di errore viene memorizzato in D0. L'esecuzione del programma prosegue per un ulteriore controllo anomalie alla destinazione di salto P1 (passo 35).



Il funzionamento dell'istruzione CHK può essere illustrato con il seguente schema a contatti, le cui funzioni sono simili a quelle dell'istruzione CHK. Per una migliore comprensione dell'esempio di cui sopra, i contatti X0, X1 e Y0 sono indicati direttamente per definire le condizioni di controllo. A seconda del programma, possono essere sostituiti con qualsiasi altro indirizzo.



Si possono verificare le seguenti condizioni di errore:

Condizione 1: Entrambi i finecorsa sono attivati mentre il convogliatore avanza.

Condizione 2: Entrambi i finecorsa sono attivati mentre il convogliatore arretra.

Condizione 3: Comando di avanzamento con nastro su finecorsa avanti.

Condizione 4: Comando di arretramento con nastro su finecorsa indietro.

Condizione 5: Comando di avanzamento con finecorsa indietro disattivato.

Condizione 6: Comando di arretramento con finecorsa avanti disattivato.

Il numero del codice di errore memorizzato in D0 corrisponde al numero della condizione di errore precedente.

L'istruzione CHK esegue il controllo anomalie in base allo schema mostrato sopra. Lo schema non può essere cambiato.

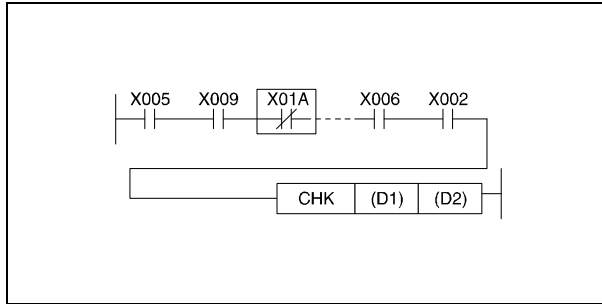
Gli operandi in d1 e d2 devono essere azzerati prima dell'esecuzione dell'istruzione CHK, dato che mantengono lo stato assunto a seguito di una precedente istruzione CHK. Se questi operandi rimangono attivi, l'istruzione CHK non può essere eseguita.

Il puntatore P254 deve essere sempre specificato come istruzione di destinazione in testa all'istruzione CHK. Questo puntatore indica l'inizio di un controllo anomalie.

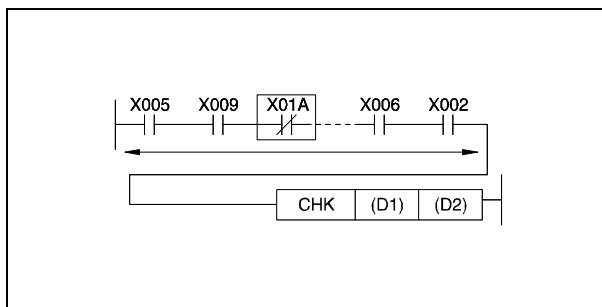
L'istruzione CHK può essere programmata in qualsiasi passo della sequenza di programma. Tuttavia è possibile programmarla solo una volta in ciascun programma.

L'istruzione CHK non può essere scritta nel modo RUN della CPU.

Le condizioni di controllo devono essere programmate tramite una istruzione LD o AND prima dell'istruzione CHK. Le condizioni di controllo non possono essere definite con altri comandi relativi ai contatti. Se viene utilizzata una istruzione ANI per impostare la condizione di controllo, il controllo anomalie non viene eseguito.



Il controllo anomalie viene eseguito nell'ordine dei contatti di ingresso che sono indicato come variabili da controllare. Se viene rilevato più di un errore, viene memorizzato solo l'errore a priorità più alta.



Il codice di errore memorizzato in d2 dipende dalla condizione di errore riscontrata e dall'indirizzo del contatto

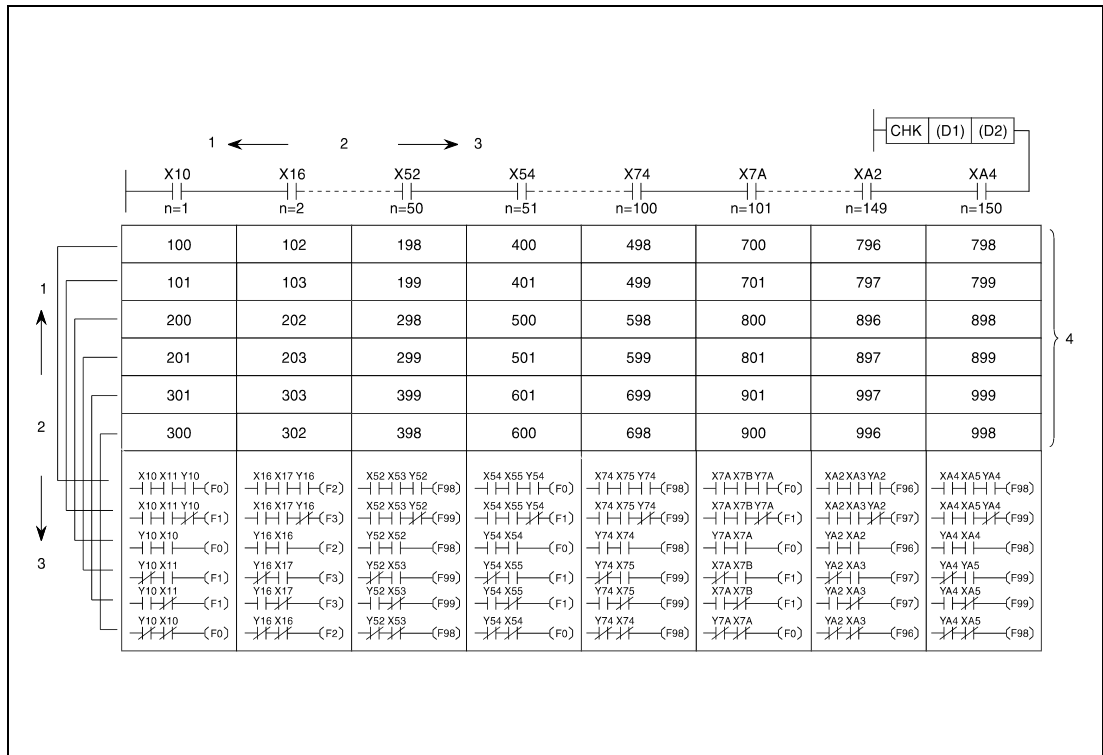
Numero condizione di errore	Indirizzo contatto di ingresso da 1 a 50	Indirizzo contatto di ingresso da 51 a 100	Indirizzo contatto di ingresso da 101 a 150
1 (codice di errore 1)	$100 + (2 \times (\text{num. contatto} - 1))$	$400 + (2 \times (\text{num. contatto} - 1))$	$700 + (2 \times (\text{num. contatto} - 1))$
2 (codice di errore 2)	$101 + (2 \times (\text{num. contatto} - 1))$	$401 + (2 \times (\text{num. contatto} - 1))$	$701 + (2 \times (\text{num. contatto} - 1))$
3 (codice di errore 3)	$200 + (2 \times (\text{num. contatto} - 1))$	$500 + (2 \times (\text{num. contatto} - 1))$	$800 + (2 \times (\text{num. contatto} - 1))$
4 (codice di errore 4)	$201 + (2 \times (\text{num. contatto} - 1))$	$501 + (2 \times (\text{num. contatto} - 1))$	$801 + (2 \times (\text{num. contatto} - 1))$
5 (codice di errore 5)	$300 + (2 \times (\text{num. contatto} - 1))$	$600 + (2 \times (\text{num. contatto} - 1))$	$900 + (2 \times (\text{num. contatto} - 1))$
6 (codice di errore 6)	$301 + (2 \times (\text{num. contatto} - 1))$	$601 + (2 \times (\text{num. contatto} - 1))$	$901 + (2 \times (\text{num. contatto} - 1))$

- ¹ Contatto numero 1
- ² Contatto numero 50
- ³ Contatto numero 51
- ⁴ Contatto numero 100
- ⁵ Contatto numero 101
- ⁶ Contatto numero 150

I numeri dei codici di errore visualizzati dopo l'esecuzione dell'istruzione CHK, indicano il tipo di errore intervenuto. Per una diagnosi veloce, è consigliabile approntare una tabella di ricerca guasti.

Codice di errore	Causa	Azione correttiva
301	Convogliatore 1: Avvio arretramento con fincorsa avanti non premuto	- controllare contatto X1 - controllare convogliatore
302	Convogliatore 1:
...

Panoramica codici di errore

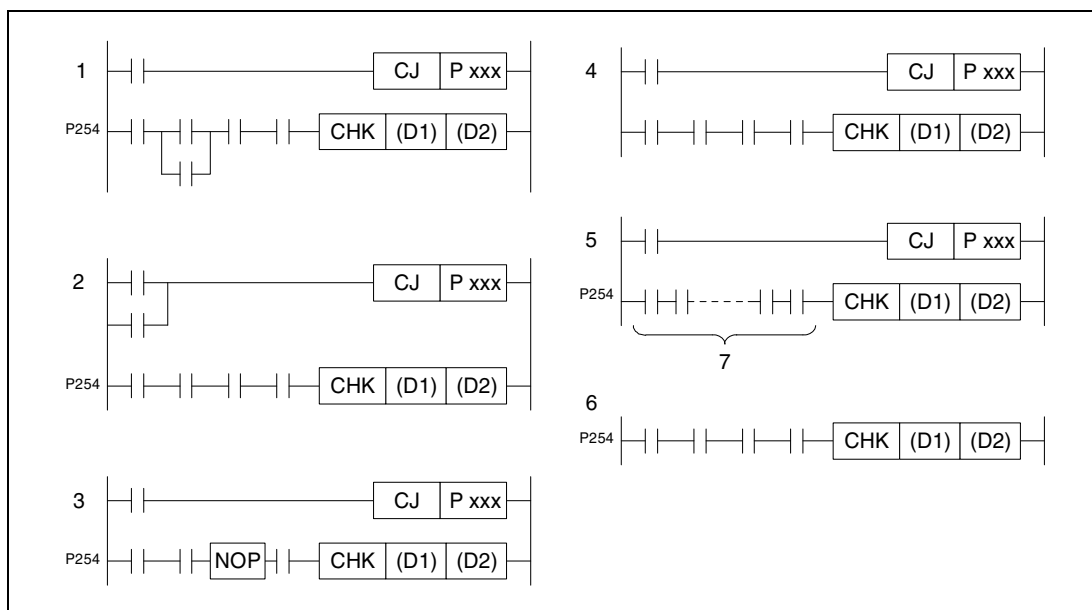


- 1 alta priorità
- 2 priorità
- 3 bassa priorità
- 4 codice di errore

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato (i numeri iparentesi si riferiscono agli schemi seguenti):

- Due contatti di ingresso sono collegati in parallelo nelle condizioni di ingresso (1) o in testa all'istruzione CJ (2).
- Una istruzione NOP programmata fra le condizioni di controllo dell'istruzioni CHK (3).
- La destinazione di salto P254 non esiste nel programma (4).
- Le condizioni di controllo dell'istruzione CHK contengono più di 150 operandi (5).
- Nessuna istruzione di salto (CJ) prima dell'istruzione CHK (CJ)(6).



⁷ Più di 150 contatti di ingresso

7.10.3 CHKCIR, CHKEND

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

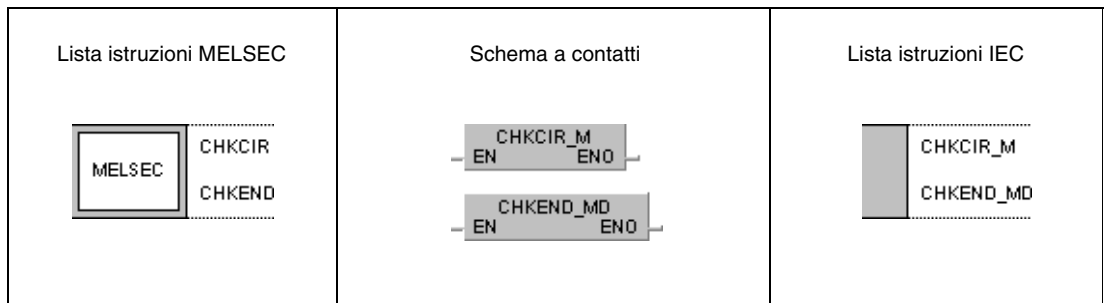
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR (solo CHKEND).

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

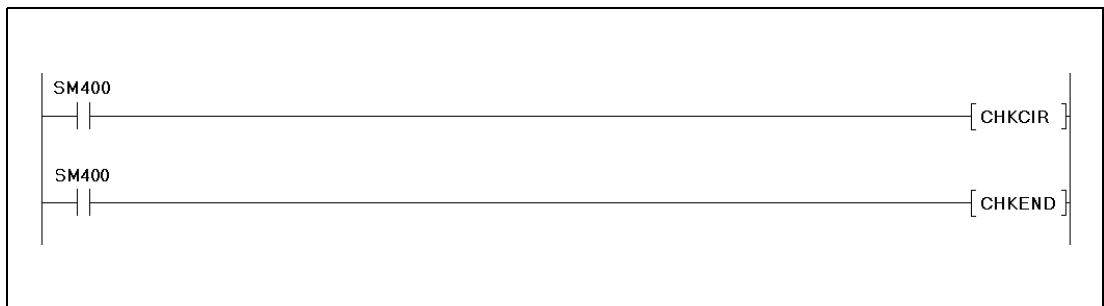
Operandi MELSEC Q

Operandi utilizzabili										Errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti	Altro			
Bit	Word		Bit	Word							
—	—	—	—	—	—	—	—	—	—	SM0	1

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
—	—	—

Funzioni **Generazione circuiti di controllo per istruzione CHK**

CHKCIR, CHKEND **Istruzioni di inizio e fine di una sezione di programma con circuiti di controllo generati.**

Le istruzioni CHKCIR e CHKEND modificano i circuiti di controllo dell'istruzione CHK. Può essere generato qualsiasi schema di controllo richiesto. L'effettivo controllo anomalie viene eseguito dalle istruzioni CHKST e CHK.

Il controllo anomalie viene eseguito in base allo schema di controllo programmato fra le istruzioni CHK e CHKEND.

NOTA

Se lo schema del circuito di controllo per l'istruzione CHK viene modificato con le istruzioni CHKCIR e CHKEND, i dispositivi periferici collegati devono essere avviati in "General Mode", e deve essere eseguita una espansione di programma.

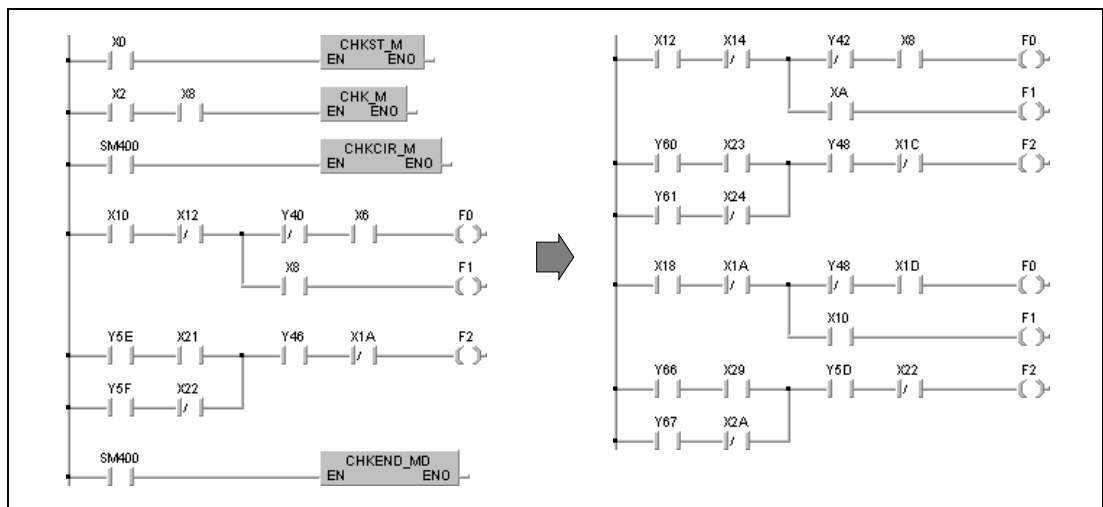
Nei casi in cui un dispositivo periferico viene avviato da una CPU Q2A, Q2AS, Q3A o Q4A, ed è stato effettuato un tentativo di generare dei circuiti di controllo anomalie modificati per l'istruzione CHK, tramite le istruzioni CHKCIR e CHKEND, non è possibile assicurare un funzionamento corretto.

Partendo dai circuiti di controllo definiti fra le istruzioni CHKCIR e CHKEND, vengono generati dei circuiti di controllo modificati, tramite indicizzazione. I circuiti di controllo anomalie programmati fra queste istruzioni possono utilizzare fino a 9 spie di errore (da F1 a F9). L'indicizzazione avviene sommando l'indirizzo specificato prima dell'istruzione CHK, con l'indirizzo dei contatti dei circuiti di controllo anomalie. Ad esempio, al contatto X10 dei circuiti di controllo anomalie, vengono assegnati gli indirizzi X12 e X18 nei circuiti di controllo indicizzati, in base ai contatti X2 e X8 programmati prima dell'istruzione CHK.

L'algorithmo di controllo anomalie dipende dallo stato del relé speciale SM710, come segue:

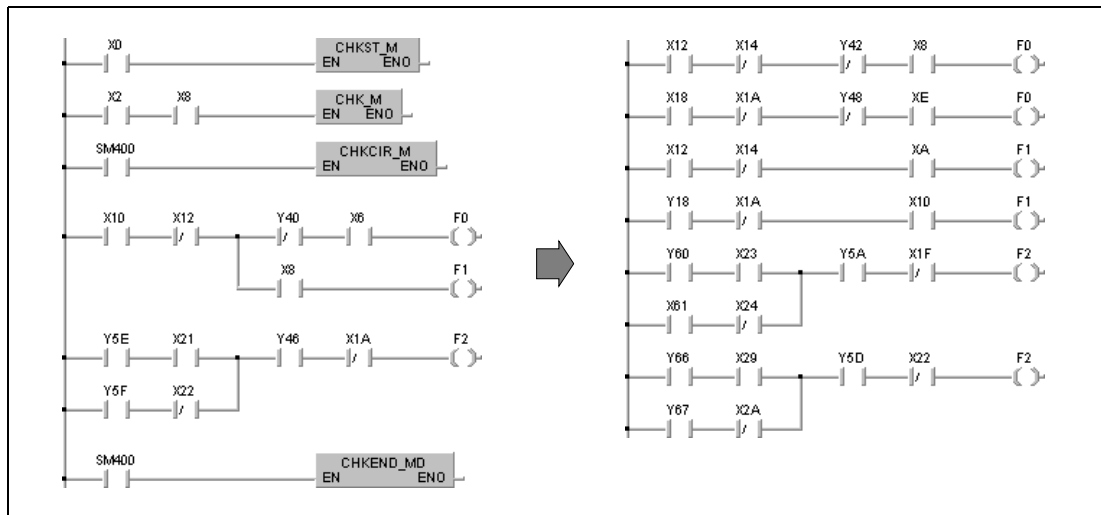
SM710 disattivato (0):

In questo caso, per prima cosa gli indirizzi dei contatti dei circuiti di controllo anomalie programmati fra le istruzioni CHKCIR e CHKEND vengono indicizzati con l'indirizzo del primo contatto programmato prima dell'istruzione CHK. Poi, ciascun circuito di controllo programmato viene indicizzato nuovamente con l'indirizzo del secondo contatto progettato prima dell'istruzione CHK. Questa operazione viene ripetuta per tutti i circuiti di controllo programmati che usano spie di errore (F) per un totale di nuovi circuiti di controllo uguale al numero dei contatti d'ingresso dell'istruzione CHK.



SM710 attivato (1):

Per prima cosa in questo caso, il primo circuito di controllo anomalia programmato che usa spie di errore, viene indicizzato con tutti gli indirizzi dei contatti programmati prima dell'istruzione CHK. Poi, il successivo circuito di controllo viene indicizzato con tutti gli indirizzi dei contatti programmati prima dell'istruzione CHK. Questa operazione viene ripetuta per tutti i circuiti di controllo programmati che usano spie di errore (F) per un totale di nuovi circuiti di controllo uguale al numero dei contatti d'ingresso dell'istruzione CHK.



Durante l'elaborazione dei circuiti di controllo anomalie indicizzati, viene controllato lo stato delle uscite (F), che possono essere attivate solo con l'istruzione OUT. Se una uscita (F) è attiva, viene attivato il relé speciale SM80. Il codice di errore, composto dall'indirizzo del contatto e dal circuito di controllo anomalia (F1 - F9) viene memorizzato nel registro speciale SD80 in formato BCD.

I circuiti di controllo anomalia fra le istruzioni CHKCIR e CHKEND possono essere programmati usando le istruzioni seguenti:

Contatti:

LD, LDI, AND, ANI, OR, ORI, ANB, ORB, MPS, MPP, MRD, istruzioni di confronto.

Bobine:

OUT F

Gli operandi utilizzati per specificare i contatti possono essere programmati usando gli ingressi X e le uscite Y.

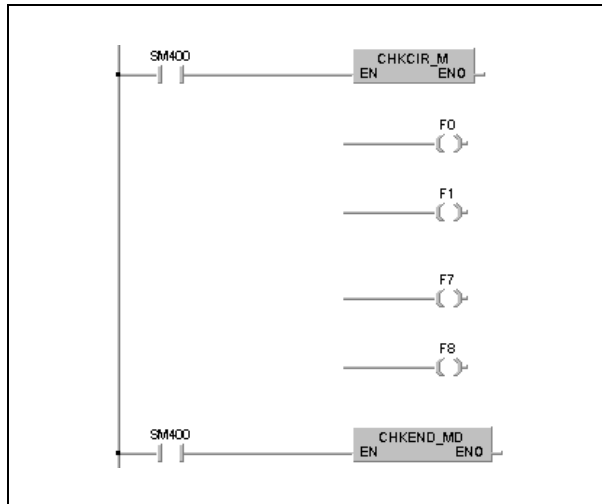
Come uscite dei circuiti di controllo anomalie, possono essere utilizzate solo spie di errore (F). I circuiti di controllo anomalie possono essere programmati con indicazione casuale delle uscite, partendo da F0, dato che queste uscite vengono considerate come contatti dummy. Per questa ragione non viene emessa nessuna segnalazione di errore se le spie di errore (F) sono sovrapposte.

Lo stato delle spie di errore (F) può essere determinato con precisione anche se una spia (F) viene programmata una seconda volta oltre l'istruzione CHK, dato che queste spie vengono elaborate separatamente.

Dato che lo stato (0/1) delle spie di errore (F) usate dall'istruzione CHK non viene aggiornato, le spie compaiono come disattivate se monitorate da un dispositivo periferico.

I circuiti di controllo anomalie programmati fra le istruzioni CHKCIR e CHKEND possono essere creati con un massimo di 256 passi di programma (rami di contatti) e 9 uscite (spie di errore F1-F9) comandate tramite istruzioni OUT F.

I circuiti di controllo anomalie fra le istruzioni CHKCIR e CHKEND sono ordinati dal circuito di controllo anomalia più alto (1 - F0) a quello più basso (9 - F8).



Le istruzioni CHKCIR e CHKEND possono essere programmate in qualsiasi passo della sequenza di programma. Queste istruzioni possono essere utilizzate complessivamente due volte in tutti i file di programma eseguiti, ed una sola volta nello stesso file di programma.

Le istruzioni CHKCIR e CHKEND non possono essere utilizzate in programmi a bassa velocità, per non generare errori di funzionamento ed arrestare l'elaborazione della CPU.

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Le istruzioni CHKCIR e CHKEND compaiono oltre due volte in tutti i file di programma (codice di errore 4235).
- Le istruzioni CHKCIR e CHKEND compaiono più di una volta in un file di programma (codice di errore 4235).
- L'istruzione CHKEND non viene eseguita dopo l'istruzione CHKCIR (codice di errore 4230).
- L'istruzione CHKEND viene eseguita senza una precedente istruzione CHKCIR (codice di errore 4230).
- Le istruzioni CHKCIR e CHKEND sono programmate in un programma a bassa velocità (codice di errore 4235).
- Più di 9 spie di errore (F) (circuiti di controllo anomalie) utilizzate (codice di errore 4235).
- I circuiti di controllo anomalie creati contengono più di 256 passi di programma (rami di contatti) (codice di errore 4235).
- I circuiti di controllo anomalie contengono operandi non validi (codice di errore 4235).
- I circuiti di controllo anomalie contengono operandi già indicizzati (codice di errore 4235).

NOTA

Gli errori seguenti, che avvengono durante l'espansione di programma su un dispositivo periferico, impediscono l'esecuzione dell'espansione di programma:

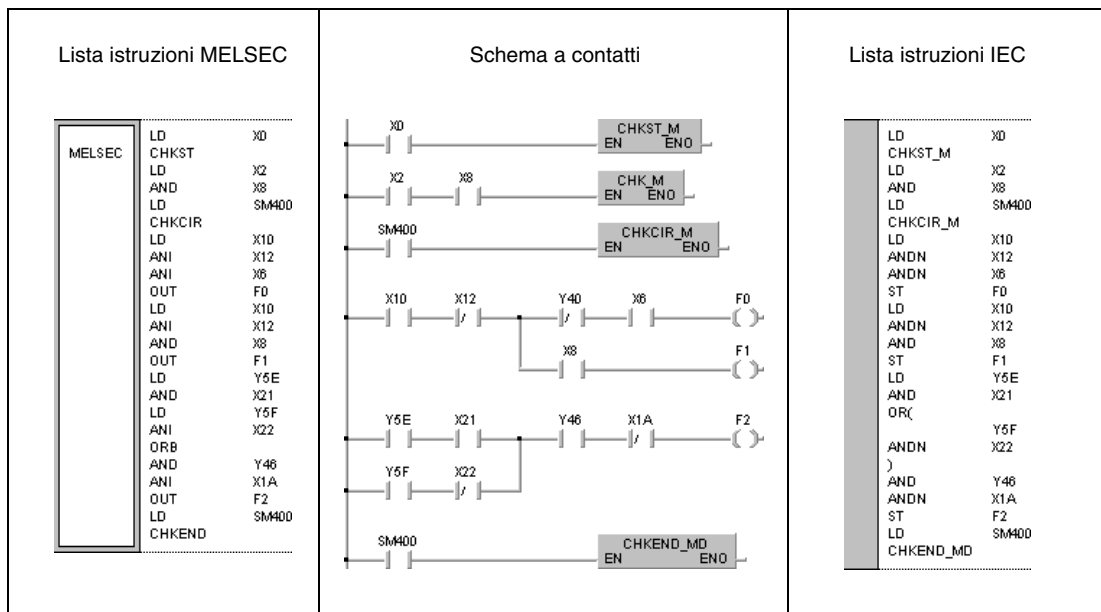
- *I circuiti di controllo anomalie contengono operandi errati.*
- *I circuiti di controllo anomalie contengono operandi già indicizzati.*

Correggere appropriatamente i circuiti di controllo anomalie, in caso si riscontri qualcuno degli errori riportati

Esempio di programma

CHKCIR, CHKEND

Il programma seguente crea circuiti di controllo anomalie indicizzati. Il funzionamento di questo programma è illustrato alla voce "Funzioni". Inoltre, vengono riportate di seguito le liste istruzioni MELSEC e IEC.



7.10.4 SLT, SLTR

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	● ¹	●	●	●	

¹ Tranne CPU A1N.

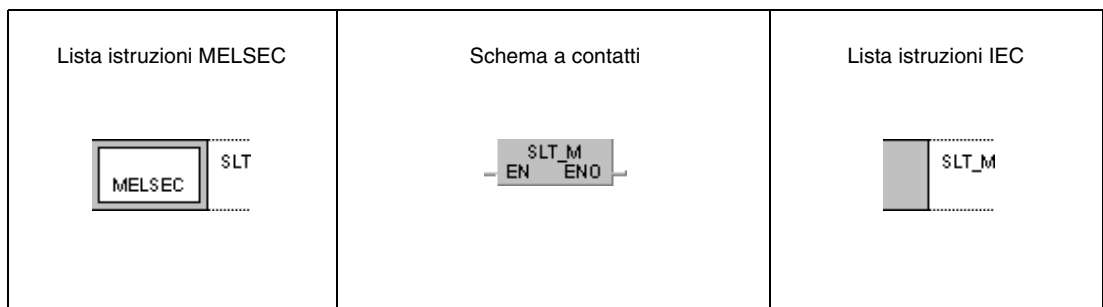
Operandi MELSEC A

Operandi utilizzabili														Definizione cifra	Numero di passi	Indice	Riporto	Errore			
Operandi a bit				Operandi a word (16-bit)						Costanti	Punta-tore	Livello									
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1						Z	V	K
															1						

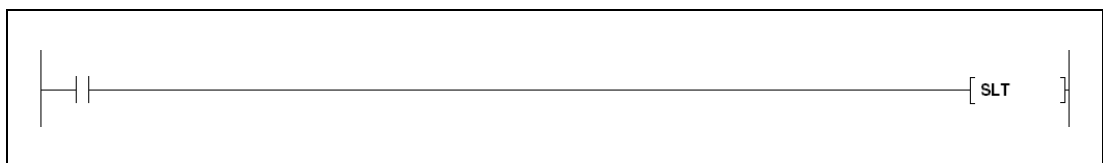
Operandi MELSEC Q

Operandi utilizzabili										Errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti	Altro			
Bit	Word		Bit	Word							
—	—	—	—	—	—	—	—	—	—	—	1

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
—	—	—

Funzioni Attivazione e disattivazione latch di stato

SLT Attiva memoria di stato

Anche quando l'esecuzione del programma viene monitorata con GX IEC Developer, non tutti gli stati degli operandi possono essere trasmessi e visualizzati. Per questa ragione le CPU contengono una zona di memoria speciale, la memoria di stato. La memoria di stato viene definita da dati di configurazione e memorizza i dati di una scansione di programma (fare riferimento ai manuali di GX Developer per ulteriori dettagli).

L'istruzione SLT memorizza temporaneamente il dato dell'operando specificato. Il dato viene inserito nella memoria di stato a ritenuta e può essere controllata e visualizzata.

L'istruzione SLT può essere eseguita una sola volta in una scansione del programma. Una ulteriore esecuzione dell'istruzione SLT può essere riabilitata solo tramite l'istruzione SLTR.

SLTR Cancella memoria di stato

L'istruzione SLTR cancella i dati memorizzati temporaneamente nell'area degli stati ritenuti e ripristina (riabilita) l'istruzione SLT.

L'istruzione SLT può essere eseguita una sola volta in una scansione del programma. Una ulteriore esecuzione dell'istruzione SLT può essere riabilitata solo tramite l'istruzione SLTR.

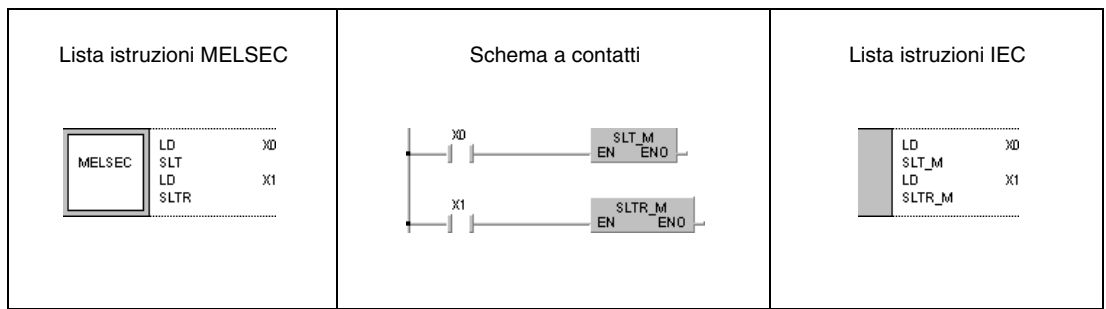
NOTA *Controllare se queste funzioni sono disponibili e supportate dalla versione di GX IEC Developer utilizzata.*

Fare riferimento al manuale utente delle CPU e di GX Developer per ulteriori dettagli sul funzionamento della memoria di stato.

L'esecuzione dell'istruzione SLT aumenta il tempo di scansione del programma, in funzione del tipo di CPU. Il valore di impostazione del timer di watchdog deve essere scelto in base all'aumento del tempo di scansione del programma. Fare riferimento al manuale utente della CPU appropriata per una stima dell'aumento del tempo di elaborazione.

Esempio di programma SLT/SLTR

Se X0 è attivo, il programma seguente esegue l'istruzione SLT. Fino a quando X1 è attivo, l'istruzione SLTR azzerava l'istruzione SLT.



7.10.5 STRA, STRAR

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	

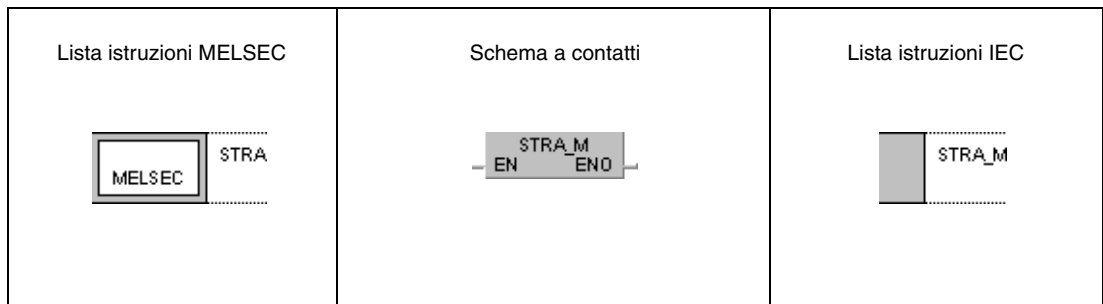
Operandi MELSEC A

Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Rapporto	Errore					
Operandi a bit						Operandi a word (16-bit)						Costanti		Puntatore							Livello				
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V				K	H (16#)	P	I	N	M9012	M9010 M9011
																							1		

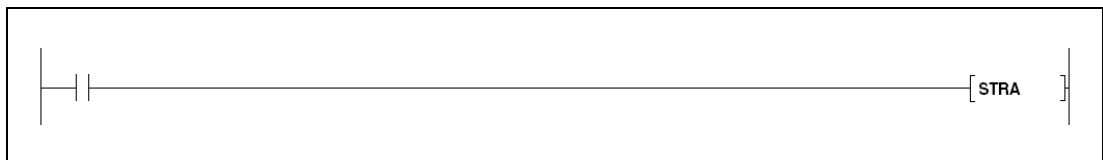
Operandi MELSEC Q

Operandi utilizzabili										Errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti	Altro			
Bit	Word		Bit	Word							
—	—	—	—	—	—	—	—	—	—	—	1

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
—	—	—

Funzioni Attivazione e disattivazione traccia di campionamento

STRA Attiva traccia di campionamento

La traccia di campionamento sorveglia dati e stato di operandi specifici per un determinato periodo di tempo, e memorizza i dati accumulati relativi agli operandi indicati in una zona di memoria separata. La scelta degli operandi e del periodo della traccia avviene tramite parametrizzazione.

STRAR Disattiva traccia di campionamento

L'istruzione STRAR cancella i dati dal file dei dati campionati e azzerava l'istruzione STRA ed il relé speciale M9043 (serie A) o i relé speciali da SN801 a SM805 (serie Q e System Q).

L'istruzione STRA può essere eseguita una sola volta dopo l'esecuzione dell'istruzione STRAR.

NOTA *Controllare se queste funzioni sono disponibili e supportate dalla versione di GX IEC Developer utilizzata.*

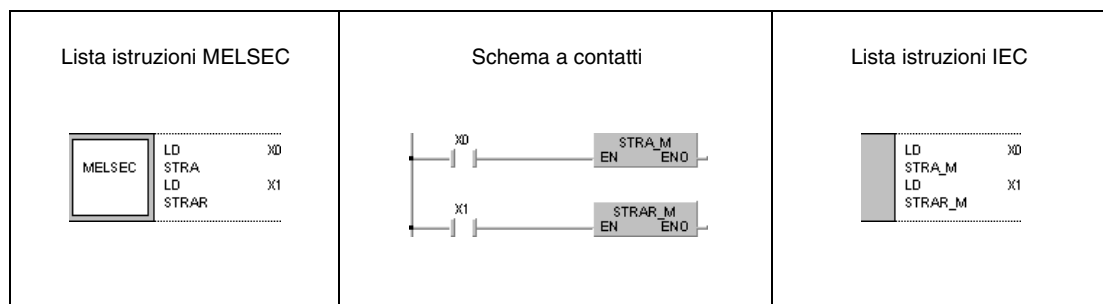
Fare riferimento al manuale utente delle CPU e di GX Developer per ulteriori dettagli sul funzionamento della traccia di campionamento.

L'esecuzione dell'istruzione SLT aumenta il tempo di scansione del programma, in funzione del tipo di CPU. Il valore di impostazione del timer di watchdog deve essere scelto in base all'aumento del tempo di scansione del programma. Fare riferimento al manuale utente della CPU appropriata per una stima dell'aumento del tempo di elaborazione.

Le istruzioni STRA e STRAR non possono essere elaborate durante l'accesso a una ROM (solo serie A).

Esempio di programma STRA/STRAR

Se X0 è attivo, il programma seguente esegue l'istruzione STRA. Fino a quando X1 è attivo, l'istruzione SLTR azzerava l'istruzione STRA.



7.10.6 PTRA, PTRAR, PTRAEXE, PTRAEXEP

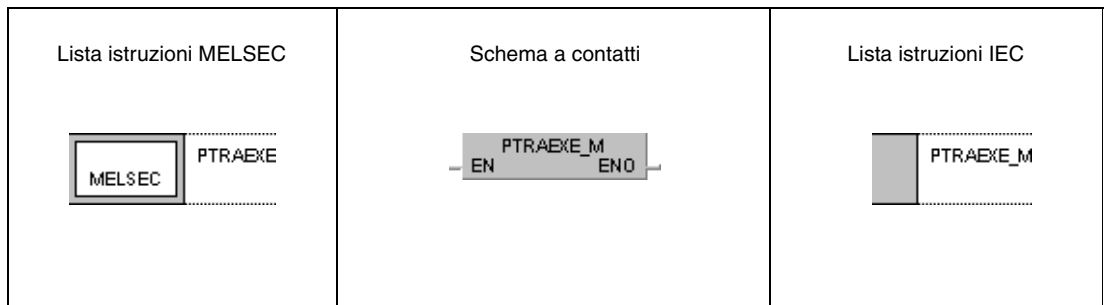
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	

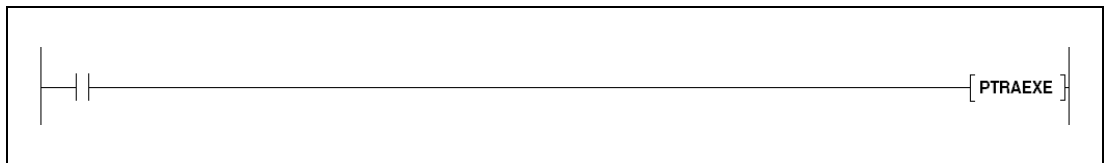
Operandi MELSEC Q

Operandi utilizzabili										Errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti	Altro			
Bit	Word		Bit	Word							
—	—	—	—	—	—	—	—	—	—	—	1

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
—	—	—

Funzioni Attivazione, disattivazione ed esecuzione traccia di programma**PTRA Attiva la traccia di programma**

La traccia di programma sorveglia dati e stato di operandi specifici per un determinato periodo di tempo, e memorizza i dati accumulati relativi ai programmi tracciati in una zona di memoria separata.

L'istruzione PTRA consente di tracciare programmi per un determinato numero di scansioni, memorizzando contemporaneamente i dati in una zona di memoria separata della CPU dedicata alla traccia programmi. L'istruzione PTRAEEXE avvia la memorizzazione dei dati di programma. I relé speciali da SM810 a SM812 devono essere attivi (1) per attivare l'acquisizione dati.

A seguito dell'esecuzione dell'istruzione PTRA, il relé speciale SM813 viene attivato. Dopo aver eseguito il numero di scansioni specificato, i dati vengono memorizzati per una ulteriore elaborazione e la traccia di programma termina.

Se il relé speciale SM811 non è attivo durante la traccia di programma, l'operazione di traccia viene arrestata.

A seguito dell'esecuzione dell'istruzione PTRA, il relé speciale SM815 viene attivato.

Prima di poter nuovamente eseguire l'istruzione PTRA, è necessario eseguire l'istruzione PTRAR.

I risultati dell'operazione di traccia programma possono essere visualizzati tramite un dispositivo periferico.

PTRAR Disattiva la traccia di programma

L'istruzione PTRAR cancella i dati contenuti nel file della traccia di programma e cancella l'istruzione PTRA ed i relé speciali da SM811 a SM815.

L'istruzione PTRA può essere eseguita una sola volta dopo l'esecuzione dell'istruzione PTRAR.

PTRAEEXE Esegue la traccia di programma.

L'istruzione PTRAEEXE avvia l'esecuzione della scansione del programma.

Se il relé speciale SM811 non è attivo durante la traccia di programma, l'operazione di traccia viene arrestata.

Se la condizione di esecuzione dell'istruzione PTRAEEXE non è attiva, la traccia di programma non viene eseguita.

NOTA *Controllare se queste funzioni sono disponibili e supportate dalla versione di GX Developer utilizzata.*

Fare riferimento al manuale utente delle CPU e di GX Developer per ulteriori dettagli sul funzionamento della traccia di programma.

7.11 Istruzioni di elaborazione stringhe di caratteri

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Conversione di dati binari a 16-/32-bit in valori decimali in codice ASCII	BINDA	BINDA_MD
		BINDA_K_MD
		BINDA_S_MD
	BINDAP	BINDA_P_MD
		BINDA_K_P_MD
		BINDA_P_S_MD
	DBINDA	DBINDA_MD
		DBINDA_K_P_MD
		DBINDA_P_S_MD
	DBINDAP	DBINDA_P_MD
		DBINDA_K_P_MD
		DBINDA_P_S_MD
Conversione di dati binari a 16-/32-bit in Codice ASCII	BINHA	BINHA_MD
		BINHA_K_MD
		BINHA_S_MD
	BINHAP	BINHA_P_MD
		BINHA_K_P_MD
		BINHA_P_S_MD
	DBINHA	DBINHA_MD
		DBINHA_K_MD
		DBINHA_S_MD
	DBINHAP	DBINHA_P_MD
		DBINHA_K_P_MD
		DBINHA_P_S_MD
Conversione di dati BCD a 4/8 cifre in Codice ASCII	BCDDA	BCDDA_MD
		BCDDA_K_MD
		BCDDA_S_MD
	BCDDAP	BCDDA_P_MD
		BCDDA_K_P_MD
		BCDDA_P_S_MD
	DBCDDA	DBCDDA_MD
		DBCDDA_K_MD
		DBCDDA_S_MD
	DBCDDAP	DBCDDA_P_MD
		DBCDDA_K_P_MD
		DBCDDA_P_S_MD

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Conversione di dati ASCII decimali in dati binari a 16-/32-bit	DABIN	DABIN_MD
		DABIN_S_MD
	DABINP	DABIN_P_MD
		DABIN_P_S_MD
	DDABIN	DDABIN_MD
		DDABIN_S_MD
	DDABINP	DDABIN_P_MD
		DDABIN_P_S_MD
Conversione di dati ASCII esadecimali in dati binari a 16-/32-bit	HABIN	HABIN_MD
		HABIN_S_MD
	HABINP	HABIN_P_MD
		HABIN_P_S_MD
	DHABIN	DHABIN_MD
		DHABIN_S_MD
	DHABINP	DHABIN_P_MD
		DHABIN_P_S_MD
Conversione di dati ASCII decimali in dati BCD a 4/8 cifre	DABCD	DABCD_MD
		DABCD_S_MD
	DABCDP	DABCD_P_MD
		DABCD_P_S_MD
	DDABCD	DDABCD_MD
		DDABCD_S_MD
	DDABCDP	DDABCD_P_MD
		DDABCD_P_S_MD
Lettura di commenti	COMRD	COMRD_MD
		COMRD_S_MD
	COMRDP	COMRD_P_MD
		COMRD_P_S_MD
Rilevamento lunghezza stringa di caratteri	LEN	LEN_E
		LEN_MD
		LEN_S_MD
	LENP	LEN_P_S_MD
Conversione di dati binari a 16-/32-bit in stringhe di caratteri	STR	STR_MD
		STR_K_MD
		STR_S_MD
	STRP	STR_P_MD
		STR_K_P_MD
		STR_P_S_MD
	DSTR	DSTR_MD
		DSTR_K_MD
		DSTR_S_MD
	DSTRP	DSTR_P_MD
DSTR_K_P_MD		
DSTR_P_S_MD		

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Conversione di stringhe di caratteri a dati BIN a 16/32 bit	VAL	VAL_MD
		VAL_S_MD
	VALP	VAL_P_MD
		VAL_P_S_MD
	DVAL	DVAL_MD
		DVAL_S_MD
DVALP	DVAL_P_MD	
	DVAL_P_S_MD	
Conversione di un dato in virgola mobile in una stringa di caratteri	ESTR	ESTR_M
	ESTRP	ESTRP_M
Conversione di una stringa di caratteri in un dato in virgola mobile	EVAL	EVAL_M
	EVALP	EVALP_M
Conversione di stringhe di caratteri alfanumeriche in codice ASCII	ASC	ASC_MD
		ASC_K_MD
		ASC_S_MD
	ASCP	ASC_P_MD
		ASC_P_S_MD
		ASC_K_P_MD
Conversione di valore ASCII esadecimale in valori binari	HEX	HEX_S_MD
		HEX_MD
		HEX_K_MD
	HEXP	HEX_P_S_MD
		HEX_P_MD
		HEX_K_P_MD
Estrazione di parti di stringhe di caratteri (parte destra di una stringa di caratteri)	RIGHT	RIGHT_M
		RIGHT
		RIGHT_E
	RIGHTP	RIGHTP_M
Estrazione di parti di stringhe di caratteri (parte sinistra di una stringa di caratteri)	LEFT	LEFT_M
		LEFT
		LEFT_E
	LEFTP	LEFTP_M
Estrazione di parti qualsiasi di una stringa di caratteri	MIDR	MIDR_M
	MIDRP	MIDRP_M
Selezione e spostamento di parti di una stringa di caratteri in una stringa di caratteri	MIDW	MIDW_M
	MIDWP	MIDWP_M
Ricerca di stringhe di caratteri	INSTR	INSTR_M
	INSTRP	INSTRP_M
Conversione dati in virgola mobile con rappresentazione BCD	EMOD	EMOD_M
	EMODP	EMODP_M
Conversione di dati BCD con formato decimale in virgola mobile	EREXP	EREXP_M
	EREXPP	EREXPP_M

7.11.1 BINDA, BINDAP, DBINDA, DBINDAP

CPU

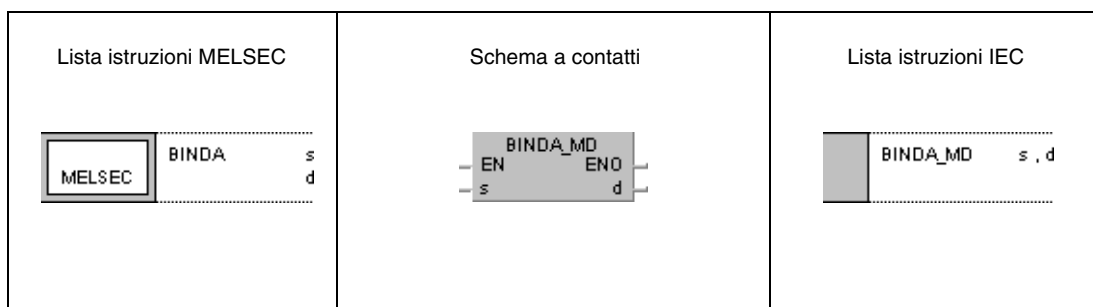
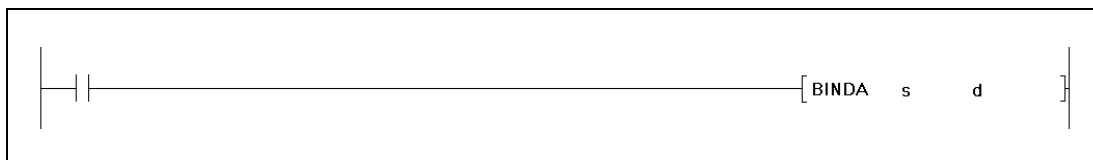
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi
MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	●	—	—	3	
d	—	●	●	—	—	—	—	—	—		

GX IEC
DeveloperGX
Developer

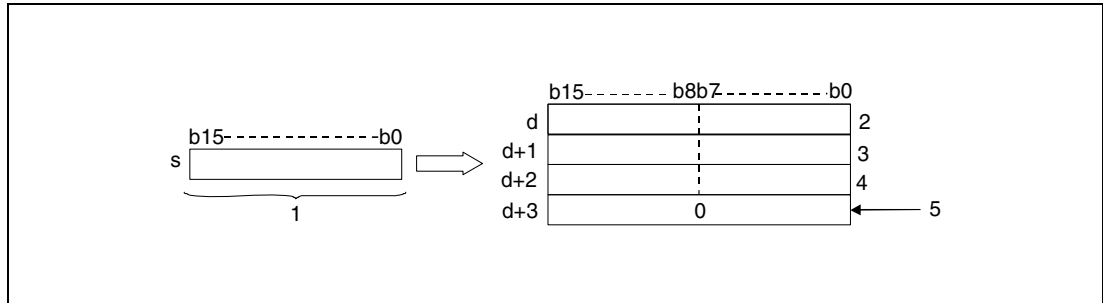
Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s	Dati binari da convertire in formato ASCII.	BIN 16/32-bit	ANY16/32
d	Indirizzo del primo operando che memorizza il risultato della conversione.	Stringa caratteri	Array [1..4]/ [1..6] di ANY16

Funzioni Conversione dati binari a 16-/32-bit in valori decimali in codice ASCII

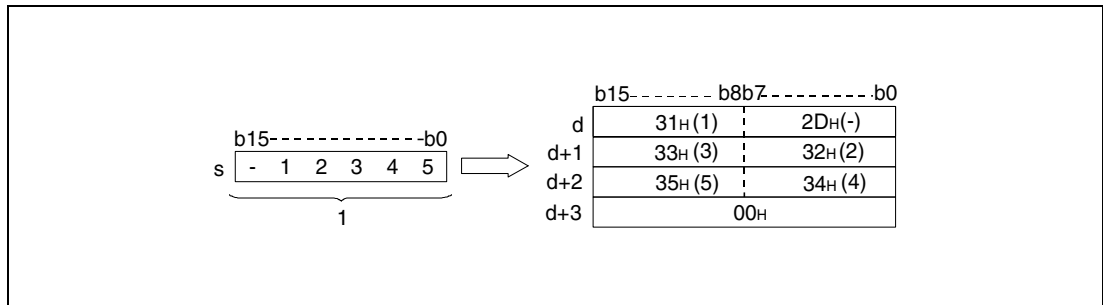
BINDA Conversione dati binari a 16-bit

L'istruzione BINDA converte un valore binario 16 bit specificato da s in un valore decimale in codice ASCII e lo memorizza nell'operando indicato da d (Array_d[1]) a d+3 (Array_d[4]).



- ¹ Dato binario a 16-bit
- ² Cifra delle decine di migliaia in codice ASCII/carattere segno
- ³ Cifra delle centinaia in codice ASCII/cifra delle migliaia in codice ASCII
- ⁴ Cifra delle unità in codice ASCII/cifra delle decine in codice ASCII
- ⁵ Con relé SM701 non attivo (0)

Il valore indicato da s viene memorizzato come valore decimale in codice ASCII a partire da d (Array_d[1]) fino a d+3 (Array_d[4]).



- ¹ Valore binario

Il valore binario a 16 bit può variare nel campo da -32768 a 32767.

I risultati delle operazioni di conversione vengono memorizzati in d come segue:

Se il valore binario a 16 bit è positivo, il carattere del segno viene memorizzato come "20H".
 Se il valore binario a 16 bit è negativo, il carattere del segno viene memorizzato come "2DH".

Il carattere del segno "20H" sostituisce gli zeri non significativi.
 Ad esempio, con il valore 00325 gli zeri delle cifre delle decine di migliaia e delle migliaia sono sostituiti da "20H", in modo che vengono memorizzate solo le cifre effettivamente utilizzate.

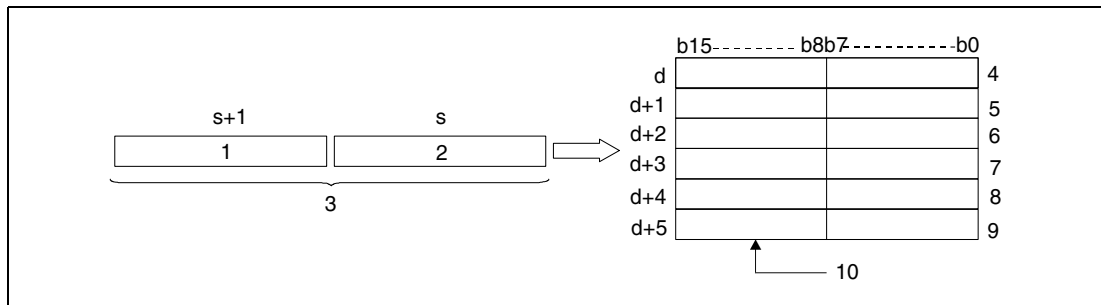
La memorizzazione dei dati nell'operando specificato da d+3 (Array_d[4]) dipende dallo stato del relé SM701.

Se il relé non è attivo, viene memorizzato uno zero "00H" nell'area d+3 (Array_d[4]).

Se il relé è attivo, il valore in d+3 (Array_d[4]) rimane invariato.

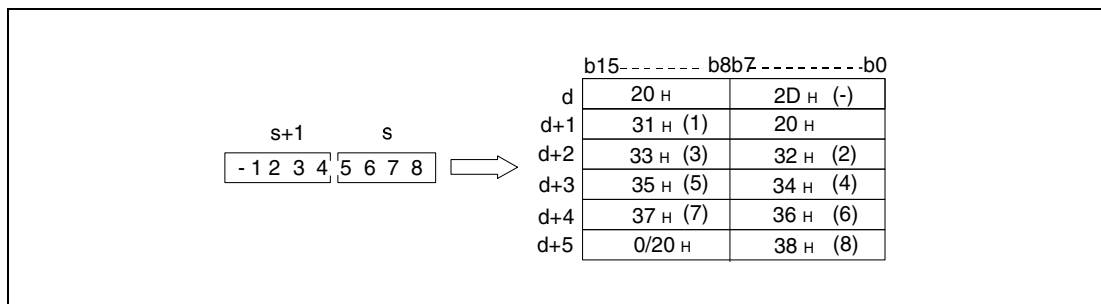
DBINDA Conversione di dati binari a 32-bit

L'istruzione DBINDA converte un valore binario 32 bit specificato da s e s+1 in un valore decimale in codice ASCII e lo memorizza negli operandi indicati da d (Array_d[1]) a d+5 (Array_d[6]).



- ¹ 16 bit più significativi
- ² 16 bit meno significativi
- ³ Dato binario a 32-bit
- ⁴ Carattere segno/cifra dei miliardi in codice ASCII
- ⁵ Cifra delle decine di milioni/cifra delle centinaia di milioni in codice ASCII
- ⁶ Cifra delle centinaia di migliaia/cifra dei milioni in codice ASCII
- ⁷ Cifra delle migliaia/cifra delle decine di migliaia in codice ASCII
- ⁸ Cifra delle decine/cifra delle centinaia in codice ASCII
- ⁹ 0 o 20H/ cifra delle unità in codice ASCII
- ¹⁰ Con relé SM701 non attivo (0)/ con relé SM701 attivo (20H)

Il valore indicato da s e s+1 viene memorizzato come valore decimale in codice ASCII a partire da d (Array_d[1]) fino a d+5 (Array_d[6]).



Il valore binario a 32 bit specificato da s può variare nel campo da -2147483648 a 2147483647.

I risultati delle operazioni di conversione sono memorizzati da d (Array_d[1]) a d+5 (Array_d[6]) come segue:

Se il valore binario è positivo, il carattere del segno viene memorizzato come "20H".
Se il valore binario è negativo, il carattere del segno viene memorizzato come "2DH".

Il carattere del segno "20H" sostituisce gli zeri non significativi.

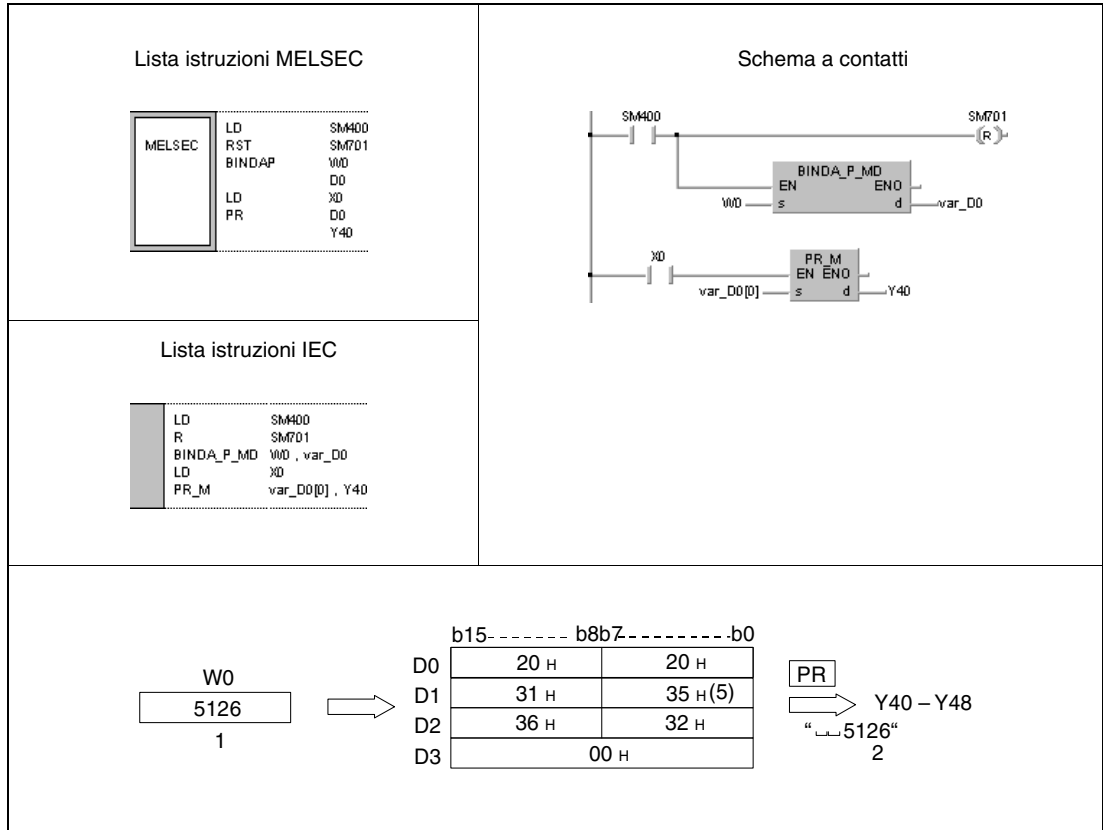
Ad esempio, con il valore 0012034560 gli zeri delle cifre dei miliardi e delle centinaia di milioni sono sostituiti da "20H", in modo che vengono memorizzate solo le cifre effettivamente utilizzate.

La memorizzazione dei dati negli 8 bit pesanti dell'operando specificato da d+5 (Array_d[6]) dipende dallo stato del relé SM701.
 Se il relé non è attivo, viene memorizzato uno zero "00H" nell'area d+5 (Array_d[6]).
 Se il relé non è attivo, viene memorizzato un carattere spazio (20H) nell'area d+5 (Array_d[6]).

Programma di esempio 1

BINDAP

Con il fronte positivo di SM400, il programma seguente invia il valore del dato binario a 16 bit in W0 come valore decimale in codice ASCII, tramite l'istruzione BINDAP. L'istruzione PR emette i caratteri sulle uscite da Y40 a Y48.

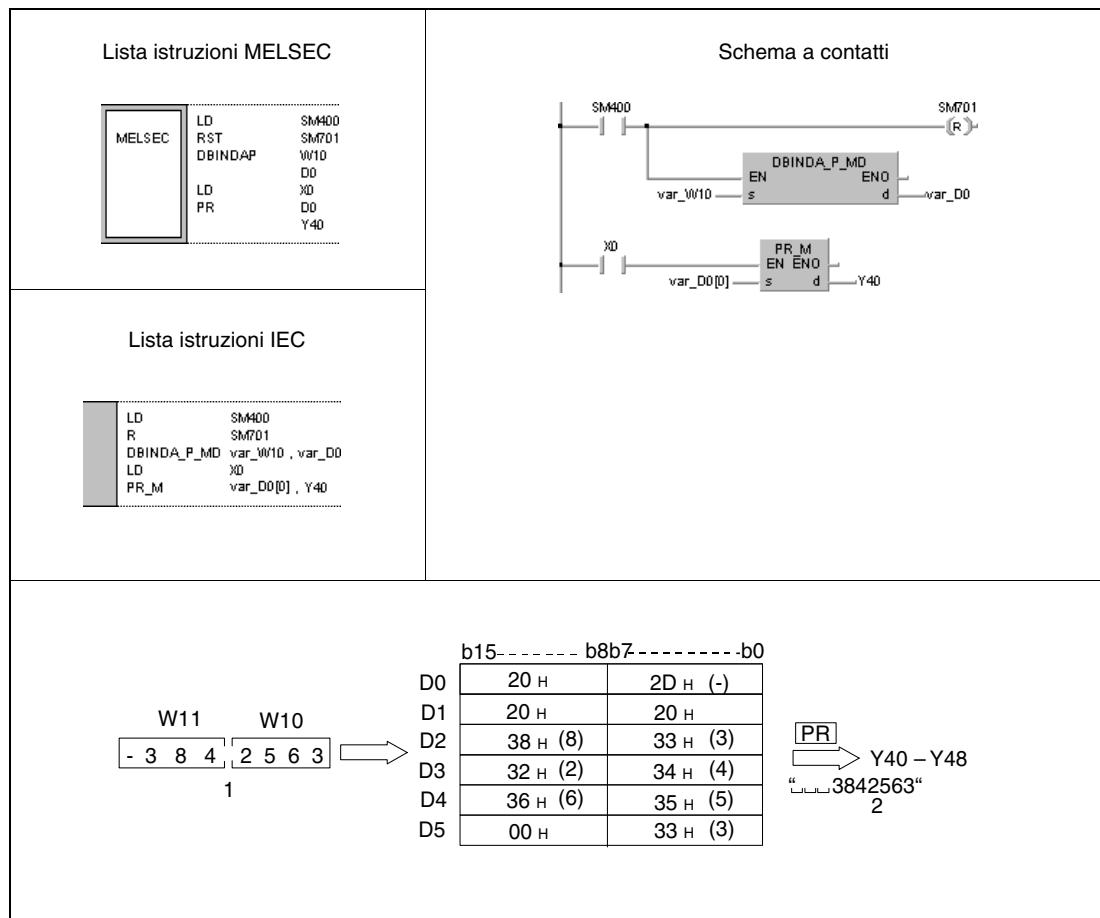


¹ Valore binario

² Uscita

Programma di esempio 2 DBINDAP

Con il fronte positivo di SM400, il programma seguente invia il valore del dato binario a 32 bit in W10 e W11 come valore decimale in codice ASCII, tramite l'istruzione DBINDAP. L'istruzione PR emette i caratteri sulle uscite da Y40 a Y48.



¹ Uscita

² Valore binario

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provocano messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.11.2 BINHA, BINHAP, DBINHA, DBINHAP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

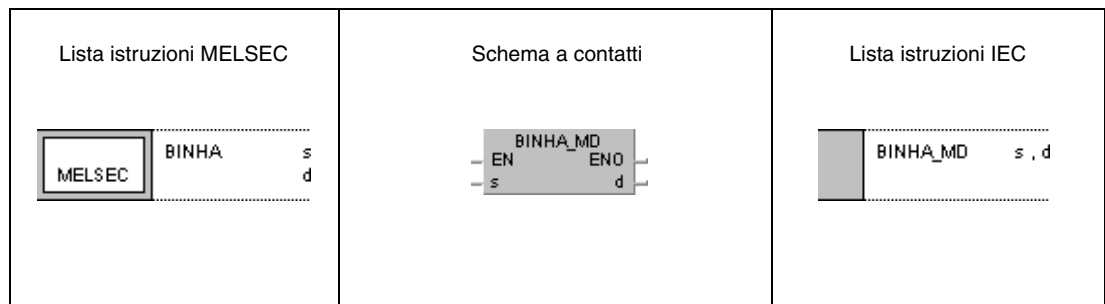
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

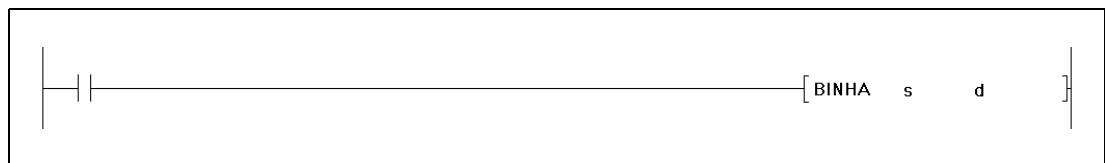
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	●	—	—	3	
d	—	●	●	—	—	—	—	—	—		

GX IEC Developer



GX Developer

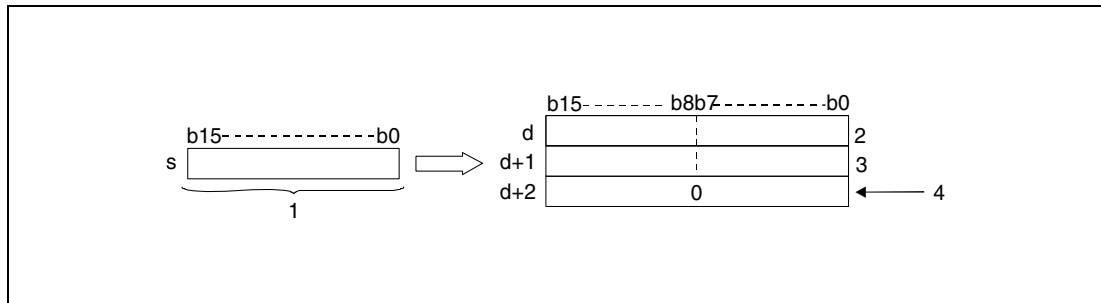


Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s	Dati binari da convertire in formato ASCII.	BIN 16/32-bit	ANY16/32
d	Indirizzo del primo operando che memorizza il risultato della conversione.	Stringa caratteri	Array [1.0.3]/ [1.0.5] di ANY16

Funzioni Conversione dati binari a 16-/32-bit in valori esadecimali in codice ASCII**BINHA Conversione dati binari a 16-bit**

L'istruzione BINHA converte un valore binario 16 bit specificato da s in un valore esadecimale in codice ASCII e lo memorizza negli operandi indicati da d (Array_d[1]) a d+2 (Array_d[3]).



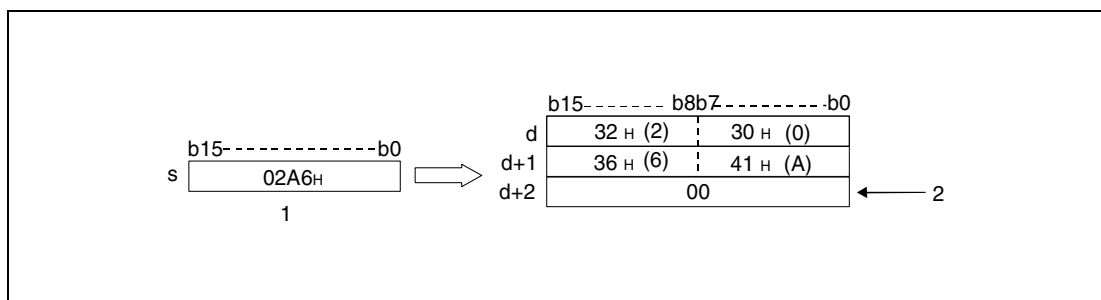
¹ Dato binario a 16-bit

² Codice ASCII della terza cifra/ codice ASCII della quarta cifra

³ Codice ASCII della prima cifra/ codice ASCII della seconda cifra

⁴ Con relé SM701 non attivo (0)

Il valore indicato da s viene memorizzato in codice ASCII a partire da d (Array_d[1]) fino a d+2 (Array_d[3]).



¹ Dato binario a 16-bit

² Con relé SM701 non attivo (0)

Il dato binario a 16-bit indicato da s può variare da 0H a FFFFH.

Il risultato della conversione viene memorizzato come valore esadecimale a 4 cifre da d (Array_d[1]) fino a d+2 (Array_d[3]).

Se una delle cifre è 0, viene considerata come valore 0 (gli zeri non vengono soppressi).

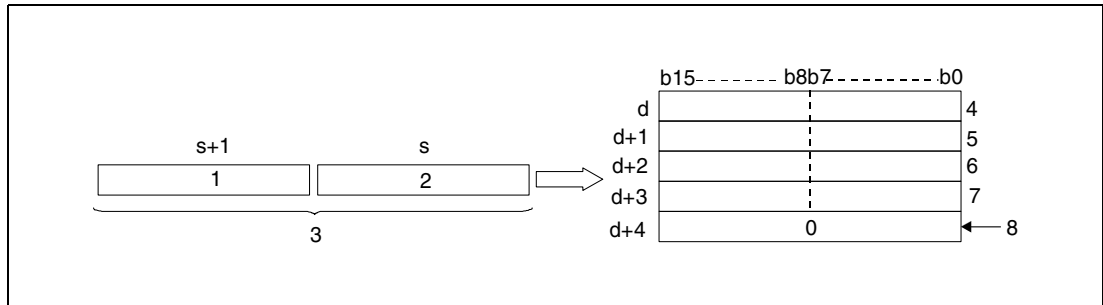
La memorizzazione dei dati nell'operando specificato da d+2 (Array_d[3]) dipende dallo stato del relé SM701 come segue:

Se il relé non è attivo, viene memorizzato uno zero "00H" nell'area d+2 (Array_d[3]).

Se il relé è attivo, il valore in d+2 (Array_d[3]) rimane invariato.

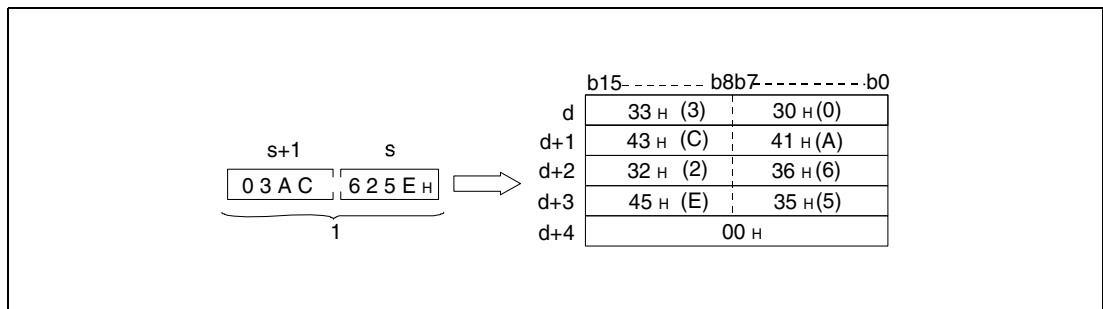
DBINHA Conversione dati binari a 32-bit

L'istruzione DBINHA converte un valore binario 32 bit specificato da s e s+1 in un valore esadecimale in codice ASCII e lo memorizza negli operandi indicati da d (Array_d[1]) a d+4 (Array_d[5]).



- ¹ 8 bit più significativi
- ² 8 bit meno significativi
- ³ Dato binario a 32-bit
- ⁴ Codice ASCII della settima cifra/ codice ASCII della ottava cifra
- ⁵ Codice ASCII della quinta cifra/ codice ASCII della sesta cifra
- ⁶ Codice ASCII della terza cifra/ codice ASCII della quarta cifra
- ⁷ Codice ASCII della prima cifra/ codice ASCII della seconda cifra
- ⁸ Con relé SM701 non attivo (0)

Il valore "03AC625EH" contenuto in s e s+1 viene memorizzato in d come segue:



¹ Confronto dati

Il dato binario a 32-bit indicato da s e s+1 può variare da 0H a FFFFFFFFH.

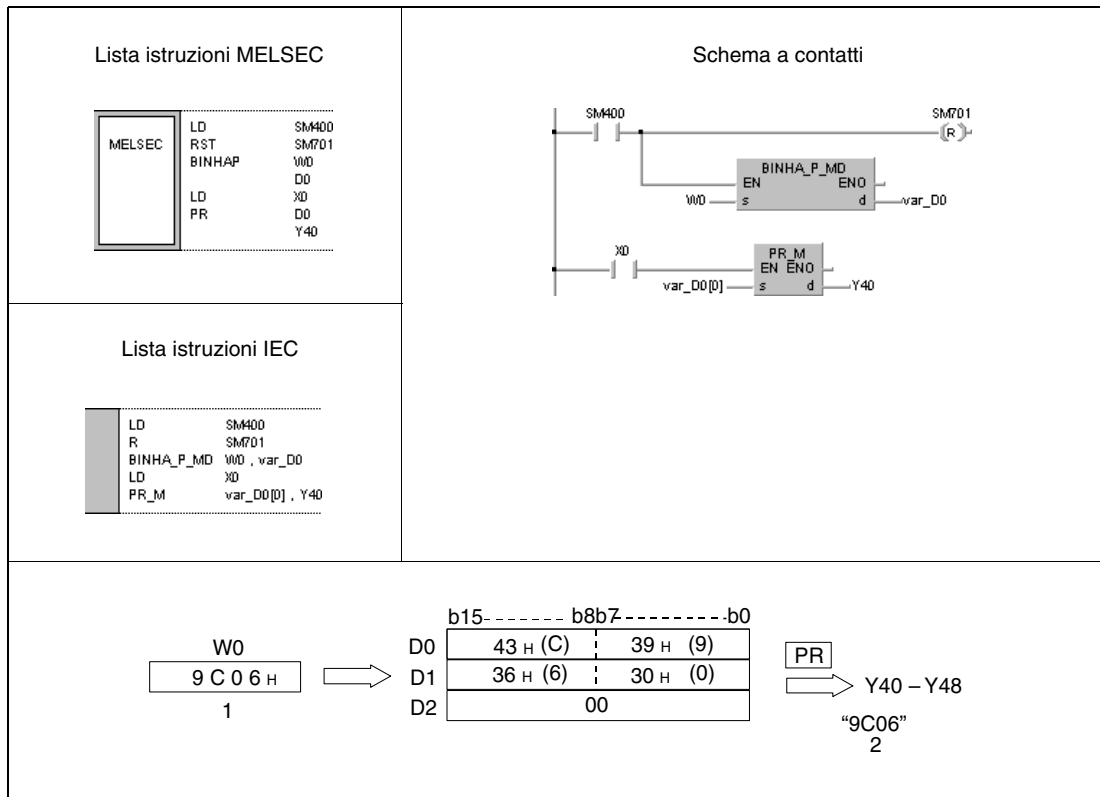
Il risultato della conversione viene memorizzato come valore esadecimale a 8 cifre da d (Array_d[1]) fino a d+4 (Array_d[5]).
 Se una delle cifre è 0, viene considerata come valore 0 (gli zeri non vengono soppressi).

La memorizzazione dei dati nell'operando specificato da d+4 (Array_d[5]) dipende dallo stato del relé SM701 come segue:

Se il relé non è attivo, viene memorizzato uno zero "00H" nell'area d+4 (Array_d[5]).
 Se il relé è attivo, il valore in d+4 (Array_d[5]) rimane invariato.

Programma di esempio 1 BINHAP

Con il fronte positivo di SM400, il programma seguente invia il valore del dato binario a 16 bit in W0 come valore decimale in codice ASCII, tramite l'istruzione BINHAP. L'istruzione PR emette i caratteri sulle uscite da Y40 a Y48.

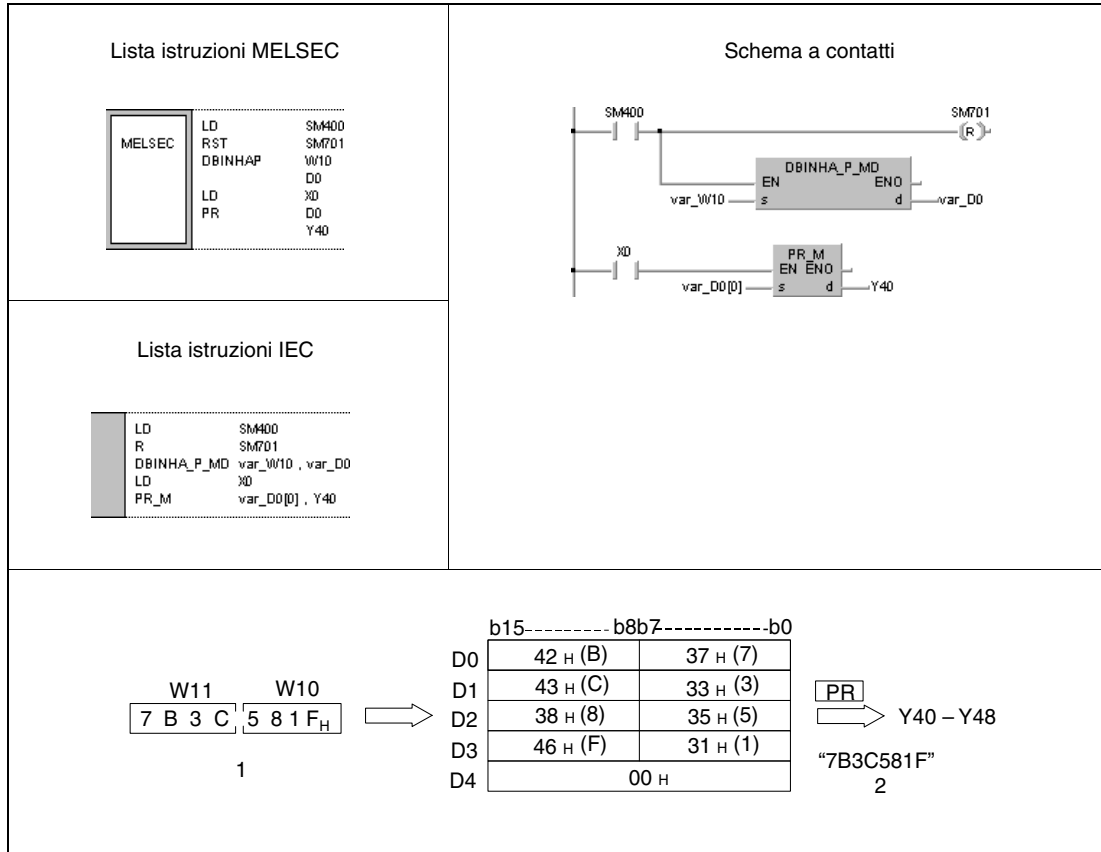


¹ Uscita

² Dato binario

Programma di esempio 2 DBINHAP

Con il fronte positivo di SM400, il programma seguente invia il valore del dato binario a 32 bit in W10 e W11 come valore decimale in codice ASCII, tramite l'istruzione DBINHAP. L'istruzione PR emette i caratteri sulle uscite da Y40 a Y48.



¹ Uscita

² Valore binario

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provocano messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.11.3 BCDDA, BCDDAP, DBCDDA, DBCDDAP**CPU**

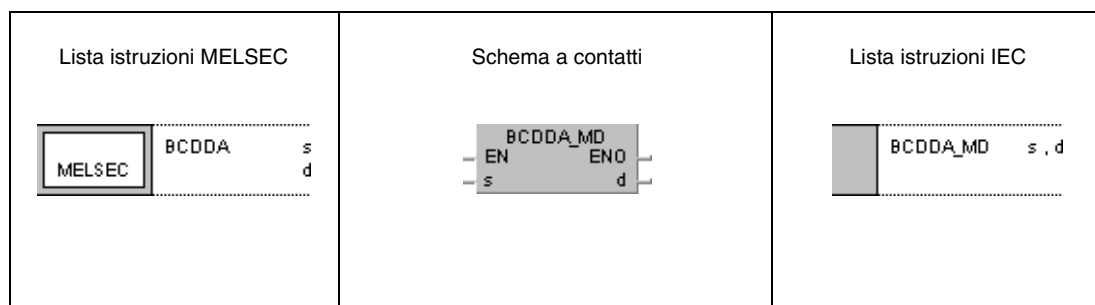
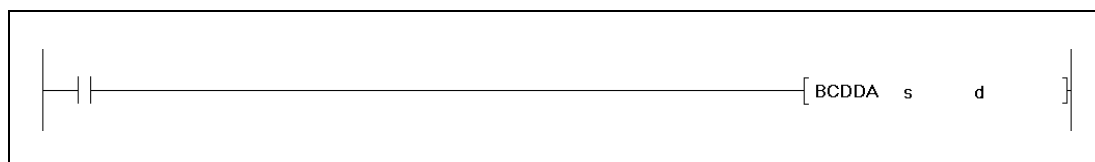
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti \$			Altro
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	●	—	SM0	3	
d	—	●	●	—	—	—	—	—			

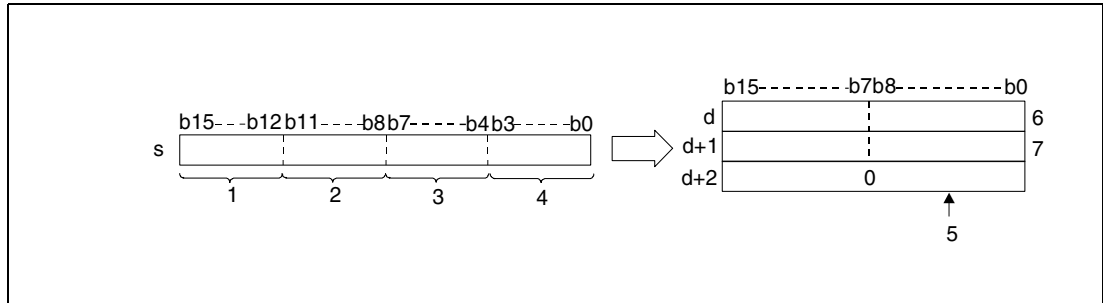
GX IEC Developer**GX Developer****Variabili**

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s	Dati BCD da convertire in formato ASCII.	Word	ANY16/32
d	Indirizzo del primo operando che memorizza il risultato della conversione.	Stringa caratteri	Array [1.0.3]/ [1.0.5] di ANY16

Funzioni Conversione di un dato BCD a 4/8 cifre in codice ASCII

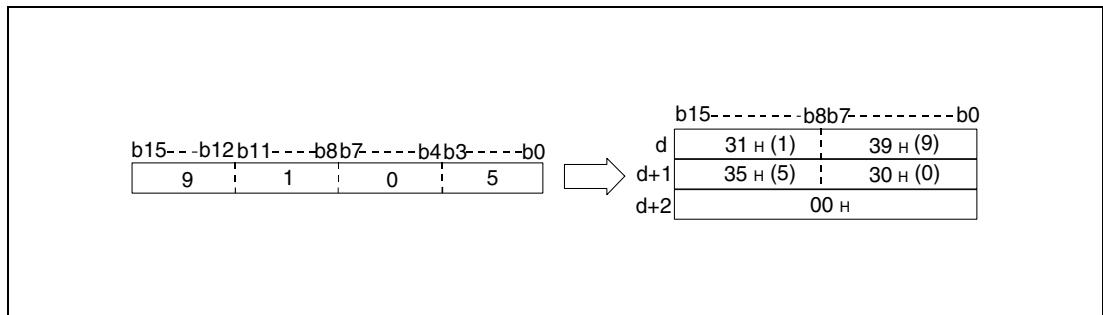
BCDDA Conversione di dato BCD a 4 cifre

L'istruzione BCDDA converte il dato BCD a 4 cifre specificato da s in formato ASCII e lo memorizza negli operandi indicati da d (Array_d[1]) a d+2 (Array_d[3]).



- ¹ Cifra delle migliaia
- ² Cifra delle centinaia
- ³ Cifra delle decine
- ⁴ Cifra delle unità
- ⁵ Con relé SM701 non attivo (0)
- ⁶ Codice ASCII della terza cifra/ codice ASCII della quarta cifra
- ⁷ Codice ASCII della prima cifra/ codice ASCII della seconda cifra

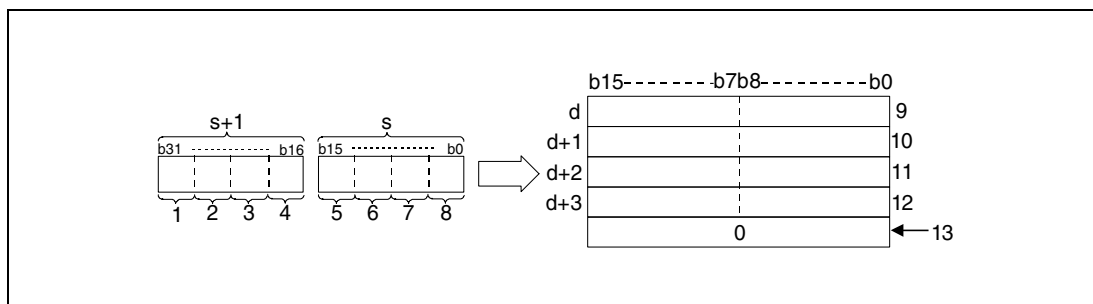
Il valore 9105 contenuto in s, viene memorizzato come segue:



Il valore BCD contenuto in s può variare da 0 a 9999.
 Il risultato della conversione viene memorizzato da d (Array_d[1]) fino a d+2 (Array_d[3]).
 Se una delle cifre è 0, viene considerata come valore "30H" (gli zeri non vengono soppressi).
 La memorizzazione dei dati nell'operando specificato da d+2 (Array_d[3]) dipende dallo stato del relé SM701 come segue:
 Se il relé non è attivo, viene memorizzato uno zero "00H" nell'area d+2 (Array_d[3]).
 Se il relé è attivo, il valore in d+2 (Array_d[3]) rimane invariato.

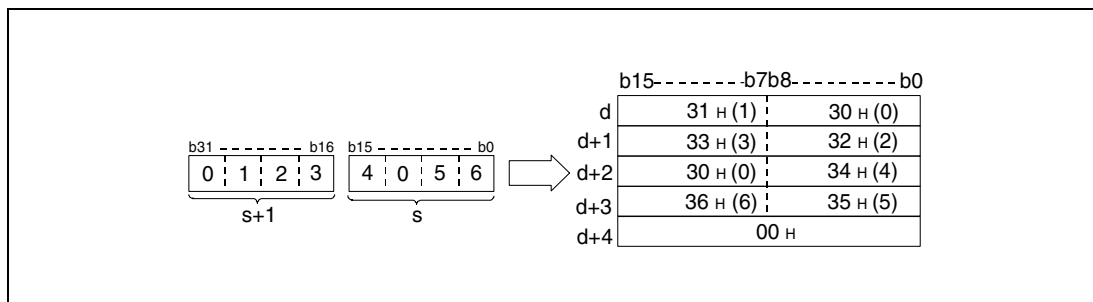
DBCDDA Conversione di dato BCD a 8 cifre

L'istruzione DBCDDA converte il dato BCD a 8 cifre specificato da s e s+1 in formato ASCII e lo memorizza negli operandi indicati da d (Array_d[1]) a d+4 (Array_d[5]).



- ¹ Cifra delle decine di milioni
- ² Cifra dei milioni
- ³ Cifra delle centinaia di migliaia
- ⁴ Cifra delle decine di migliaia
- ⁵ Cifra delle migliaia
- ⁶ Cifra delle centinaia
- ⁷ Cifra delle decine
- ⁸ Cifra delle unità
- ⁹ Codice ASCII della settima cifra/ codice ASCII della ottava cifra
- ¹⁰ Codice ASCII della quinta cifra/ codice ASCII della sesta cifra
- ¹¹ Codice ASCII della terza cifra/ codice ASCII della quarta cifra
- ¹² Codice ASCII della prima cifra/ codice ASCII della seconda cifra
- ¹³ Con relé SM701 non attivo (0)

Il valore 01234056 contenuto in s e s+1 viene memorizzato in d come segue:



Il valore BCD contenuto in s e s+1 può variare da 0 a 99999999.

Il risultato della conversione viene memorizzato da d (Array_d[1]) fino a d+4 (Array_d[5]).

Se una delle cifre è 0, viene considerata come valore "30H" (gli zeri non vengono soppressi).

La memorizzazione dei dati nell'operando specificato da d+4 (Array_d[5]) dipende dallo stato del relé SM701.

Se il relé non è attivo, viene memorizzato uno zero "00H" nell'area d+4 (Array_d[5]).

Se il relé è attivo, il valore in d+4 (Array_d[5]) rimane invariato.

Errori di esecuzione

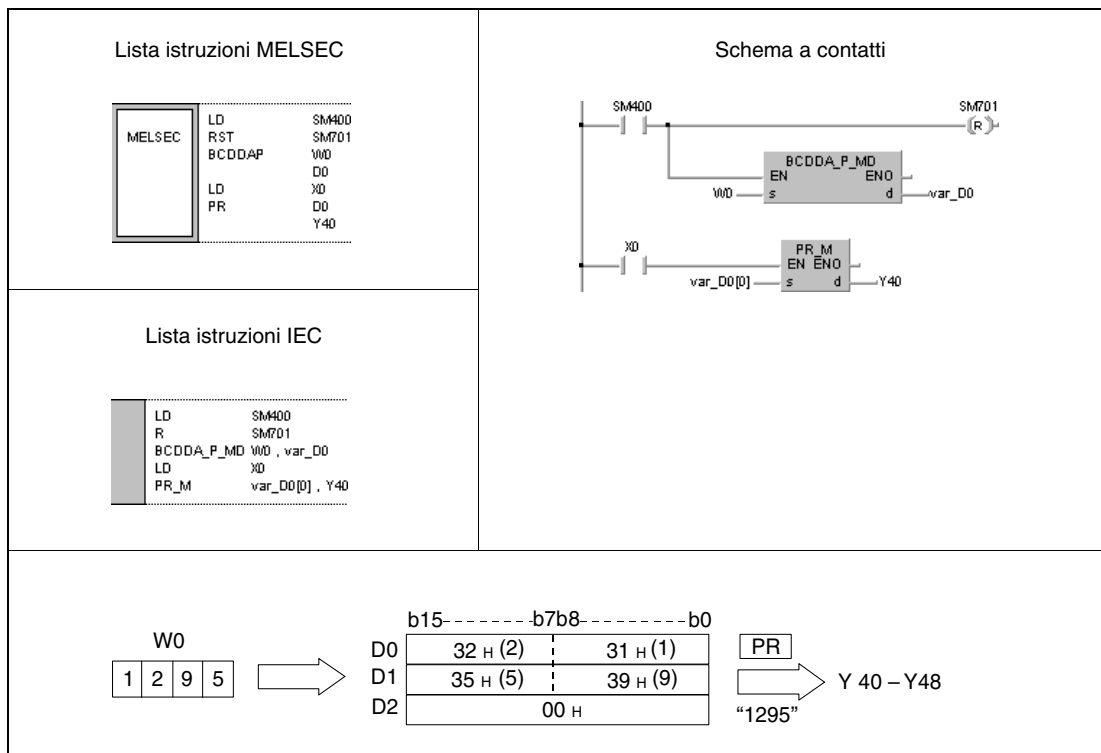
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il dato BCD in s supera il campo da 0 a 9999 durante l'esecuzione di una istruzione BCDDA (codice di errore: 4100).
- Il dato BCD in s supera il campo da 0 a 99999999 durante l'esecuzione di una istruzione DBCDDA (codice di errore: 4100).

Programma di esempio 1

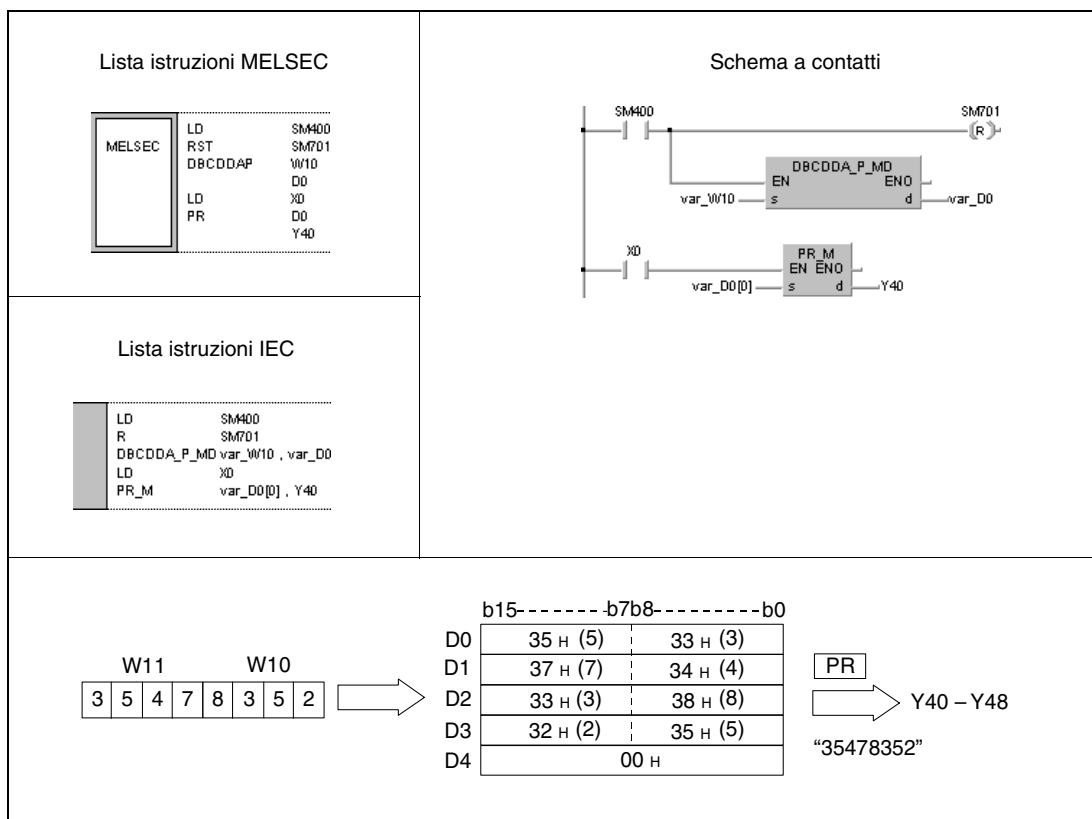
BCDDAP

Con il fronte positivo di SM400, il programma seguente invia il valore del dato BCD a 4 cifre in W0 come valore decimale in codice ASCII, tramite l'istruzione BCDDAP. L'istruzione PR emette i caratteri sulle uscite da Y40 a Y48.



Programma di esempio 2 DBCDDAP

Con il fronte positivo di SM400, il programma seguente invia il valore del dato binario a 8 bit in W10 e W11 come valore decimale in codice ASCII, tramite l'istruzione PR. L'istruzione PR emette i caratteri sulle uscite da Y40 a Y48.

**NOTA**

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provocano messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.11.4 DABIN, DABINP, DDABIN, DDABINP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

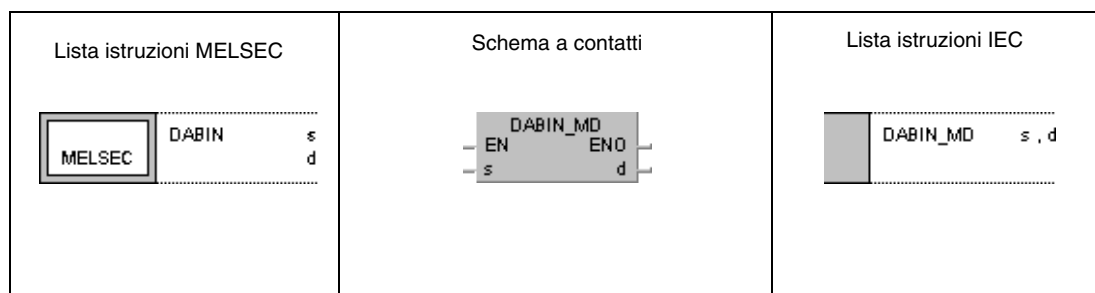
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

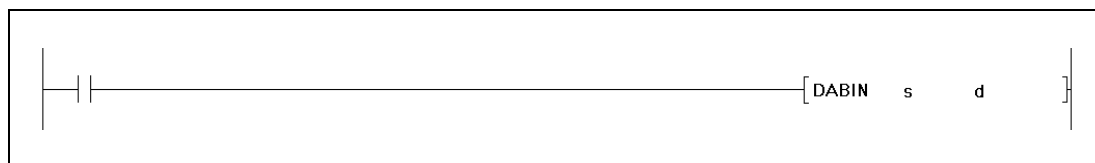
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti \$			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	●	—	SM0	3
d	●	●	●	●	●	●	●	—	—		

GX IEC Developer



GX Developer

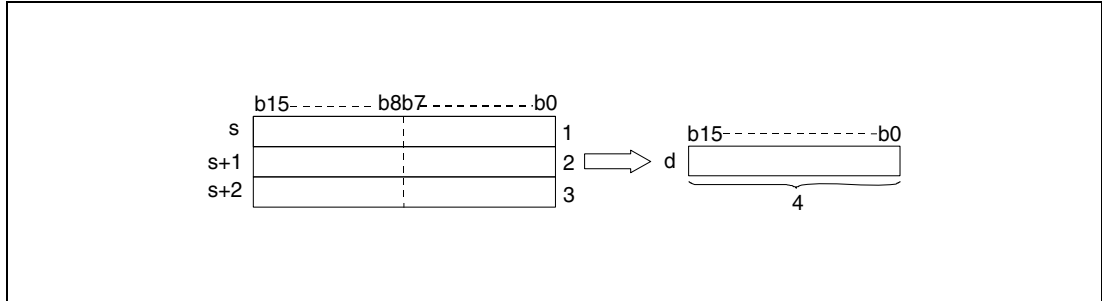


Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s	Area di memoria che contiene il dato ASCII da convertire.	Stringa caratteri	Array [1.0.3]/ [1..6] di ANY16
d	Area di memoria che contiene il risultato della conversione.	BIN 16/32-bit	ANY16/32

Funzioni Conversione di dati decimali ASCII a dati BIN a 16/32 bit**DABIN Conversione di dati binari a 16-bit**

L'istruzione DABIN converte il dato ASCII decimale contenuto nell'area da s (Array_s[1]) a s+2 (Array_s[3]) in formato BIN a 16 bit e lo memorizza negli operandi indicati da d.



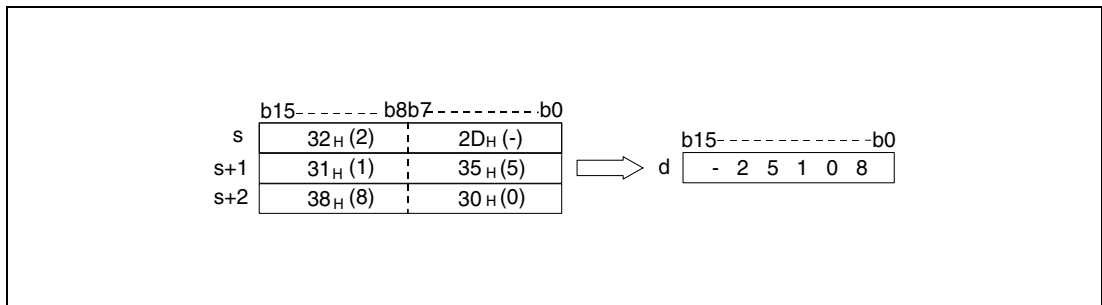
¹ Codice ASCII della cifra delle decine di migliaia/ carattere del segno

² Codice ASCII della cifra delle centinaia/ codice ASCII della cifra delle migliaia

³ Codice ASCII della cifra delle unità/ codice ASCII della cifra delle decine

⁴ Dato binario BIN a 16 bit

Il valore contenuto nell'area da s (Array_s[1]) a s+2 (Array_s[3]) viene memorizzato in d come -25018H come segue:



Il valore ASCII contenuto da s (Array_s[1]) a s+2 (Array_s[3]) può variare nel campo da -32768 a 32767.

Se il valore binario è positivo, il carattere del segno viene memorizzato come "20H".

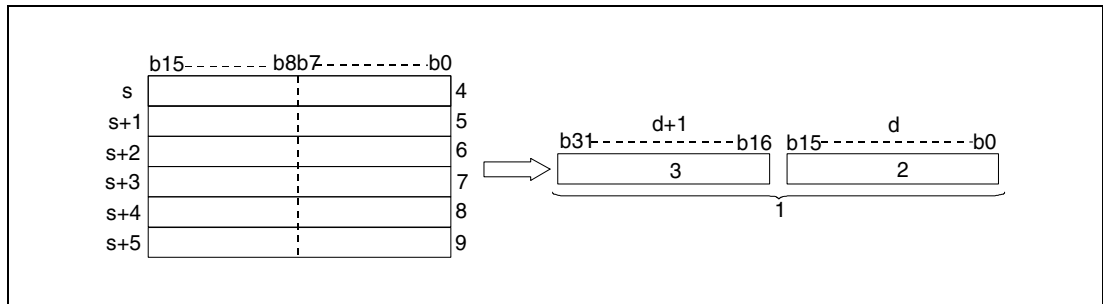
Se il risultato è negativo, viene memorizzato il valore "2DH".

Ciascuna cifra memorizzata del codice ASCII può variare da "30H" a "39H".

Se una cifra contiene il valore "20H" o "00H", il suo contenuto viene automaticamente sovrascritto con il valore "30H".

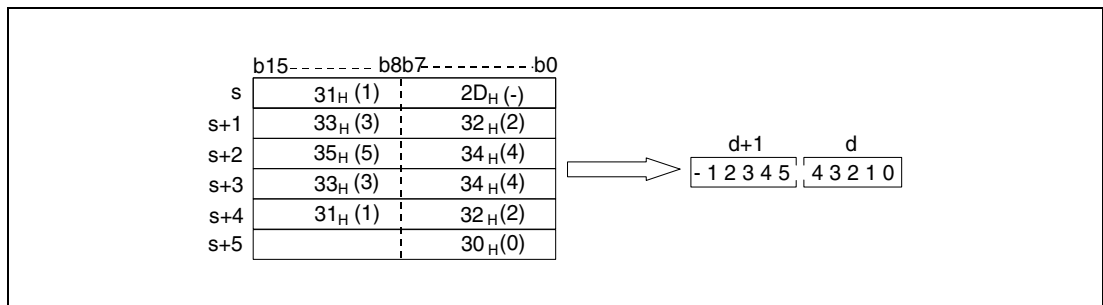
DDABIN Conversione in dati BIN a 32-bit

L'istruzione DDABIN converte il dato ASCII decimale contenuto nell'area da s (Array_s[1]) a s+5 (Array_s[6]) in formato BIN a 32 bit e lo memorizza negli operandi indicati da d e d+1.



- ¹ Dato binario BIN a 32 bit
- ² 16 bit meno significativi
- ³ 16 bit più significativi
- ⁴ Codice ASCII della cifra dei miliardi/ carattere del segno
- ⁵ Codice ASCII della cifra delle decine di milioni/ codice ASCII della cifra delle centinaia di milioni
- ⁶ Codice ASCII della cifra delle centinaia di migliaia/ codice ASCII della cifra dei milioni
- ⁷ Codice ASCII della cifra delle migliaia/ codice ASCII della cifra delle decine di migliaia
- ⁸ Codice ASCII della cifra delle decine/ codice ASCII della cifra delle centinaia
- ⁹ Ignorato/ codice ASCII della cifra delle decine

Il valore contenuto nell'area da s (Array_s[1]) a s+5 (Array_s[6]) viene memorizzato in d come -1234543210H come segue:



Il valore ASCII contenuto da s (Array_s[1]) a s+5 (Array_s[6]) può variare nel campo da -2147483648 a 2147483647.

Se il valore binario è positivo, il carattere del segno viene memorizzato come "20H".

Se il risultato è negativo, viene memorizzato il valore "2DH".

Ciascuna cifra memorizzata del codice ASCII può variare da "30H" a "39H".

Se una cifra contiene il valore "20H" o "00H", il suo contenuto viene automaticamente sovrascritto con il valore "30H".

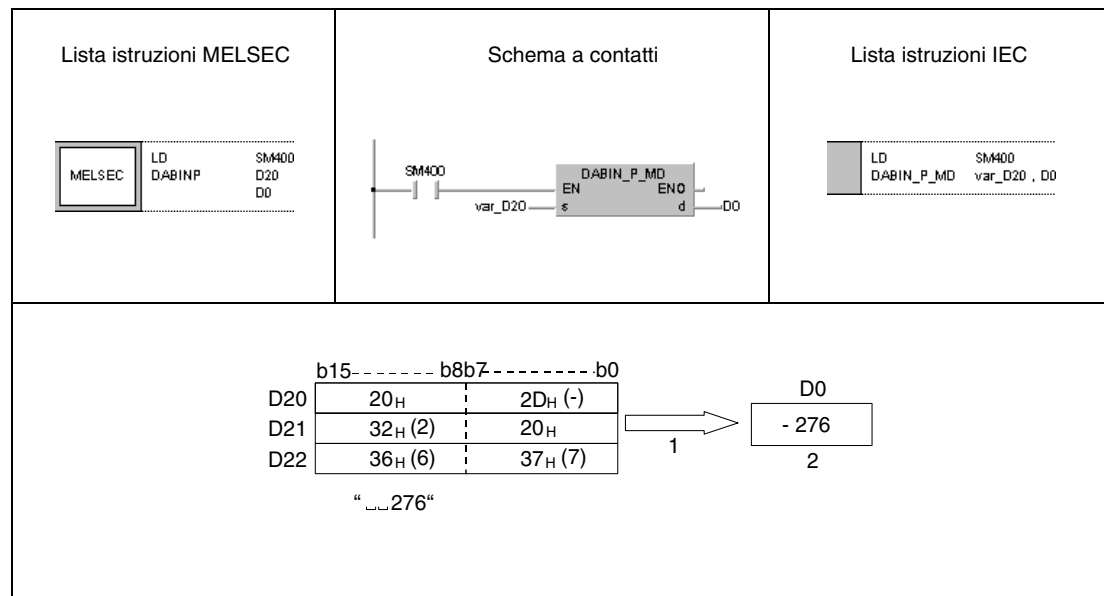
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il carattere del segno contenuto nei 16 bit più leggeri dell'operando s (Array_s[1]) contiene un valore diverso da "30H" a "39H, "20H" o "00H" (codice di errore 4100).
- Il codice ASCII contenuto nell'area da s (Array_s[1]) a s+5 (Array_s[6]) contiene valori diversi da "30H" a "39H, "20H" o "00H" (codice di errore 4100).
- Il codice ASCII contenuto nell'area da s (Array_s[1]) a s+5 (Array_s[6]) supera i campi di valori seguenti:
 Per l'istruzione DABIN da -32768 a 32767
 Per l'istruzione DDABIN da -2147483648 a 2147483647 (codice di errore 4100).

Programma di esempio 1**DABINP**

Con il fronte positivo di SM400, il programma seguente converte il valore ASCII decimale a cinque cifre da D20 (var_D20 Array [0]) a D22 (var_D20 Array [2]) in valore binario e lo memorizza in D0.

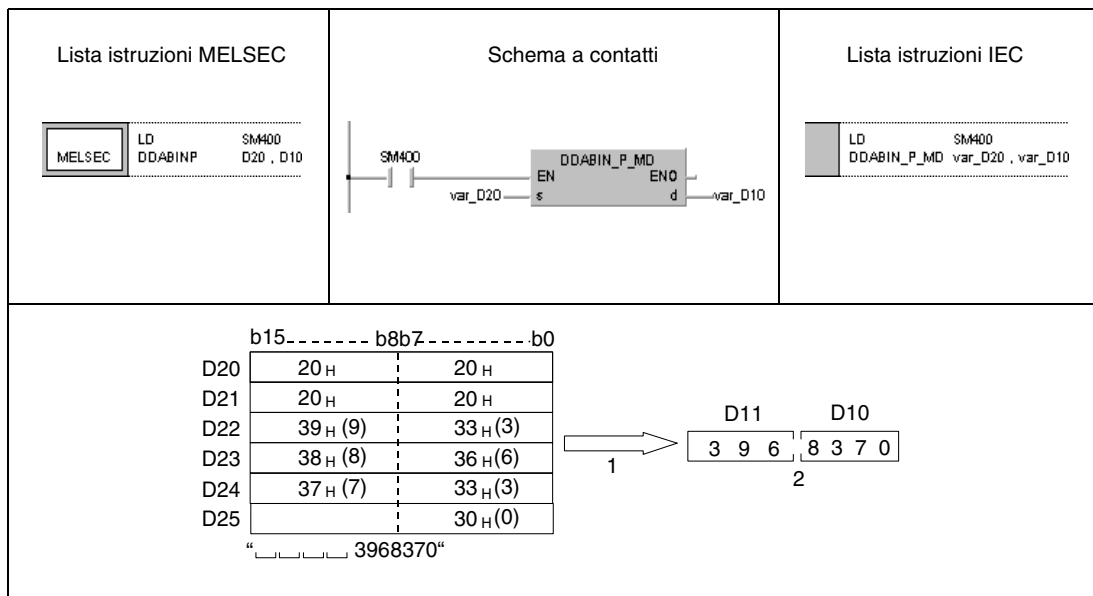


¹ Viene letto come -00276

² Valore binario

Programma di esempio 2 DDABINP

Con il fronte positivo di SM400, il programma seguente converte il valore ASCII decimale a dieci cifre da D20 (var_D20 Array [0]) a D25 (var_D20 Array [5]) in valore binario e lo memorizza in D10 e D11.



- ¹ Viene letto come +0003968370
- ² Valore binario

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provocano messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.11.5 HABIN, HABINP, DHABIN, DHABINP**CPU**

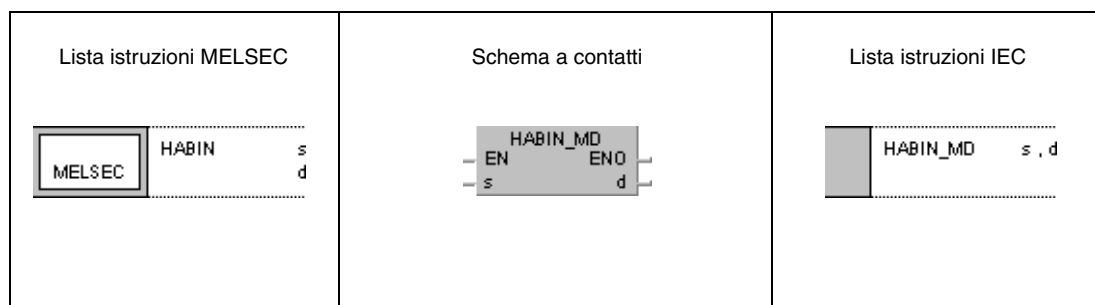
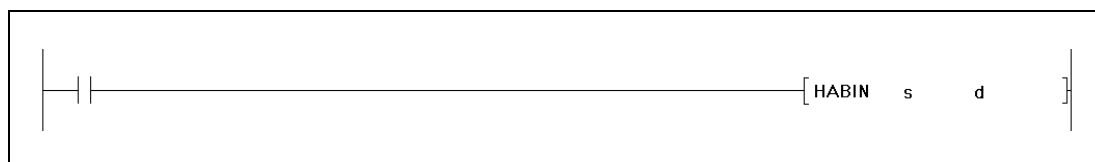
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti \$			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	●	—	SM0	3
d	●	●	●	●	●	●	●	—	—		

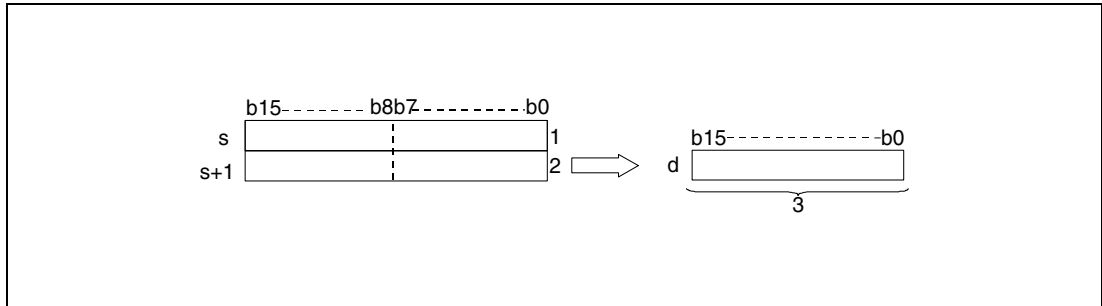
GX IEC Developer**GX Developer****Variabili**

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s	Area di memoria che contiene il dato ASCII da convertire.	Stringa caratteri	ANY32/Array [1..4] di ANY16
d	Area di memoria che contiene il risultato della conversione.	BIN 16/32-bit	ANY16/32

Funzioni Conversione di dati esadecimali ASCII a dati BIN a 16/32 bit

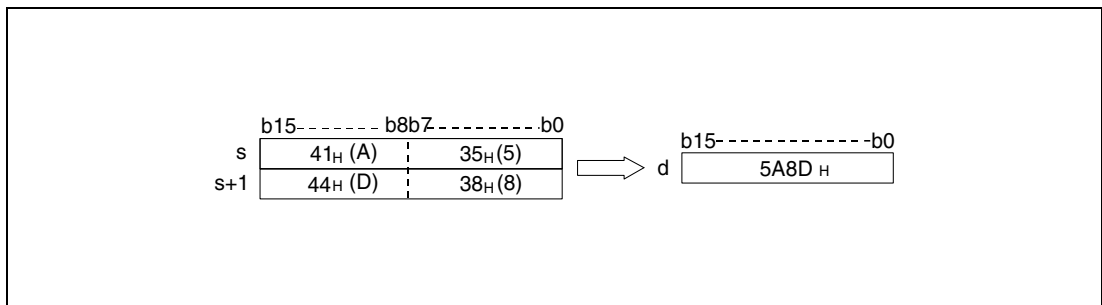
HABIN Conversione in dati BIN a 16-bit

L'istruzione HABIN converte il dato ASCII esadecimale specificato da s e s+1 in formato BIN a 16 bit e lo memorizza negli operandi indicati da d.



- ¹ Codice ASCII della terza cifra/ codice ASCII della quarta cifra
- ² Codice ASCII della prima cifra/ codice ASCII della seconda cifra
- ³ Dato binario BIN a 16 bit

Il valore "5A8D_H" contenuto in s e s+1 viene memorizzato come segue dopo l'esecuzione:

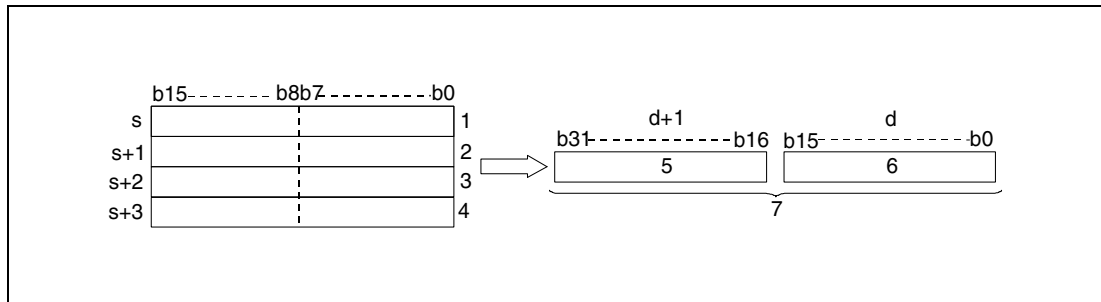


Il valore ASCII contenuto da s a s+1 può variare da 0000_H a FFFF_H.

Ciascuna cifra memorizzata del codice ASCII può variare da "30_H" a "39_H" e da "41_H" a "46_H".

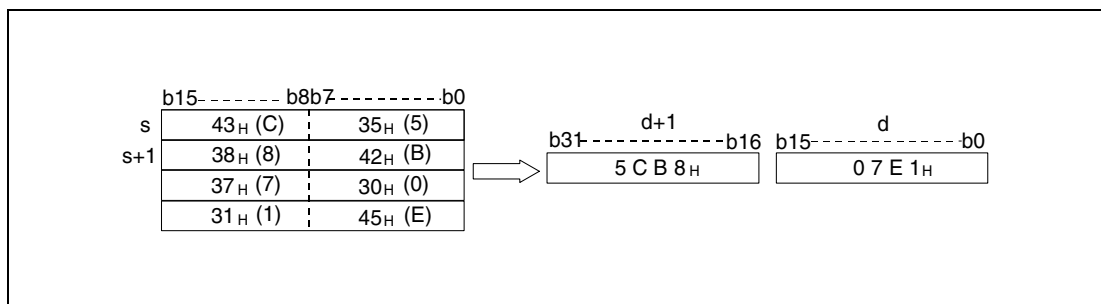
DHABIN Conversione in dati BIN a 32-bit

L'istruzione DHABIN converte il dato ASCII esadecimale contenuto nell'area da s (Array_s[1]) a s+3 (Array_s[4]) in formato BIN a 32 bit e lo memorizza negli operandi indicati da d e d+1.



- ¹ Codice ASCII della settima cifra/ codice ASCII della ottava cifra
- ² Codice ASCII della quinta cifra/ codice ASCII della sesta cifra
- ³ Codice ASCII della terza cifra/ codice ASCII della quarta cifra
- ⁴ Codice ASCII della prima cifra/ codice ASCII della seconda cifra
- ⁵ 16 bit più significativi
- ⁶ 16 bit meno significativi
- ⁷ Dato binario BIN a 32 bit

Il valore "5CB807E1" contenuto da s (Array_s[1]) a s+3 (Array_s[4]) viene memorizzato dopo l'esecuzione in d e d+1 come segue:



Il valore ASCII contenuto da s (Array_s[1]) a s+3 (Array_s[4]) può variare nel campo da 00000000H a FFFFFFFFH.

Ciascuna cifra memorizzata del codice ASCII può variare da "30H" a "39H" e da "41H" a "46H".

Errori di esecuzione

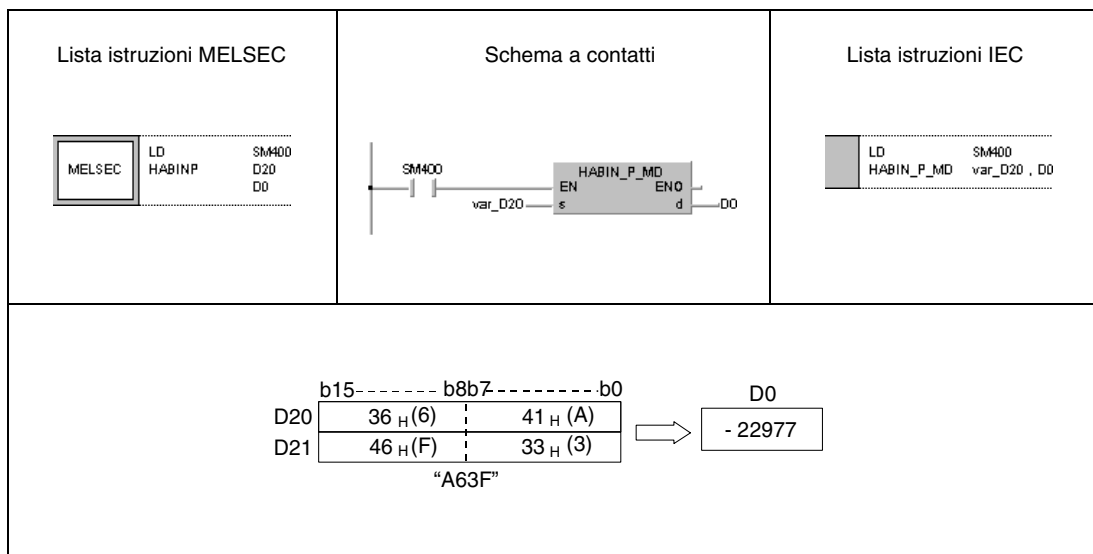
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il codice ASCII contenuto nell'area da s (Array_s[1]) a s+3 (Array_s[4]) non è nel campo da "30H" a "39H" e da "41H" a "46H" (codice di errore 4100).

Programma di esempio 1

HABINP

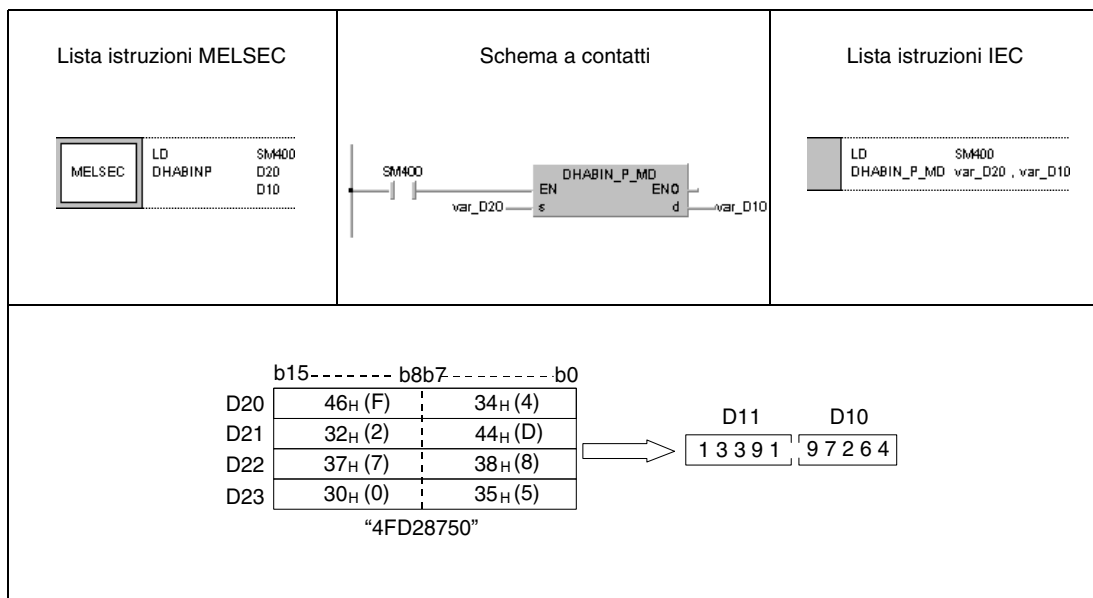
Con il fronte positivo di SM400, il programma seguente converte il valore ASCII a quattro cifre da D20 (var_D20 Array [0]) a D21 (var_D20 Array [1]) in valore binario e lo memorizza in D0.



Programma di esempio 2

DHABINP

Con il fronte positivo di SM400, il programma seguente converte il valore ASCII a otto cifre da D20 (var_D20 Array [0]) a D23 (var_D20 Array [3]) in valore binario e lo memorizza in D10 e D11.



NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provocano messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.11.6 DABCD, DABCDP, DDABCD, DDABCDP**CPU**

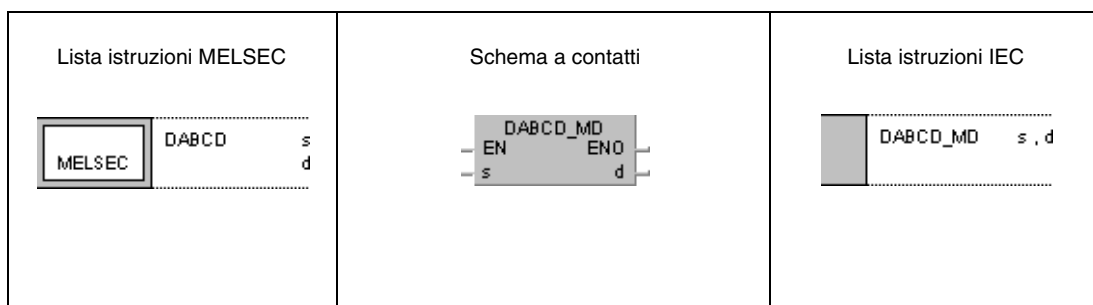
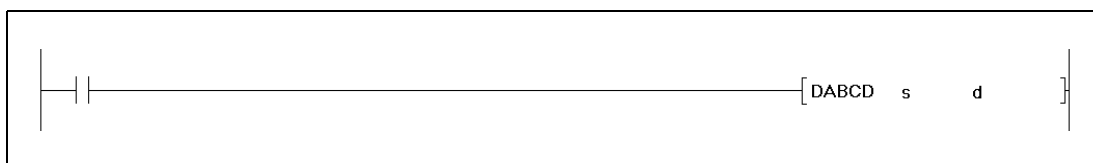
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti \$			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	●	—	SM0	3
d	●	●	●	●	●	●	●	—	—		

GX IEC Developer**GX Developer****Variabili**

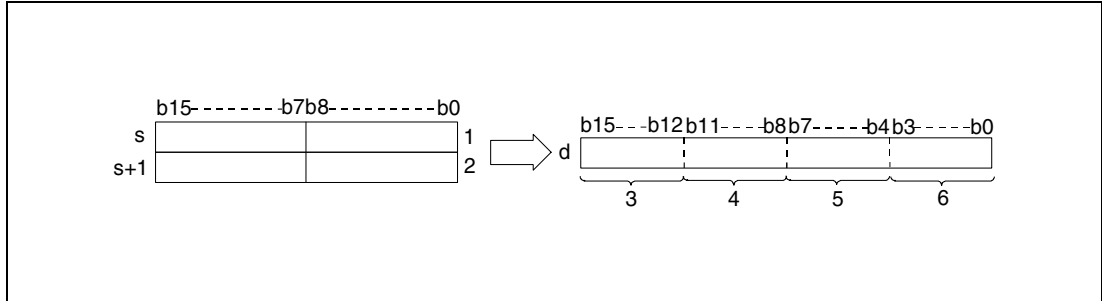
Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s	Area di memoria che contiene il dato ASCII da convertire.	Stringa caratteri	ANY32/Array [1..4] di ANY16
d	Area di memoria che contiene il risultato della conversione.	Dati BCD a 4/8 cifre	ANY16/32

Funzioni

Conversione di un dato ASCII decimale in codice BCD a 4/8 cifre

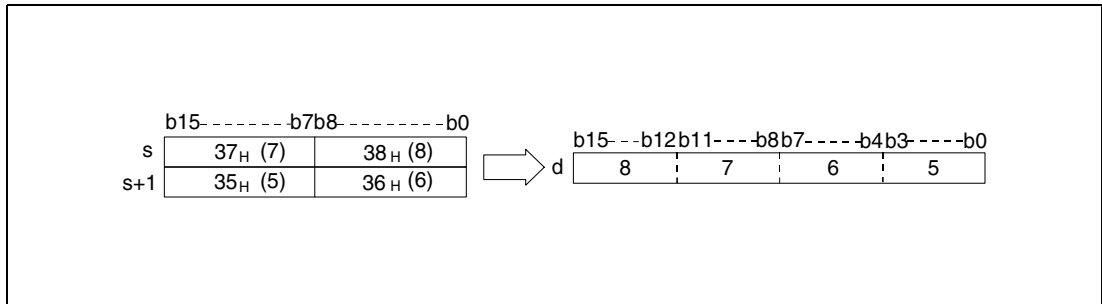
DABCD Conversione in dato BCD a 4 cifre

L'istruzione DABCD converte il dato ASCII decimale specificato da s e s+1 in formato BCD a 4 cifre e lo memorizza negli operandi indicati da d.



- ¹ Codice ASCII della cifra delle centinaia/ codice ASCII della cifra delle migliaia
- ² Codice ASCII della cifra delle unità/ codice ASCII della cifra delle decine
- ³ Cifra delle migliaia
- ⁴ Cifra delle centinaia
- ⁵ Cifra delle decine
- ⁶ Cifra delle unità

Il valore 8765 contenuto in s e s+1 viene memorizzato in d come segue:



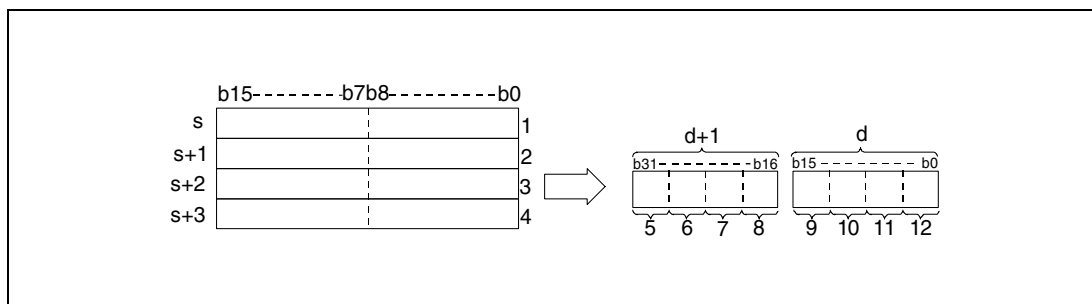
Il valore ASCII contenuto in s e s+1 può variare da 0 a 9999.

Ciascuna cifra memorizzata del codice ASCII può variare da "30H" a "39H".

Se una cifra contiene il valore "20H" o "00H", il suo contenuto viene automaticamente sovrascritto con il valore "30H".

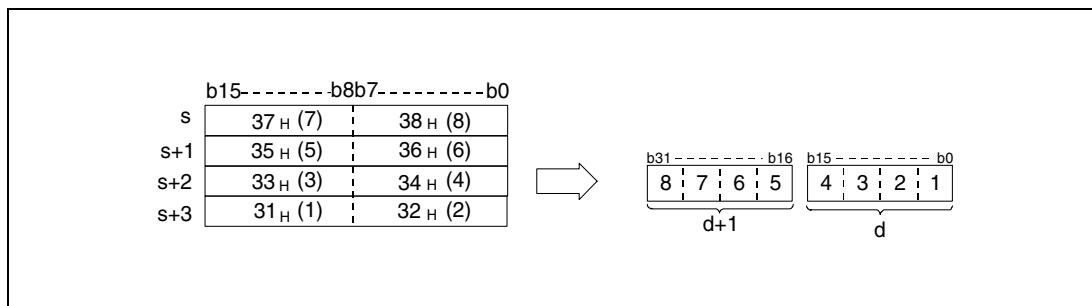
DDABCD Conversione in dato BCD a 8 cifre

L'istruzione DDABCD converte il dato ASCII contenuto nell'area da s (Array_s[1]) a s+3 (Array_s[4]) in formato BCD a 8 cifre e lo memorizza negli operandi indicati da d e d+1.



- ¹ Codice ASCII della cifra dei milioni/ codice ASCII della cifra delle decine di milioni
- ² Codice ASCII della cifra delle decine di migliaia/ codice ASCII della cifra delle centinaia di migliaia
- ³ Codice ASCII della cifra delle centinaia/ codice ASCII della cifra delle migliaia
- ⁴ Codice ASCII della cifra delle unità/ codice ASCII della cifra delle decine
- ⁵ Cifra delle decine di milioni
- ⁶ Cifra dei milioni
- ⁷ Cifra delle centinaia di migliaia
- ⁸ Cifra delle decine di migliaia
- ⁹ Cifra delle migliaia
- ¹⁰ Cifra delle centinaia
- ¹¹ Cifra delle decine
- ¹² Cifra delle unità

Il valore 87654321 contenuto da s (Array_s[1]) a s+3 (Array_s[4]) viene memorizzato in d e d+1 come segue:



Il valore ASCII contenuto da s (Array_s[1]) a s+3 (Array_s[4]) può variare nel campo da 0 a 99999999.

Ciascuna cifra memorizzata del codice ASCII può variare da "30H" a "39H".

Se una cifra contiene il valore "20H" o "00H", il suo contenuto viene automaticamente sovrascritto con il valore "30H".

Errori di esecuzione

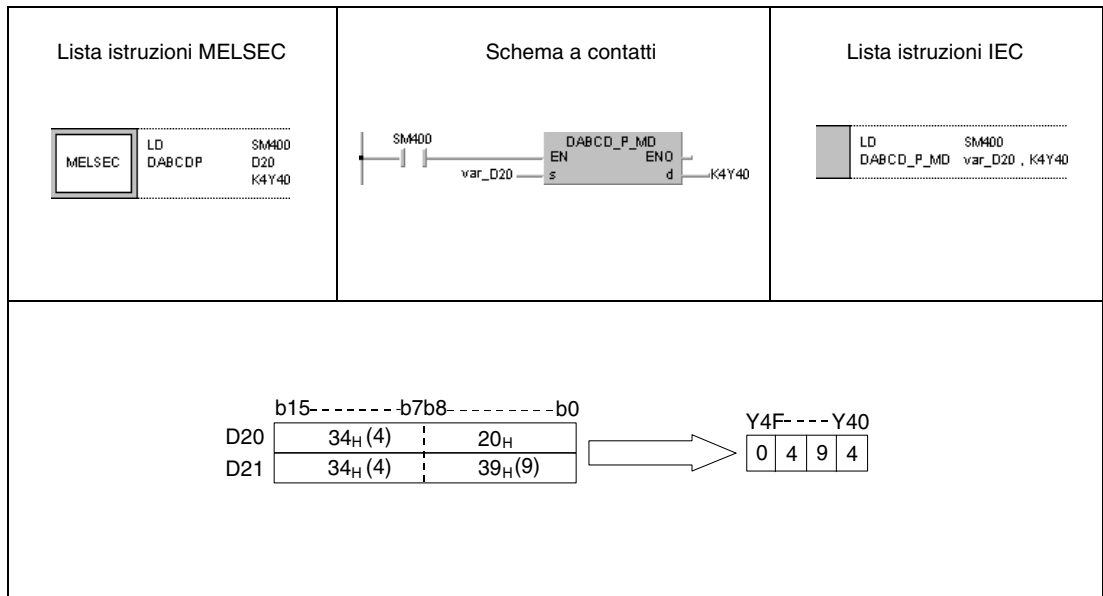
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il codice ASCII nei registri separati da s (Array_s[1]) a s+3 (Array_s[4]) non è nel campo ammesso da "30H" a "39H" (codice di errore 4100).

Programma di esempio 1

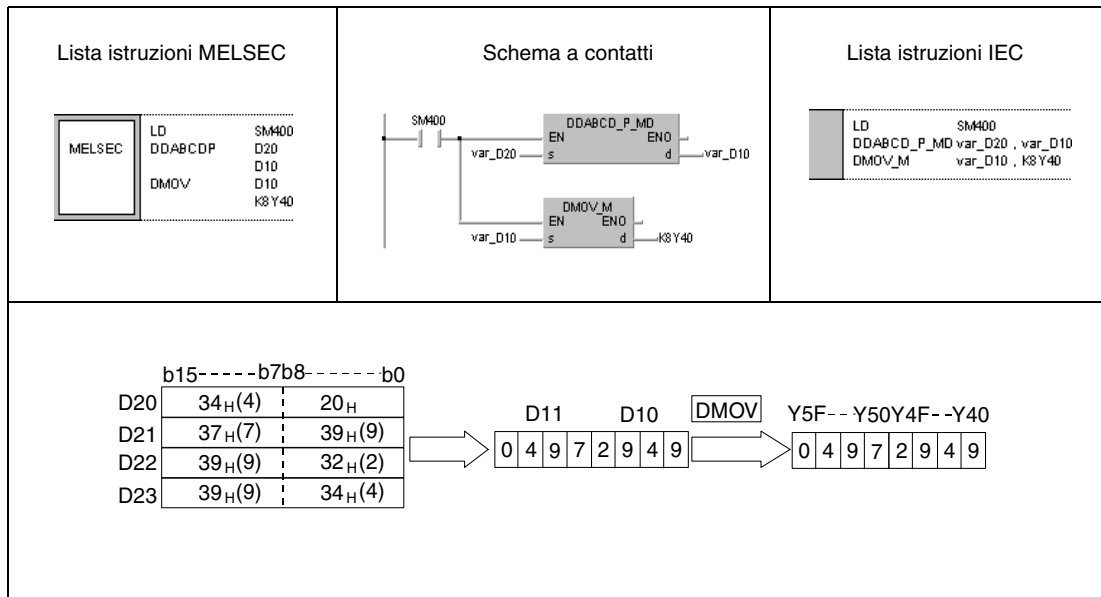
DABCDP

Con il fronte positivo di SM400, il programma seguente converte il valore ASCII da D20 (var_D20 Array [0]) a D21 (var_D20 Array [1]) in valore BCD a 4 cifre e lo memorizza da Y40 a Y4F.



Programma di esempio 2 DDABCDP

Con il fronte positivo di SM400, il programma seguente converte il valore ASCII da D20 (var_D20 Array [0]) a D23 (var_D20 Array [3]) in valore BCD a 8 cifre, lo memorizza in D10 e D11 e lo invia sulle uscite da Y40 a Y5F.



7.11.7 COMRD, COMRDP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

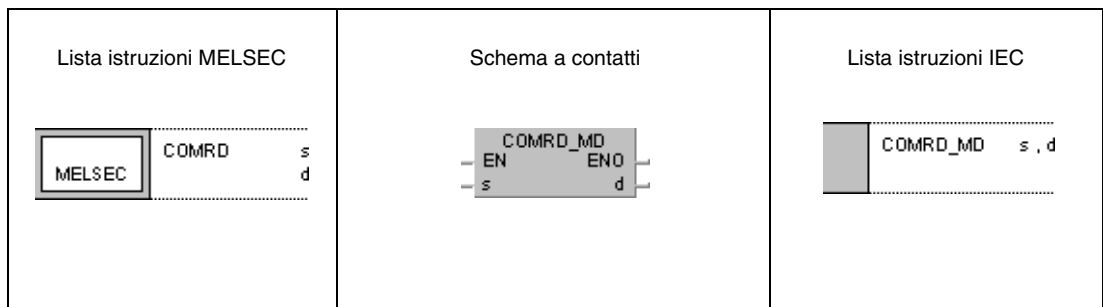
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

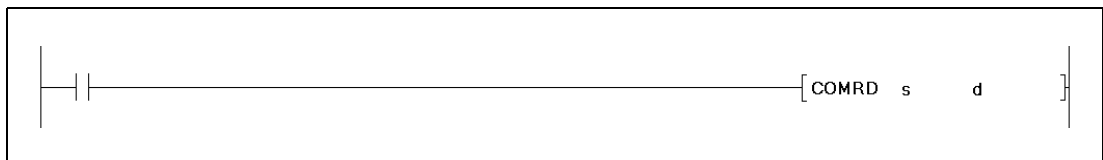
Operandi MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro BL\S, BL\TR, BL, P, I, J, U		
	Bit	Word		Bit	Word						
s	●	●	●	●	●	—	—	●	SM0	3	
d	—	●	●	—	—	—	—	—			

GX IEC Developer



GX Developer

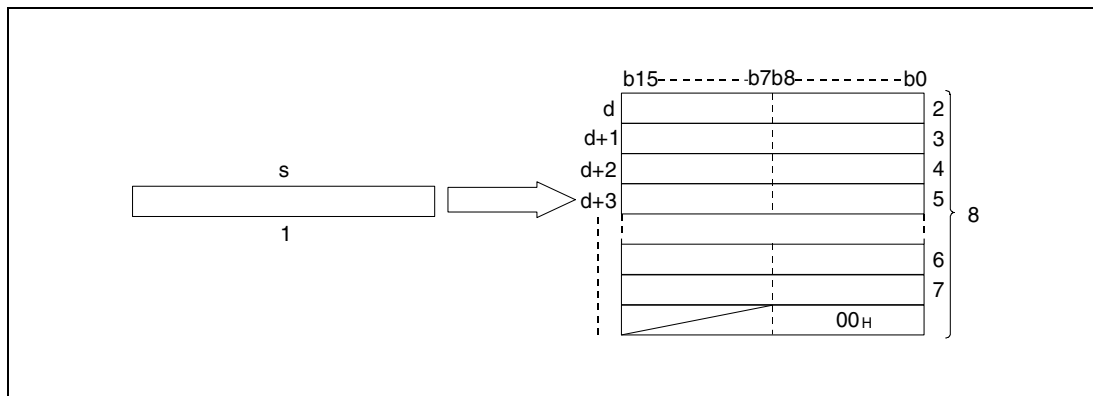


Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s	Indirizzo del primo operando che contiene il commento da leggere.	Indirizzo operando	ANY16
d	Indirizzo del primo operando che memorizza il commento letto.	Stringa caratteri	Array [1.0.8] di ANY16

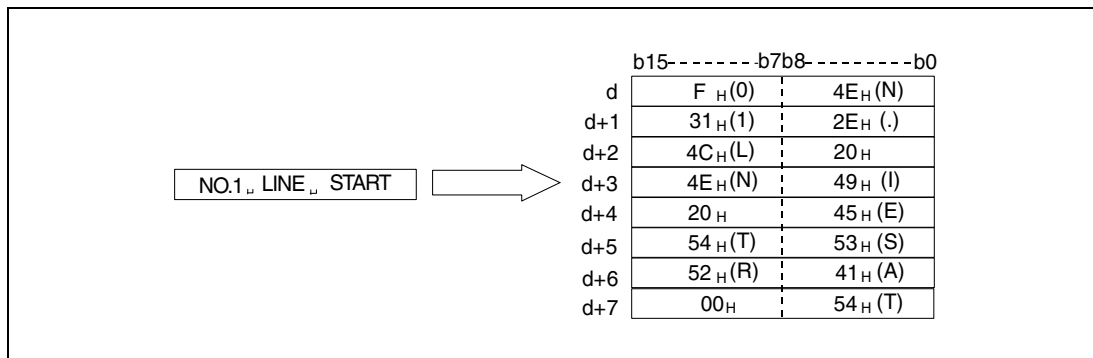
Funzioni **Letture commento operando****COMRD Istruzione di lettura**

L'istruzione COMRD legge il commento dall'operando indicato da s e lo memorizza in codice ASCII nell'area da d (Array_d[1]) a d+7 (Array_d[8]).



- ¹ Commento
² Codice ASCII del secondo carattere/ codice ASCII del primo carattere
³ Codice ASCII del quarto carattere/ codice ASCII del terzo carattere
⁴ Codice ASCII del sesto carattere/ codice ASCII del quinto carattere
⁵ Codice ASCII dell'ottavo carattere/ codice ASCII del settimo carattere
⁶ Codice ASCII del trentesimo carattere/ codice ASCII del ventinovesimo carattere
⁷ Codice ASCII del trentaduesimo carattere/ codice ASCII del trentunesimo carattere
⁸ Memorizza un massimo di 32 caratteri.

I dati di commento contenuti in s nella sequenza "NO.1 LINE START" vengono memorizzati a partire da d (Array_d[1]), come segue:



La zona di indirizzi degli operandi indicati da s deve essere posta all'interno dell'area per i commenti.

Se in s non viene specificato nessun commento, i caratteri vengono convertiti in spazi.

Un commento non deve superare la lunghezza massima di 32 caratteri.

Il contenuto del byte che segue l'ultimo carattere dipende dallo stato del relé speciale SM701, come segue:

Se SM701 non è attivo, viene inserito uno zero

Se SM701 è attivo, il dato non viene modificato.

SM720 viene attivato per una scansione di programma al termine dell'esecuzione dell'istruzione COMRD. SM721 è attivo (ON) durante l'esecuzione dell'istruzione COMRD. Se SM721 è già attivo, quando si attiva l'istruzione COMRD, non viene eseguita alcuna elaborazione.

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- L'area di indirizzi indicata da s supera il campo dei commenti (codice di errore 4100).
- L'istruzione COMRD(P) viene eseguita mentre si scrive un commento in RUN (codice di errore 4100).
- Il file specificato non esiste (codice di errore 2410).

NOTA

Una CPU del System Q termina l'elaborazione di COMRD(P) dopo diverse scansioni. Una CPU QnA esegue tutta l'elaborazione in una scansione.

Il segnale di avvio (comando) dell'istruzione COMRD(P) è disabilitato, se viene attivato prima del completamento di una precedente istruzione COMRD(P) (SM720 deve essere stato attivato).

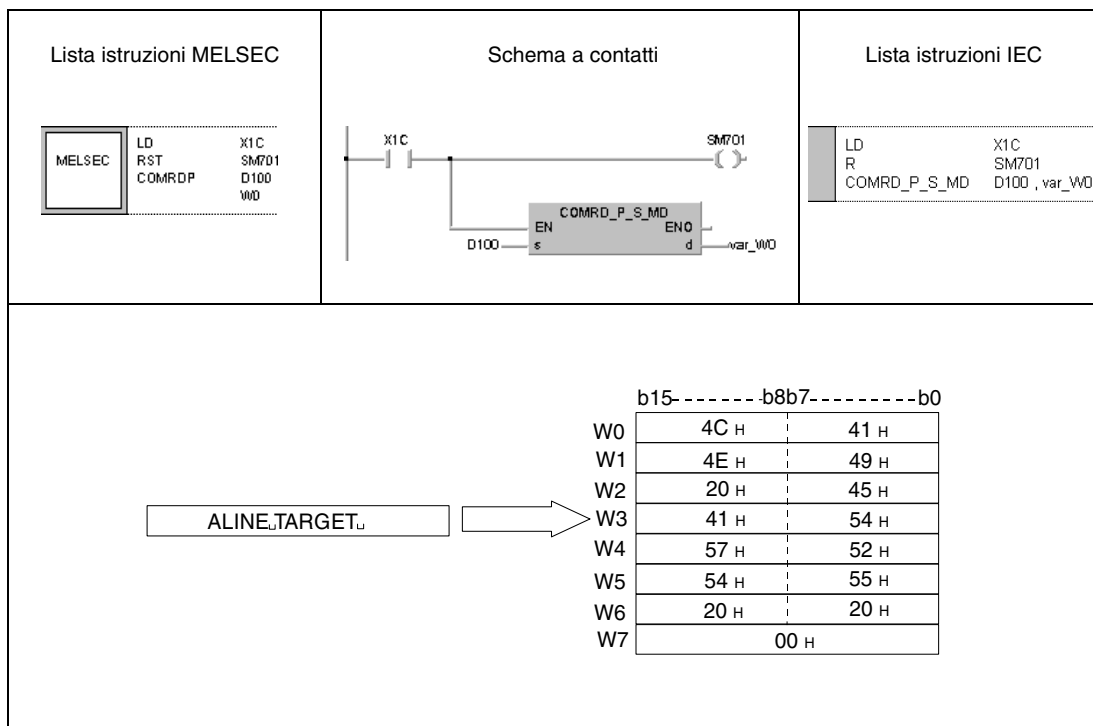
Non è possibile accedere contemporaneamente a due o più file di commento.

Le istruzioni che seguono non possono essere eseguite contemporaneamente a causa dell'uso comune di SM721:

Istruzione	ON durante l'esecuzione	ON per una scansione al termine dell'esecuzione dell'istruzione	ON se l'esecuzione dell'istruzione termina con errore
S.FREAD S.FWRITE	SM721	Bit indicato dall'istruzione	Bit indicato dall'istruzione + bit successivo
PRC COMRD		SM720	—

Esempio di programma COMRDP

Con il fronte positivo di X1C, il programma seguente memorizza un commento specificato da D100 come codice ASCII da W0 (var_W0 Array [0]) a W7 (var_W0 Array [7]).

**NOTA**

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.11.8 LEN, LENP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

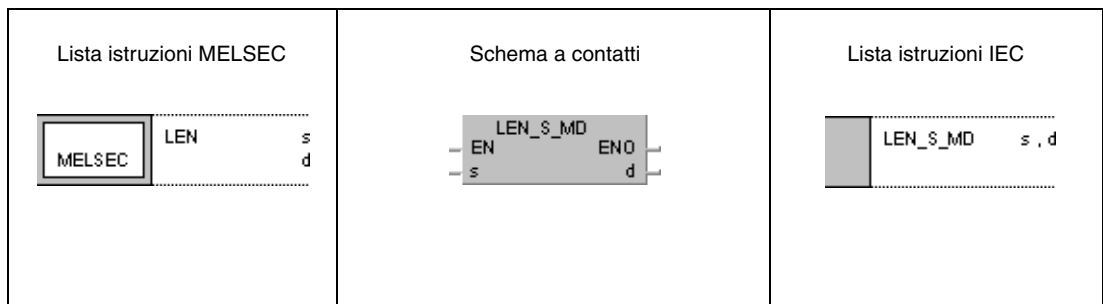
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

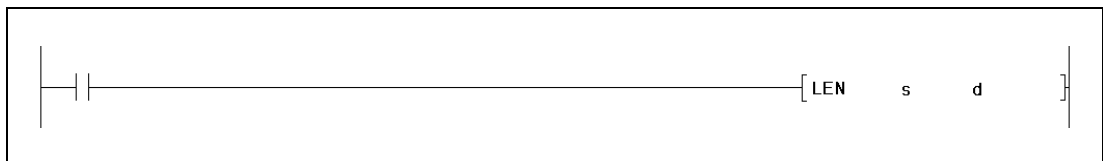
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti \$			Altro U
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	●	—	SM0	3
d	●	●	●	●	●	●	●	—	—		

GX IEC Developer



GX Developer

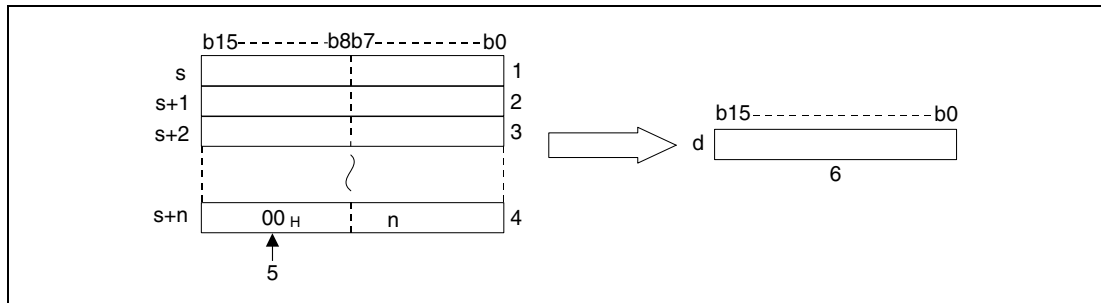


Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene la stringa di caratteri di cui si vuole conoscere la lunghezza.	Stringa caratteri
d	Indirizzo dell'area contenente la lunghezza rilevata della stringa di caratteri.	BIN 16-bit

Funzioni **Rilevamento della lunghezza di stringhe di caratteri****LEN** **Rilevamento lunghezza**

L'istruzione di lunghezza rileva la lunghezza della stringa di caratteri specificata da s e memorizza il risultato nell'operando indicato da d.



¹ Secondo carattere/ primo carattere

² Quarto carattere/ terzo carattere

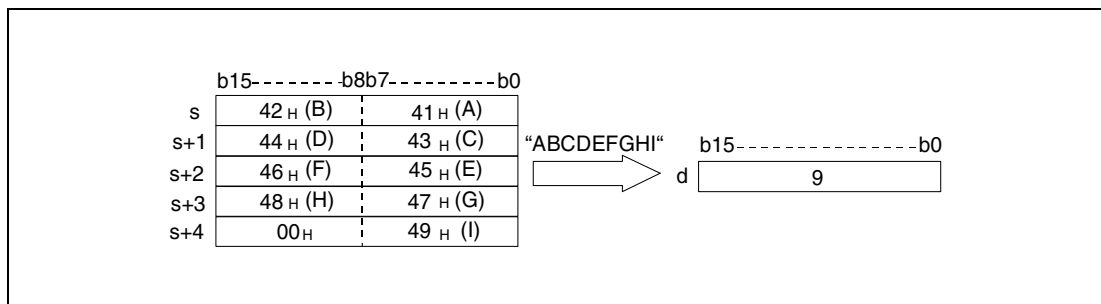
³ Sesto carattere/ quinto carattere

⁴ ennesimo carattere

⁵ Fine della stringa

⁶ Lunghezza della stringa

La lunghezza della stringa di caratteri "ABCDEFGHI" contenuta in s viene memorizzata in d come "9":



La stringa di caratteri contenuta in s viene elaborata fino ad incontrare il codice di carattere "00H". Il risultato è memorizzato in d.

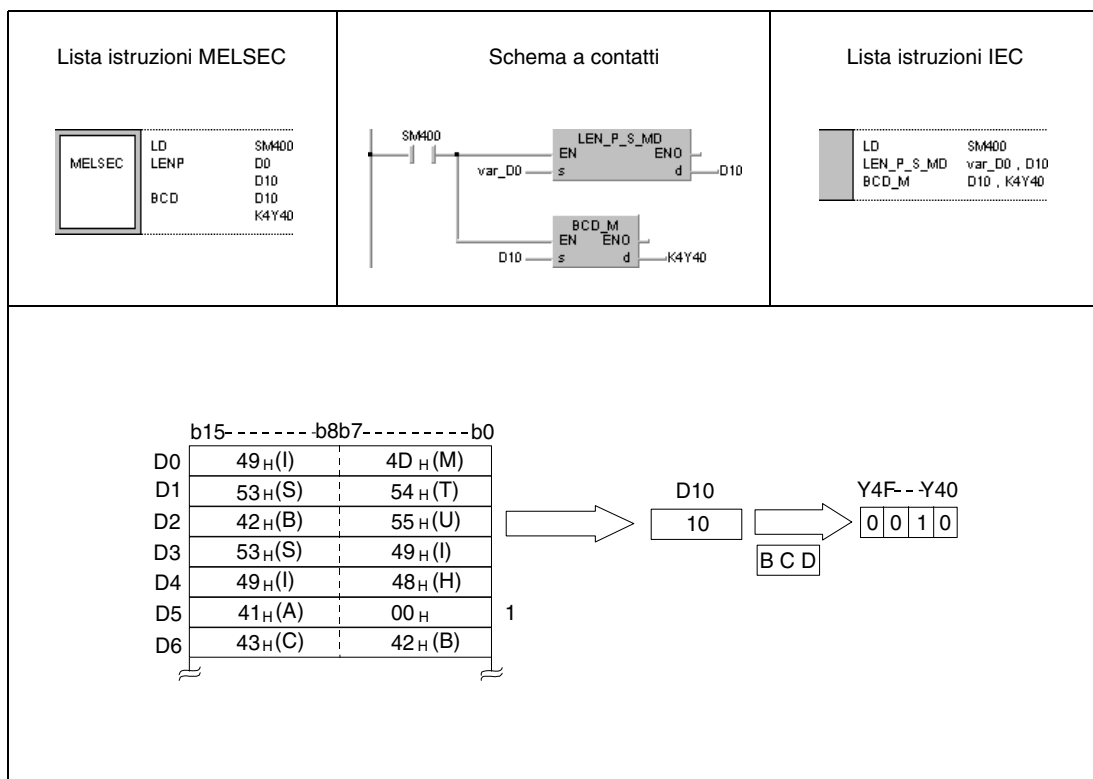
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il codice carattere "00H" assente nell'ultimo byte di s (codice di errore 4101).

Esempio di programma LENP

Con il fronte di salita di SM400, il programma seguente legge la stringa di caratteri memorizzata in D0, rileva la sua lunghezza ed invia il risultato come dato BCD a 4 cifre sulle uscite da Y40 a Y4F.



¹ I caratteri successivi al codice "00H" non vengono considerati (viene rilevata solo la lunghezza della stringa "MITSUBISHI")

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.11.9 STR, STRP, DSTR, DSTRP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

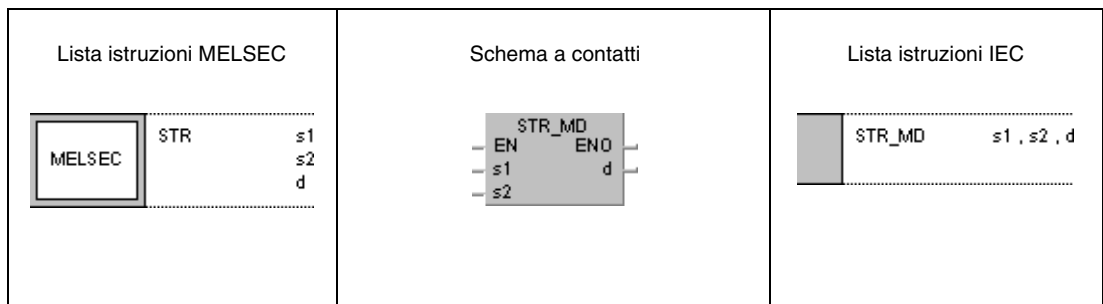
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

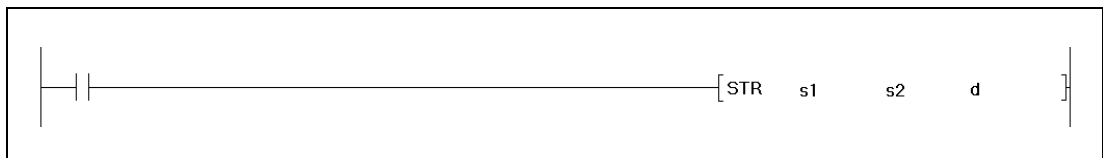
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	●	●	●	●	●	●	—	—	SM0	4	
s2	●	●	●	●	●	●	●	—			
d	—	●	●	—	—	—	—	—			

GX IEC Developer



GX Developer



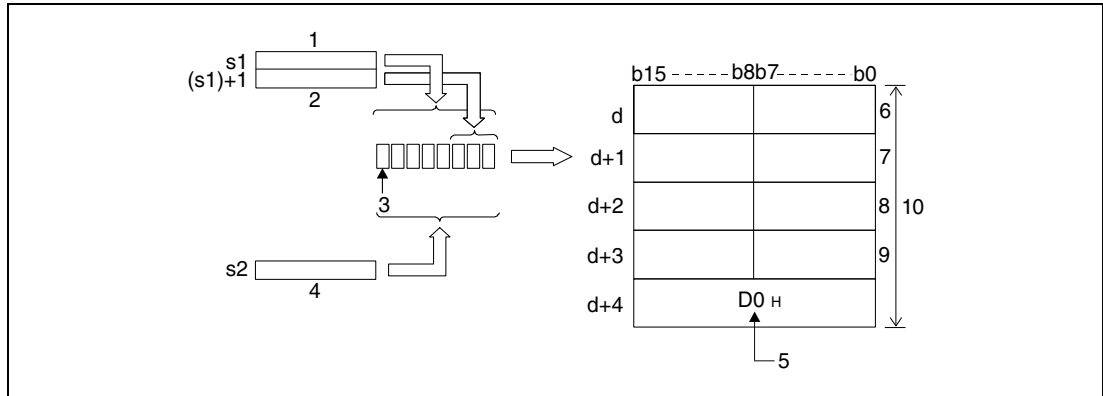
Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s1	Indirizzo del primo operando che contiene il numero di cifre del valore numerico da convertire.	BIN 16-bit	ANY32
s2	Dato binario da convertire.	BIN 16/32-bit	ANY16/32
d	Indirizzo del primo operando che memorizza la stringa di caratteri convertita.	Stringa caratteri	Array [1.0.5]/ [1..6] di ANY16

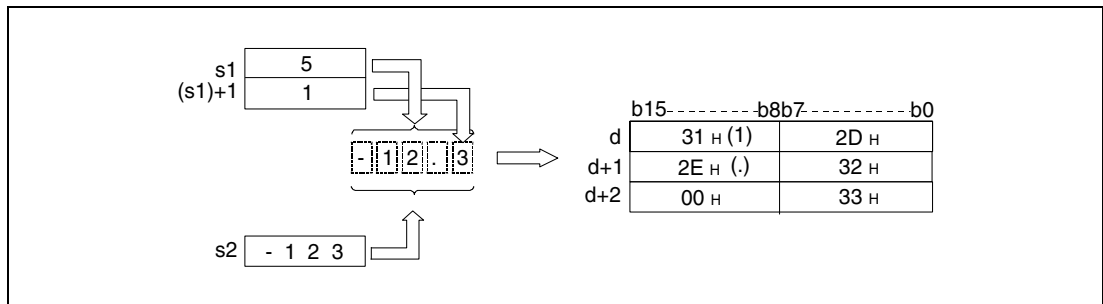
Funzioni Conversione dati BIN a 16/32 bit in stringhe di caratteri

STR Conversione di dati binari a 16-bit

L'istruzione STR aggiunge un punto decimale al valore binario a 16 bit nell'operando indicato da s2, in corrispondenza della cifra indicata dagli operandi s1 e s1+1, poi converte il dato in una stringa di caratteri e lo memorizza nell'area di operandi indicata da d (Array_d[1]) a d+4 (Array_d[5]).



- 1 Numero delle cifre
- 2 Posti decimali
- 3 Segno
- 4 Valore binario
- 5 Indicazione di fine stringa, inserita automaticamente.
- 6 Posizione carattere in ASCII; numero cifre -1/ codice ASCII del segno
- 7 Posizione carattere in ASCII; numero cifre -3/ posizione carattere in ASCII; numero cifre -2
- 8 Posizione carattere in ASCII; numero cifre -5/ posizione carattere in ASCII; numero cifre -4
- 9 Posizione carattere in ASCII; numero cifre -7/ posizione carattere in ASCII; numero cifre -6
- 10 Numero delle cifre

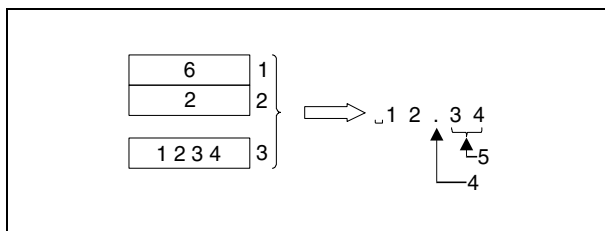


Il numero di cifre contenuto nell'operando indicato da s1 può variare da 2 a 8.
 Il numero di posti decimali contenuto nell'operando indicato da (s1)+1 può variare da 0 a 5 e non deve superare il numero di cifre meno 3.
 Il dato a 16 bit contenuto nell'operando indicato da s2 può variare da -32768 a 32767.

Dopo la conversione in stringa di caratteri, la stringa viene memorizzata negli operandi da d (Array_d[1]) a d+4 (Array_d[5]) come segue:

Il segno positivo del dato binario viene memorizzato con il carattere ASCII "20H" (blank).
 Il segno negativo del dato binario viene memorizzato con il carattere ASCII "2DH" (carattere "meno").

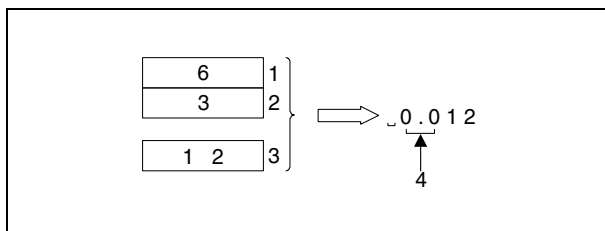
Se il numero di posti decimali è maggiore di zero, il punto decimale "2EH" (.) viene inserito automaticamente prima della prima cifra specificata.



- ¹Numero delle cifre
- ²Numero di posti decimali
- ³Valore binario
- ⁴Punto decimale inserito automaticamente
- ⁵Posti decimali

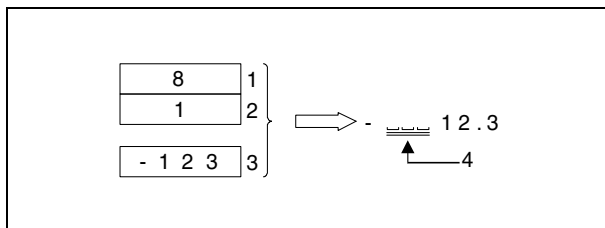
Se il numero di posti decimali è zero, il carattere "2DH" (.) non viene inserito.

Se il numero dei posti decimali è superiore al numero di cifre del valore binario, le cifre mancanti vengono sostituite da zeri, il valore binario viene fatto scorrere verso destra, ed il punto decimale viene inserito di conseguenza (0.□□□□□).



- ¹Numero delle cifre
- ²Numero di posti decimali
- ³Valore binario
- ⁴Zeri e punto decimale inseriti automaticamente

Se il numero di cifre, compreso segno e punto decimale, è maggiore del numero di cifre del valore binario, le cifre mancanti fra segno e valore numerico sono automaticamente sostituite da "20H" (blank).

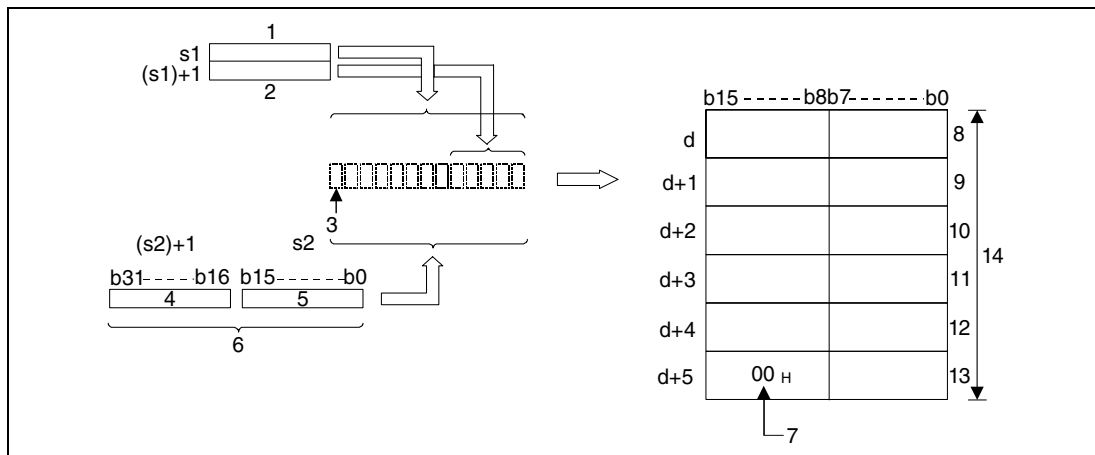


- ¹Numero delle cifre
- ²Numero di posti decimali
- ³Valore binario
- ⁴Caratteri spazio (blank) inseriti automaticamente.

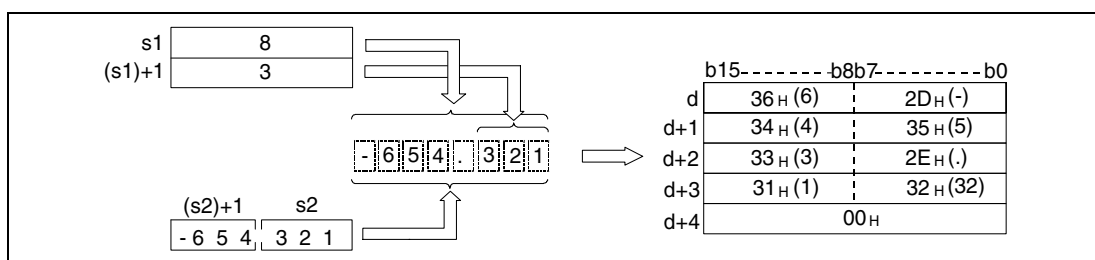
Alla fine della stringa di caratteri risultante, viene inserito automaticamente il codice "00H".

DSR Conversione di dati BIN a 32-bit

L'istruzione DSTR aggiunge un punto decimale al valore binario a 32 bit nell'operando indicato da s2 e (s2)+1, in corrispondenza della cifra indicata dagli operandi s1 e (s1)+1, poi converte il dato in una stringa di caratteri e lo memorizza nell'area di operandi indicata da d (Array_d[1]) a d+5 (Array_d[6]).



- 1 Numero delle cifre
- 2 Posti decimali
- 3 Segno
- 4 16 bit più significativi
- 5 16 bit meno significativi
- 6 Valore binario
- 7 Indicazione di fine stringa, inserita automaticamente.
- 8 Posizione carattere in ASCII; numero cifre -1/ codice ASCII del segno
- 9 Posizione carattere in ASCII; numero cifre -3/ posizione carattere in ASCII; numero cifre -2
- 10 Posizione carattere in ASCII; numero cifre -5/ posizione carattere in ASCII; numero cifre -4
- 11 Posizione carattere in ASCII; numero cifre -7/ posizione carattere in ASCII; numero cifre -6
- 12 Posizione carattere in ASCII; numero cifre -9/ posizione carattere in ASCII; numero cifre -8
- 13 Indicazione fine stringa/ posizione carattere in ASCII; numero cifre -10
- 14 Numero delle cifre



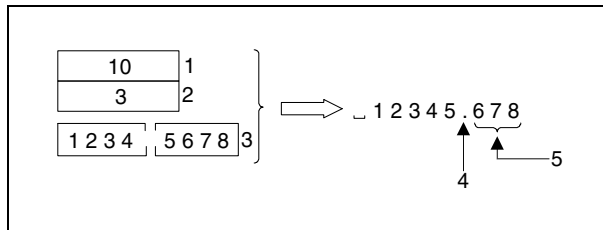
Il numero di cifre contenuto nell'operando indicato da s1 può variare da 2 a 13.
 Il numero di posti decimali contenuto nell'operando indicato da (s1)+1 può variare da 0 a 10 e non deve superare il numero di cifre meno 3.
 Il dato a 32 bit contenuto nell'operando indicato da s2 e (s2)+1 può variare da -2147483648 a 32147483647.

Dopo la conversione in stringa di caratteri, la stringa viene memorizzata negli operandi da d (Array_d[1]) a d+5 (Array_d[6]) come segue:

Il segno positivo del dato binario viene memorizzato con il carattere ASCII "20H" (blank).

Il segno negativo del dato binario viene memorizzato con il carattere ASCII "2DH" (carattere "meno").

Se il numero di posti decimali è maggiore di zero, il punto decimale "2EH" (.) viene inserito automaticamente prima della prima cifra specificata.



¹Numero delle cifre

²Numero di posti decimali

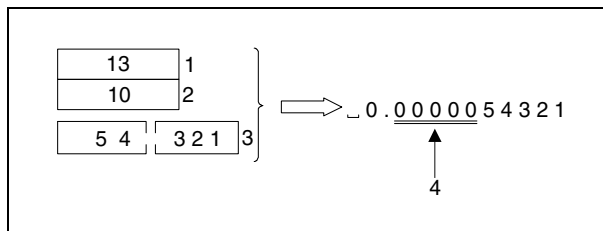
³Valore binario

⁴Punto decimale inserito automaticamente

⁵Posti decimali

Se il numero di posti decimali è zero, il carattere "2DH" (.) non viene inserito.

Se il numero dei posti decimali è superiore al numero di cifre del valore binario, le cifre mancanti vengono sostituite da zeri, il valore binario viene fatto scorrere verso destra, ed il punto decimale viene inserito di conseguenza (0.□□□□□).



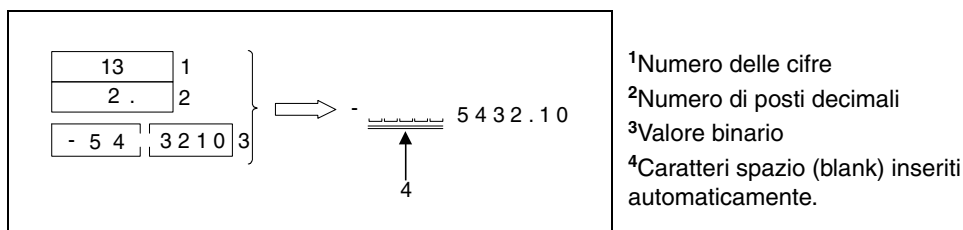
¹Numero delle cifre

²Posti decimali

³Valore binario

⁴Zeri e punto decimale inseriti automaticamente

Se il numero di cifre, compreso segno e punto decimale, è maggiore del numero di cifre del valore binario, le cifre mancanti fra segno e valore numerico sono automaticamente sostituite da "20H" (blank).



Alla fine della stringa di caratteri risultante, viene inserito automaticamente il codice "00H".

Errori di esecuzione

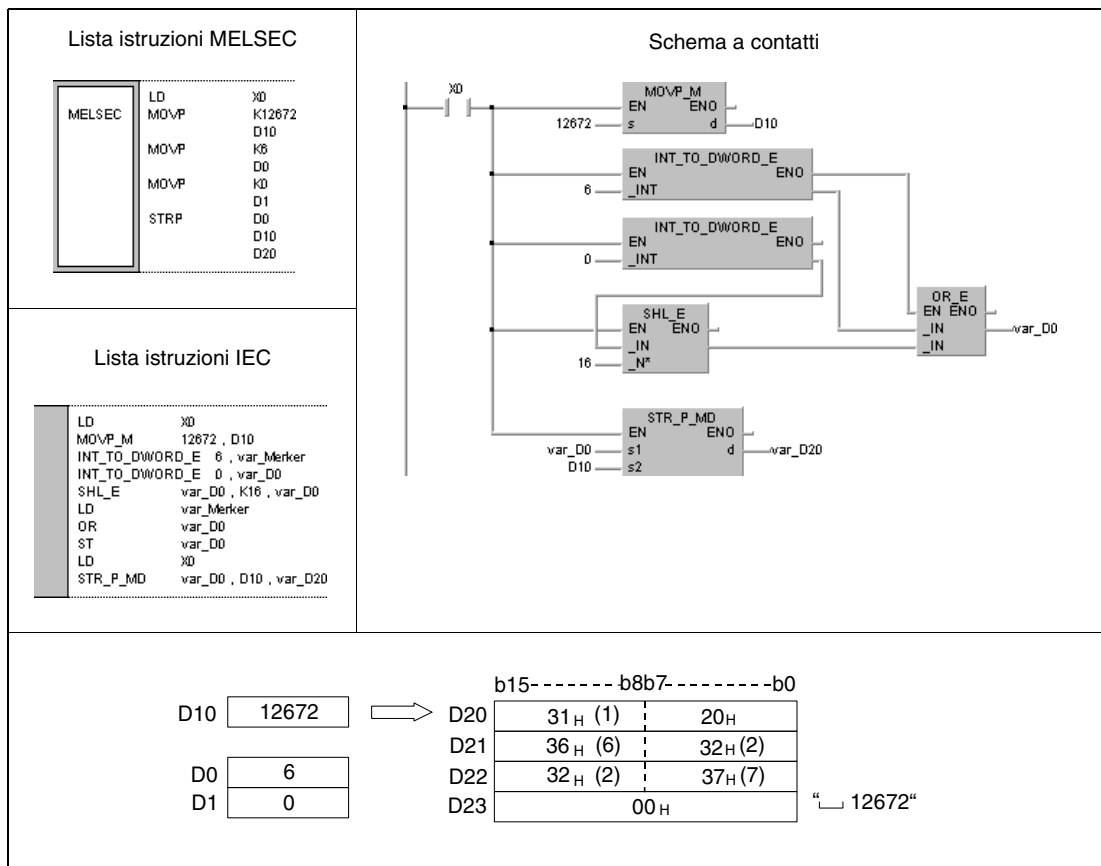
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il numero di cifre contenuto in s1 supera il campo di valori indicato di seguito (codice di errore 4100):
 Campo di valori per istruzione STR: da 2 a 8
 Campo di valori per istruzione DSTR: da 2 a 13
- Il numero di cifre contenuto in (s1)+1 supera il campo di valori indicato di seguito (codice di errore 4100):
 Campo di valori per istruzione STR: da 0 a 5
 Campo di valori per istruzione DSTR: da 0 a 10
- I valori contenuti in s1 e (s1)+1 non corrisponde alla relazione seguente:
 Il numero di cifre meno 3 è maggiore o uguale al numero di posti decimali (codice di errore 4100).
- Il numero delle cifre contenute in s1 e (s1)+1 è inferiore alle cifre dei valori binari in s2 e (s2)+1 (codice di errore 4100).
- L'area che contiene la stringa di caratteri indicata a partire da d (Array_d[1]) supera il campo operando ammesso (codice di errore 4100).

**Programma
di esempio 1**

STRP

Con il fronte positivo di X0, il programma seguente converte il valore binario specificato da D10 nel numero di cifre indicate da D0 e D1. Il risultato viene memorizzato nell'area da D20 (var_D20 Array [1]) a D23 (var_D20 Array [4]).



Programma di esempio 2 DSTRP

Con il fronte positivo di X0, il programma seguente converte il valore binario specificato da D10 e D11 nel numero di cifre indicate da D0 e D1. Il risultato viene memorizzato nell'area da D20 (var_D20 Array [1]) a D23 (var_D20 Array [7]).

Lista istruzioni MELSEC

```

LD      X0
DMOV_P_M  K12345678, var_D11
MOV_P   D11, K6
MOV_P   D2, K0
MOV_P   D3, K0
DSTRP   D2, D11, D30
        
```

Schema a contatti

Lista istruzioni IEC

```

LD      X0
DMOV_P_M  12345678, var_D11
INT_TO_DWORD_E  6, var_Marker
INT_TO_DWORD_E  0, var_D1
SHL_E      var_D1, K16, var_D1
LD      var_Marker
OR      var_D1
ST      var_D1
LD      X0
DSTR_P_MD  var_D1, var_D11, var_D30
        
```

D12	D11	D30	b15-----b8	b7-----b0	“ 0.012345678 ”
12345678			30 H (0)	20 H	
D0	12		30 H (0)	2E H (.)	
D1	9		32 H (2)	31 H (1)	
			34 H (4)	33 H (3)	
			36 H (6)	35 H (5)	
			38 H (8)	37 H (7)	
			00 H		

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provocano messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.11.10 VAL, VALP, DVAL, DVALP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

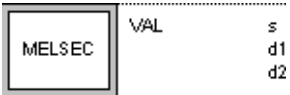
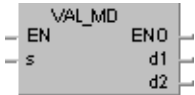
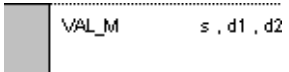
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

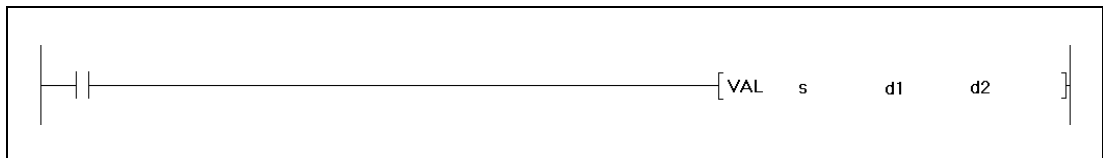
Operandi MELSEC Q

Operandi utilizzabili									Errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Indice file Zn	Costanti \$	Altro		
Bit	Word		Bit	Word						
—	●	●	—	—	—	—	●	—	SM0	4
●	●	●	—	—	—	—	—	—		
●	●	●	●	●	●	●	—	—		

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
		

GX Developer



Variabili

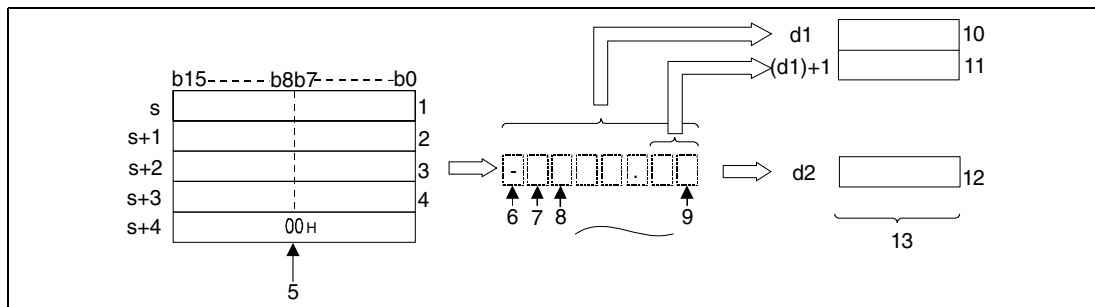
Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s	Indirizzo del primo operando contenente la stringa di caratteri dei dati binari da convertire.	Stringa caratteri	Array [1.0.5]/ [1.0.7] di ANY16
d1	Indirizzo del primo operando contenente il numero di cifre del dato binario dopo la conversione.	BIN 16-bit	ANY32
d2	Indirizzo del primo operando che memorizza il dato binario convertito.	BIN 16/32-bit	ANY16/32

Funzioni Conversione da stringhe di caratteri a dati BIN a 16/32 bit

VAL Conversione in dato binario a 16-bit

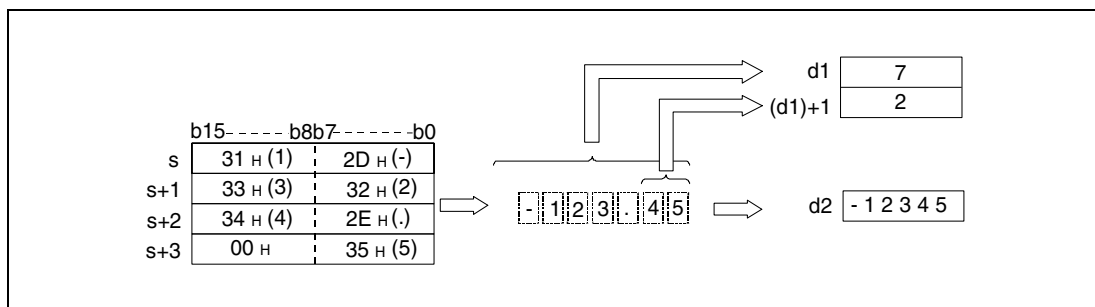
L'istruzione VAL converte la stringa di caratteri memorizzata nell'area da s (Array_s[1]) a s+4 (Array_s[5]) in un dato BIN a 16 bit. Il numero di cifre ed il valore binario vengono memorizzati in d1, (d1)+1, e d2.

Per la conversione nel formato BIN 16-bit tutti i dati contenuti nell'area da s (Array_s[1]) a s+4 (Array_s[5]) vengono considerati come stringa di caratteri fino al codice "00H".



- 1 Codice ASCII del primo carattere/ codice ASCII del segno
- 2 Codice ASCII del terzo carattere/ codice ASCII del secondo carattere
- 3 Codice ASCII del quinto carattere/ codice ASCII del quarto carattere
- 4 Codice ASCII del settimo carattere/ codice ASCII del sesto carattere
- 5 Indica la fine della stringa di caratteri
- 6 Carattere segno
- 7 Primo carattere
- 8 Secondo carattere
- 9 Settimo carattere
- 10 Numero delle cifre
- 11 Numero di posti decimali
- 12 Valore intero, il punto decimale non viene considerato
- 13 BIN 16-bit

Si deve convertire la stringa "-123.45" contenuta nell'area da s (Array_s[1]) a s+4 (Array_s[5]). Il risultato viene memorizzato in d1, (d1)+1 e d2 come segue:



Il numero totale dei caratteri contenuti da s (Array_s[1]) a s+4 (Array_s[5]) può variare nel campo da 2 a 8.

Il numero delle cifre decimali memorizzate nell'area da s (Array_s[1]) a s+4 (Array_s[5]) può variare da 0 a 5. In generale, il numero delle cifre decimali non deve superare il numero totale delle cifre meno 3.

Il valore numerico di una stringa di caratteri da convertire, senza considerare il punto decimale, deve essere nel campo da -32768 a 32767.

Il valore numerico dei caratteri ASCII della stringa, senza considerare il carattere del segno e il punto decimale, deve essere nel campo da "30H" a "39H".

Il segno positivo del dato binario viene memorizzato con il carattere ASCII "20H" (blank).

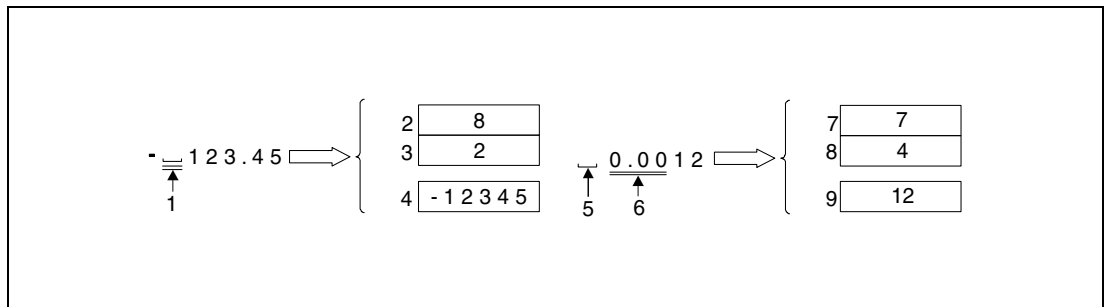
Il segno negativo del dato binario viene memorizzato con il carattere ASCII "2DH" (carattere "meno").

Il punto decimale viene memorizzato con il carattere ASCII "2EH".

Il numero totale di cifre memorizzate in d1, (d1)+1, e d2 comprende tutti i caratteri che rappresentano il valore numerico, comprensivo del segno d1 e delle cifre decimali (d1)+1.

Nel dato binario memorizzato in d2 dopo la conversione, il punto decimale viene ignorato.

Se i caratteri "20H" (blank) o "30H" (zero) compaiono fra il carattere del segno ed il primo valore numerico, vengono ignorati nella conversione.

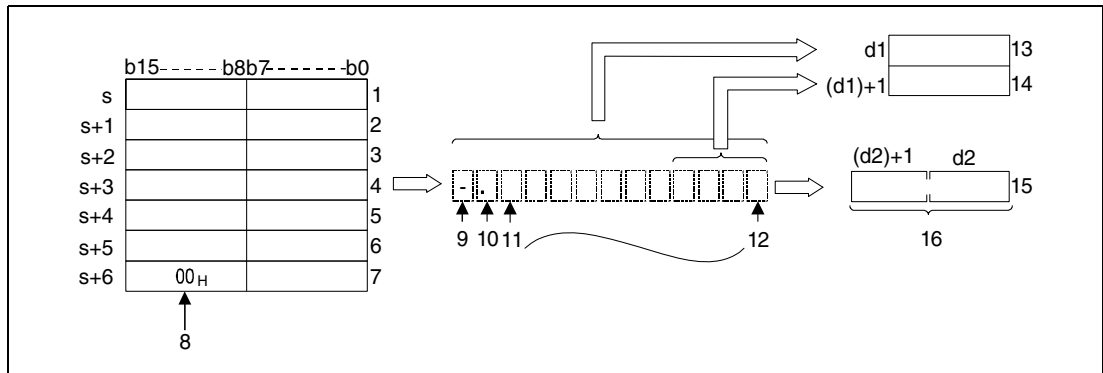


- ¹ Questi caratteri non vengono considerati
- ² Numero delle cifre
- ³ Numero di posti decimali
- ⁴ Valore binario
- ⁵ Carattere segno
- ⁶ Questi caratteri non vengono considerati
- ⁷ Numero delle cifre
- ⁸ Numero di posti decimali
- ⁹ Valore binario

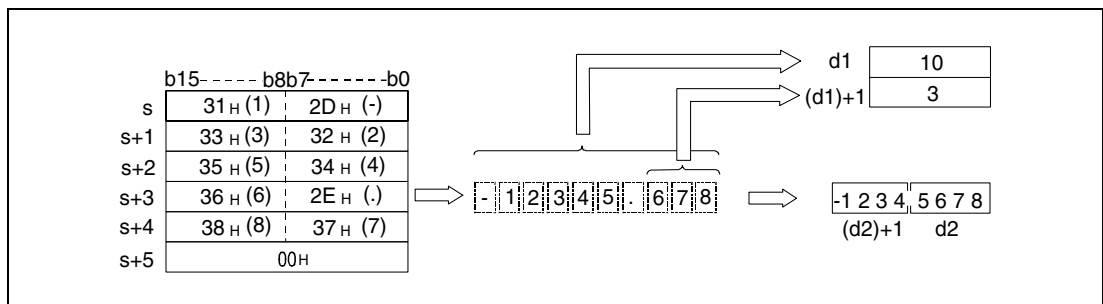
DVAL Conversione in dati BIN a 32-bit

L'istruzione DVAL converte la stringa di caratteri memorizzata nell'area da s (Array_s[1]) a s+6 (Array_s[7]) in un dato BIN a 32 bit. Il numero di cifre ed il valore binario vengono memorizzati in d1, (d1)+1, d2 e (d2)+1.

Per la conversione nel formato BIN 32-bit tutti i dati contenuti nell'area da s (Array_s[1]) a s+6 (Array_s[7]) vengono considerati come stringa di caratteri fino al codice "00H".



- 1 Codice ASCII del primo carattere/ codice ASCII per il carattere del segno
- 2 Codice ASCII del terzo carattere/ codice ASCII del secondo carattere
- 3 Codice ASCII del quinto carattere/ codice ASCII del quarto carattere
- 4 Codice ASCII del settimo carattere/ codice ASCII del sesto carattere
- 5 Codice ASCII del nono carattere/ codice ASCII dell'ottavo carattere
- 6 Codice ASCII dell'undicesimo carattere/ codice ASCII del decimo carattere
- 7 Codice ASCII del carattere zero/ codice ASCII del dodicesimo carattere
- 8 Indica la fine della stringa di caratteri
- 9 Carattere segno
- 10 Primo carattere
- 11 Secondo carattere
- 12 Dodicesimo carattere
- 13 Numero delle cifre
- 14 Numero di posti decimali
- 15 Valore intero, il punto decimale non viene considerato
- 16 BIN 32-bit



Il numero totale dei caratteri contenuti da s (Array_s[1]) a s+6 (Array_s[7]) può variare nel campo da 2 a 13.

Il numero delle cifre decimali memorizzate nell'area da s (Array_s[1]) a s+6 (Array_s[7]) può variare da 0 a 10. In generale, il numero delle cifre decimali non deve superare il numero totale delle cifre meno 3.

Il valore numerico di una stringa di caratteri da convertire, senza considerare il punto decimale, deve essere nel campo da -2147483648 a 2147483647.

Il valore numerico dei caratteri ASCII della stringa, senza considerare il carattere del segno e il punto decimale, deve essere nel campo da "30H" a "39H".

Il segno positivo del dato binario viene memorizzato con il carattere ASCII "20H" (blank).

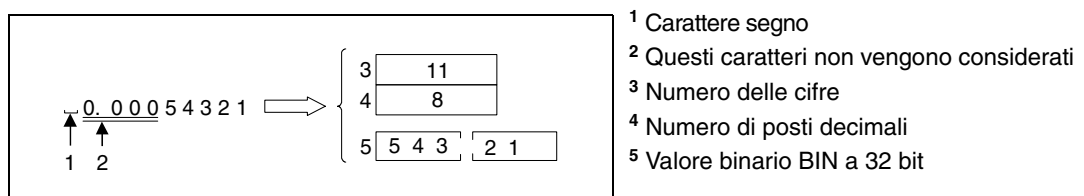
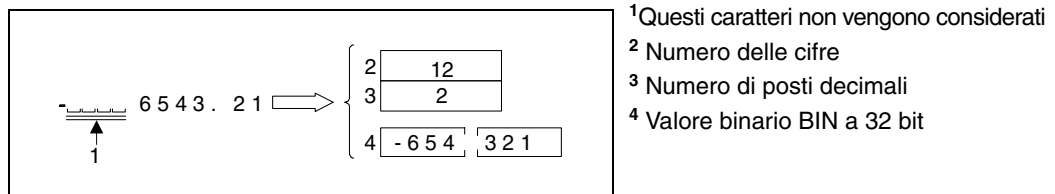
Il segno negativo del dato binario viene memorizzato con il carattere ASCII "2DH" (carattere "meno").

Il punto decimale viene memorizzato con il carattere ASCII "2EH".

Il numero totale di cifre memorizzate in d1, (d1)+1, d2 e (d2)+1 comprende tutti i caratteri che rappresentano il valore numerico, comprensivo del segno d1 e delle cifre decimali (d1)+1.

Nel dato binario memorizzato in d2 e (d2)+1 dopo la conversione, il punto decimale viene ignorato.

Se i caratteri "20H" (blank) o "30H" (zero) compaiono fra il carattere del segno ed il primo valore numerico, vengono ignorati nella conversione.



Errori di esecuzione

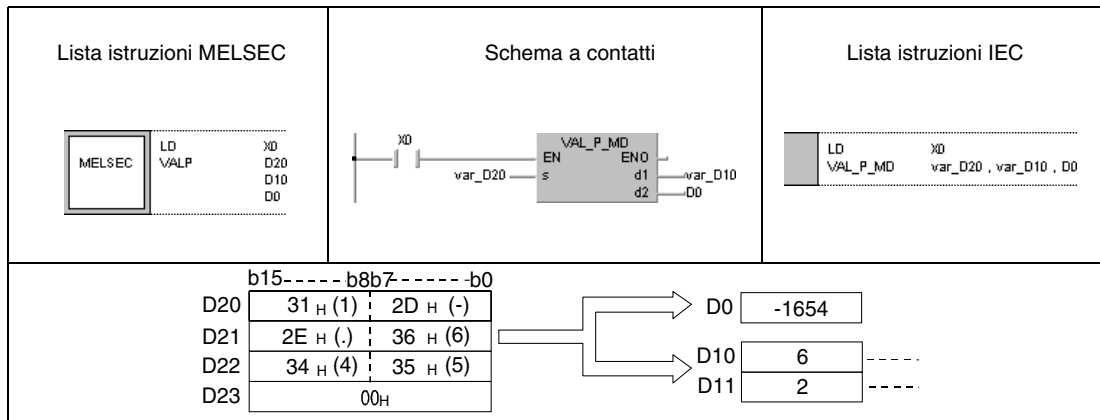
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il numero di cifre contenute a partire da s (Array_s[1]) supera il campo di valori da 2 a 8 (VAL) o da 2 a 13 (DVAL) (codice di errore 4101).
- Il numero di cifre decimali memorizzato in (d1)+1 supera il campo di valori da 0 a 5 (VAL) o da 0 a 10 (DVAL) (codice di errore 4100).
- Il numero di cifre meno 3 è maggiore o uguale al numero di posti decimali (codice di errore 4100).
- Il carattere del segno contiene un codice ASCII diverso da "20H" o "2DH". (codice di errore 4100).
- Valore specificato con caratteri ASCII diversi da "30H", "39H", o "2EH" (codice di errore 4100).
- Il valore di partenza contiene più di un punto decimale (codice di errore 4100).
- Il valore binario supera il campo di valori da -32768 a 32767 (VAL) oppure da -2147483648 a 2147483647 (DVAL) dopo la conversione (codice di errore 4100)
- Il carattere ASCII "00H" non è posizionato nella cifra corretta (codice di errore 4100).

Programma di esempio 1

VALP

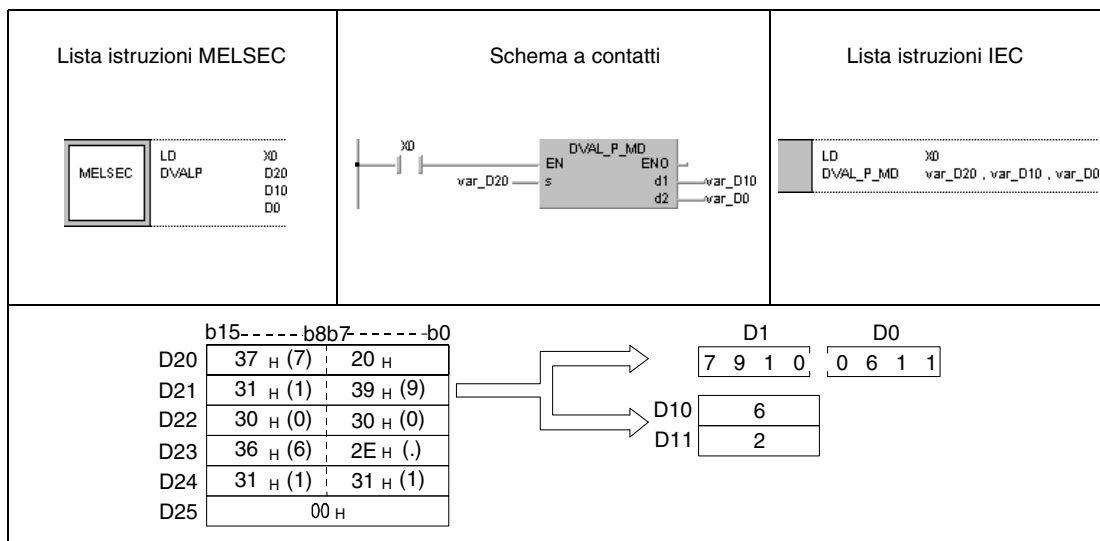
Con il fronte positivo di X0, il programma seguente converte la stringa di caratteri contenuta nell'area da D20 (var_ D20 Array [1]) a D23 (var_ D20 Array [4]) in un valore intero, converte questo valore in valore binario a 16 bit e lo memorizza in D0.



Programma di esempio 2

DVALP

Con il fronte positivo di X0, il programma seguente converte la stringa di caratteri contenuta nell'area da D20 (var_ D20 Array [1]) a D24 (var_ D20 Array [5]) in un valore intero, converte questo valore in valore binario a 32 bit e lo memorizza in D0 e D1.



NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provocano messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.11.11 ESTR, ESTRP

CPU


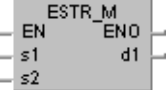

AnS	AnN	AnA, AnAS	AnU	QnA(S), Q4AR	System Q
				●	● ¹

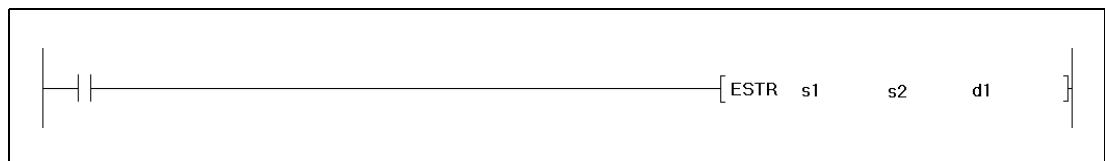
¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi
MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Indice file Zn	Costanti E	Altro		
	Bit	Word		Bit	Word						
s1	—	●	●	—	●	●	—	●	—	SM0	4
s2	—	●	●	—	—	—	—	—	—		
d	—	●	●	—	—	—	—	—	—		

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
 ESTR s1 s2 d1	 ESTR_M EN ENO s1 d1 s2	 ESTR_M s1 , s2 , d1

GX
Developer

Variabili

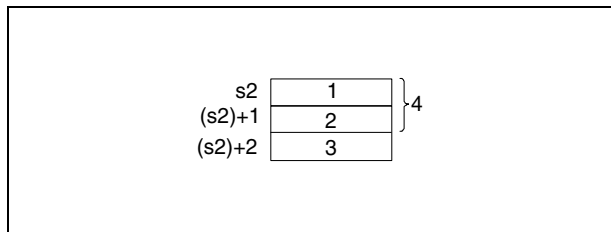
Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s1	Dato in virgola mobile da convertire o indirizzo del primo operando che contiene questo dato.	Numero reale	Numero reale
s2	Indirizzo del primo operando contenente la stringa di caratteri del dato da convertire.	BIN 16-bit	Array [1.0.3] di ANY16
d	Indirizzo del primo operando che memorizza il dato convertito.	Stringa caratteri	Stringa caratteri

Funzioni Conversione di un dato in virgola mobile in stringa di caratteri

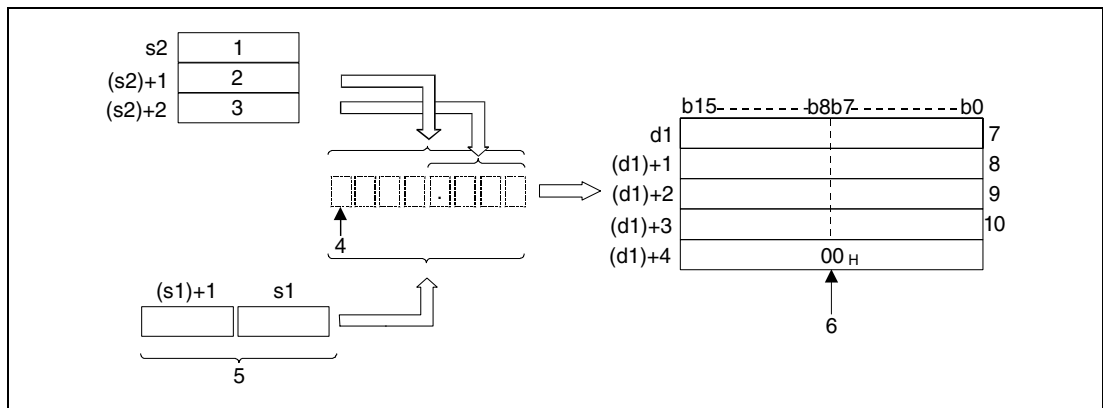
ESTR Conversione di dati in virgola mobile

L'istruzione ESTR converte il dato in virgola mobile (numero reale) indicato da s1 e (s1)+1 in una stringa di caratteri. Il formato della stringa di caratteri viene specificato da s2 (Array_s2[1]) a (s2)+2 (Array_s2[3]). Il risultato è memorizzato a partire da d.

Il formato dei dati dopo la conversione dipende dal formato specificato da s2 (Array_s2[1]) a (s2)+2 (Array_s2[3]).



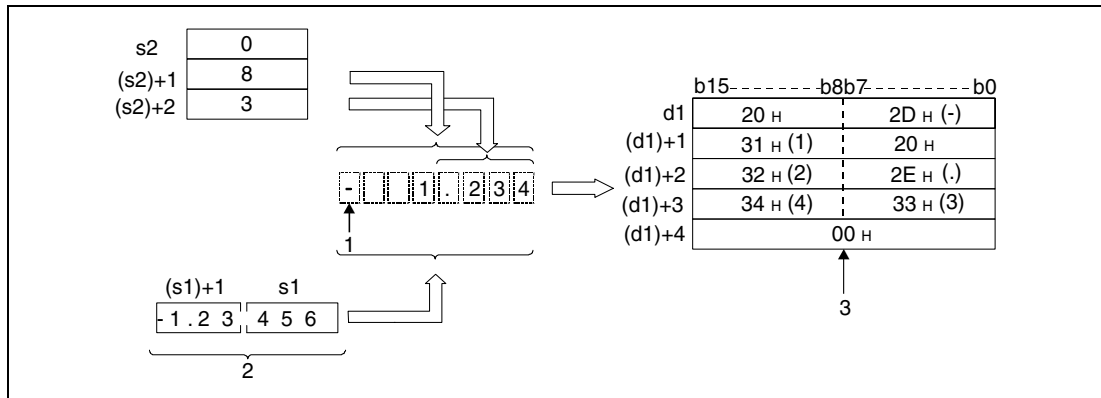
- ¹ Formato dati (formato decimale "0"/ formato esponenziale "1")
- ² Numero delle cifre
- ³ Numero di posti decimali



- ¹ Formato dati (formato decimale "0"/ formato esponenziale "1")
- ² Numero delle cifre
- ³ Numero di posti decimali
- ⁴ Carattere segno
- ⁵ Dato in virgola mobile (numero reale)
- ⁶ Fine stringa, inserito automaticamente.
- ⁷ Posizione carattere in ASCII; numero cifre -1/ codice ASCII del segno
- ⁸ Posizione carattere in ASCII; numero cifre -3/ posizione carattere in ASCII; numero cifre -2
- ⁹ Posizione carattere in ASCII; numero cifre -5/ posizione carattere in ASCII; numero cifre -4
- ¹⁰ Posizione carattere in ASCII; numero cifre -7/ posizione carattere in ASCII; numero cifre -6

Formato decimale

Il numero reale -1,23456 viene convertito in una stringa di caratteri per un totale di 8 cifre (comprese 3 cifre decimali). Il risultato è memorizzato a partire da d.



- ¹ Carattere segno
- ² Dato in virgola mobile (numero reale)
- ³ Fine stringa, inserito automaticamente

Il numero totale delle cifre del numero da convertire, contenuto in (s2)+1 (Array_s2[2]) viene rappresentato come segue:

Se il numero delle cifre decimali è zero, il numero totale delle cifre è ≥ 2 .

Se il numero delle cifre decimali è un valore diverso, il numero totale delle cifre è 3 più il numero delle cifre decimali.

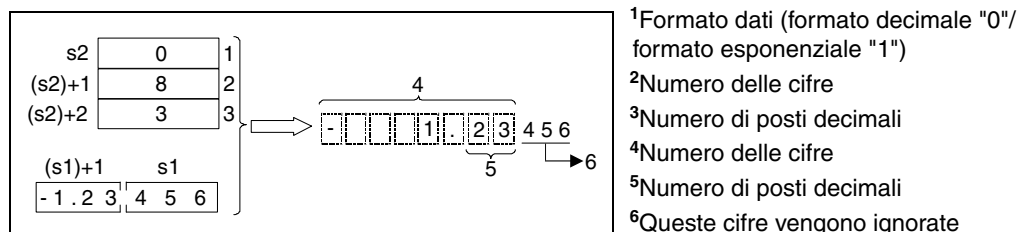
Il numero delle cifre decimali ammesso può variare tra 0 e 7. In generale, il numero delle cifre decimali deve essere inferiore o uguale al numero totale delle cifre meno 3.

Dopo la conversione, la stringa viene memorizzata in d come segue:

Il segno positivo del dato in virgola mobile viene memorizzato con il carattere ASCII "20H" (blank).

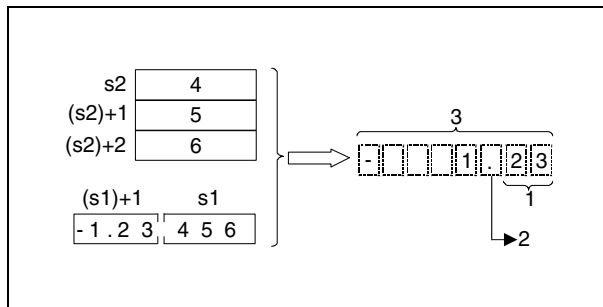
Il segno negativo del dato in virgola mobile viene memorizzato con il carattere ASCII "2DH" (carattere "meno").

Nei casi in cui il numero effettivo delle cifre decimali del dato in virgola mobile supera il numero di cifre decimali specificato, le cifre eccedenti vengono trascurate.

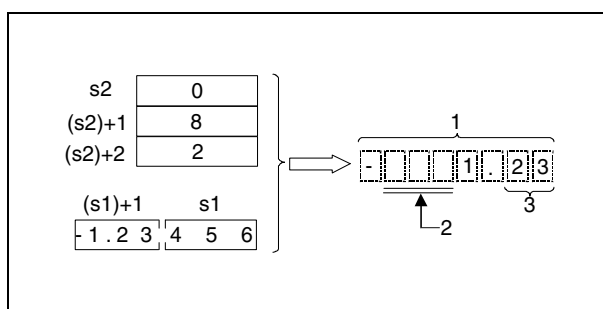


Se il numero di posti decimali è diverso da zero, il punto decimale "2EH" (.) viene inserito automaticamente nella cifra specificata.

Se il numero di posti decimali specificati è zero, il carattere "2DH" (.) non viene inserito.

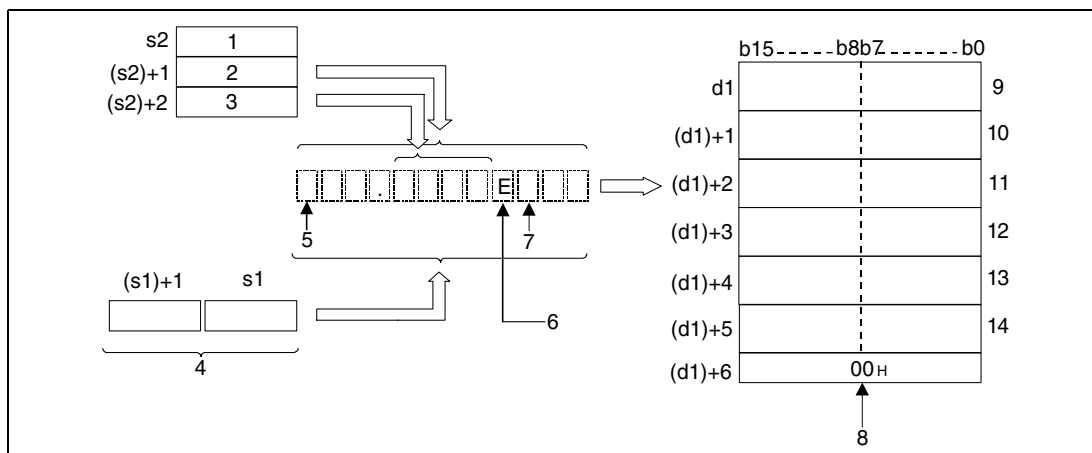


Se il numero totale di tutte le cifre da rappresentare, escluso il segno, è inferiore al numero delle cifre decimali e punto decimale, le cifre fra il segno e la prima cifra significativa sono sostituite dal carattere "20H" (blank).



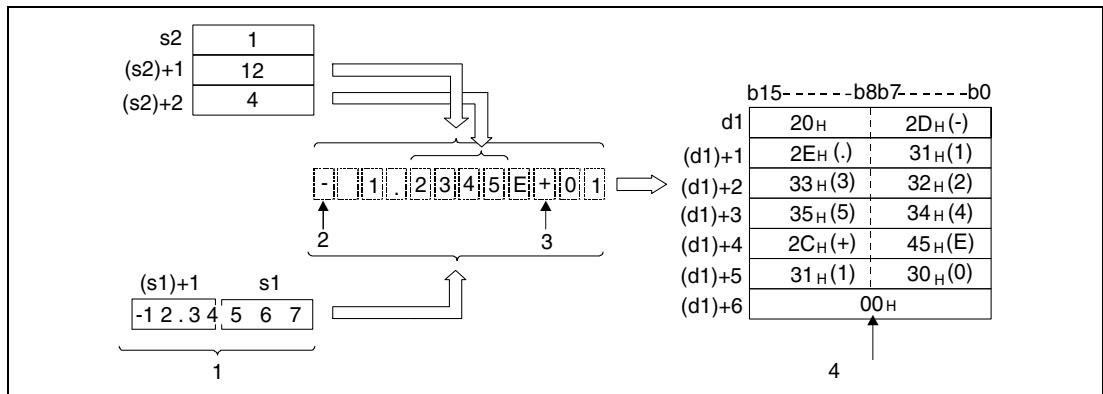
Alla fine della stringa di caratteri, viene inserito automaticamente il codice "00H".

Formato esponenziale



- 1 Formato dati (formato esponenziale (1))
- 2 Numero delle cifre
- 3 Numero di posti decimali
- 4 Numero in virgola mobile (numero reale)
- 5 Segno del valore intero
- 6 Il carattere "E" viene inserito automaticamente
- 7 Segno dell'esponente
- 8 Indicazione di fine stringa, inserita automaticamente.
- 9 Posizione carattere in ASCII; numero cifre -1/ codice ASCII del segno
- 10 Posizione carattere in ASCII; numero cifre -3/ posizione carattere in ASCII; numero cifre -2
- 11 Posizione carattere in ASCII; numero cifre -5/ posizione carattere in ASCII; numero cifre -4
- 12 Posizione carattere in ASCII; numero cifre -7/ posizione carattere in ASCII; numero cifre -6
- 13 Segno dell'esponente/ 45H (E)
- 14 Posizione carattere in ASCII; numero cifre -11 (esponente)/
posizione carattere in ASCII; numero cifre -10 (esponente)

Si deve rappresentare il numero reale -12,34567 in notazione esponenziale. Il numero totale delle cifre è 12. Il numero delle cifre decimali è 4. Il risultato viene memorizzato a partire da d.



- 1 Numero in virgola mobile (numero reale)
- 2 Segno del valore intero
- 3 Segno dell'esponente
- 4 Indicazione di fine stringa, inserita automaticamente.

Il numero totale delle cifre del numero da convertire, contenuto in (s2)+1 (Array_s2[2]) viene rappresentato come segue:

Se il numero delle cifre decimali è zero, il numero totale delle cifre è >= 2.

Se il numero delle cifre decimali è un valore diverso, il numero totale delle cifre è 7 più il numero delle cifre decimali.

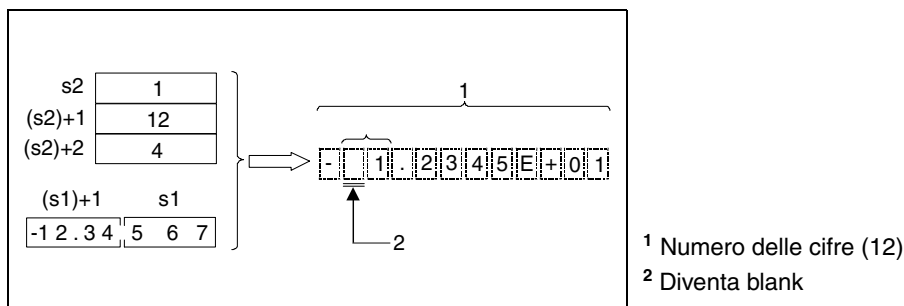
Il numero delle cifre decimali ammesso può variare tra 0 e 7. In generale, il numero delle cifre decimali deve essere inferiore o uguale al numero totale delle cifre meno 7.

Dopo la conversione, la stringa viene memorizzata in d come segue:

Il segno positivo del dato in virgola mobile viene memorizzato con il carattere ASCII "20H" (blank).

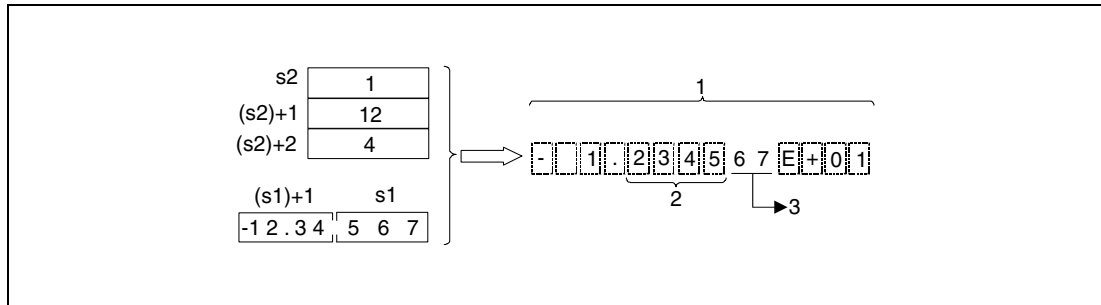
Il segno negativo del dato in virgola mobile viene memorizzato con il carattere ASCII "2DH" (carattere "meno").

Il campo intero è composto da due cifre. Se il campo intero è composto da una sola cifra, viene inserito un carattere ASCII blank fra il carattere del segno e la cifra intera.



- 1 Numero delle cifre (12)
- 2 Diventa blank

Se il valore in virgola mobile del campo decimale è più lungo del campo di memorizzazione relativo, le cifre che non possono essere memorizzate vengono troncate.

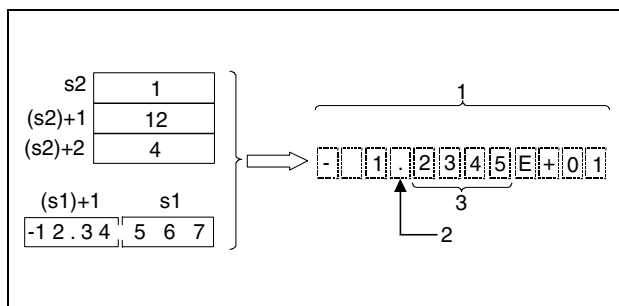


¹ Numero delle cifre (12)

² Numero delle cifre del campo decimale (4)

³ Queste cifre vengono ignorate

Se il numero di posti decimali è diverso da zero, il punto decimale "2EH" (.) viene inserito automaticamente nella cifra specificata.



¹ Numero delle cifre (12)

² Inserito automaticamente

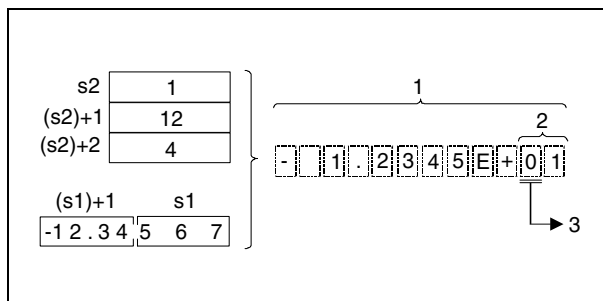
³ Numero delle cifre del campo decimale (4)

Se il numero di posti decimali specificati è zero, il carattere "2DH" (.) non viene inserito.

Con esponente positivo, viene inserito e memorizzato il codice ASCII "2CH" (+).

Con esponente negativo, viene inserito e memorizzato il codice ASCII "2DH" (-).

Il campo esponente è composto da due cifre. Se il campo esponente contiene una sola cifra, viene inserito il codice ASCII "30H" (0) fra il segno dell'esponente e l'esponente.



¹ Numero delle cifre (12)

² Fisso a due cifre

³ Impostato a zero automaticamente

Alla fine della stringa di caratteri, viene inserito automaticamente il codice "00H".

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Gli operandi specificati in s1 e (s1)+1 non sono zero o superano il campo di valori $\pm 2^{-127} \leq s1 < \pm 2^{129}$ (codice di errore 4100).
- Il formato in s2 (Array_s2[1]) è diverso da 0 o 1 (codice di errore 4100).
- Il numero di cifre in (s2)+1 (Array_s2[2]) supera il campo di valori (codice di errore 4100):

Per il formato decimale

Il numero delle cifre decimali è zero (numero totale delle cifre ≥ 2).
 Numero delle cifre decimali diverso da zero (numero delle cifre \geq (numero cifre decimali + 3)).

Per il formato esponenziale

Il numero delle cifre decimali è zero (numero totale delle cifre ≥ 2).
 Numero delle cifre decimali diverso da zero (numero delle cifre \geq (numero cifre decimali + 7)).

- Il numero di cifre in (s2)+2 (Array_s2[3]), che costituisce la parte decimale, supera il campo di valori (codice di errore 4100):

Per il formato decimale

Il numero delle cifre che forma la parte decimale è inferiore o uguale al numero totale di cifre meno 3.

Per il formato esponenziale

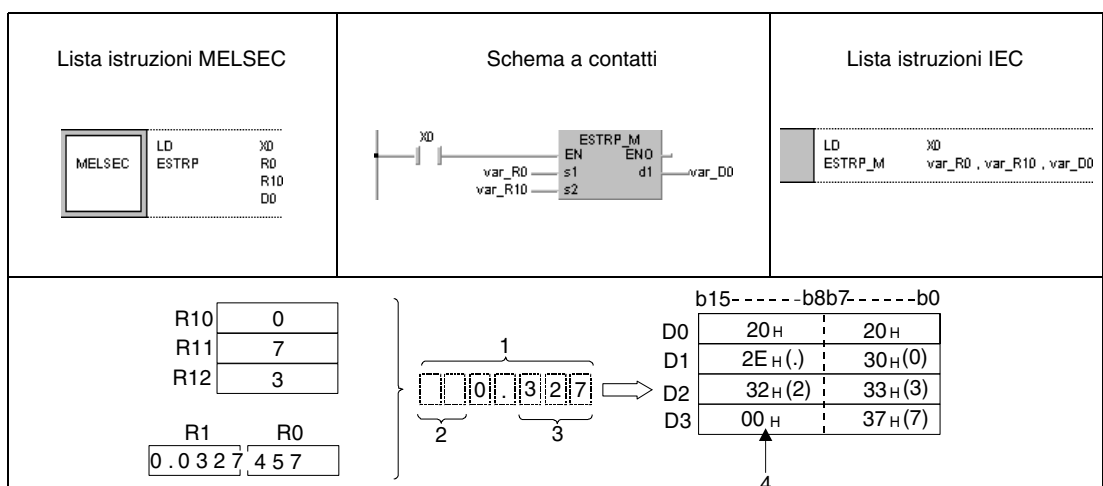
Il numero delle cifre che forma la parte decimale è inferiore o uguale al numero totale di cifre meno 7.

- Il campo di memorizzazione in s supera il campo operando ammesso (codice di errore 4101).

Programma di esempio 1

ESTRP

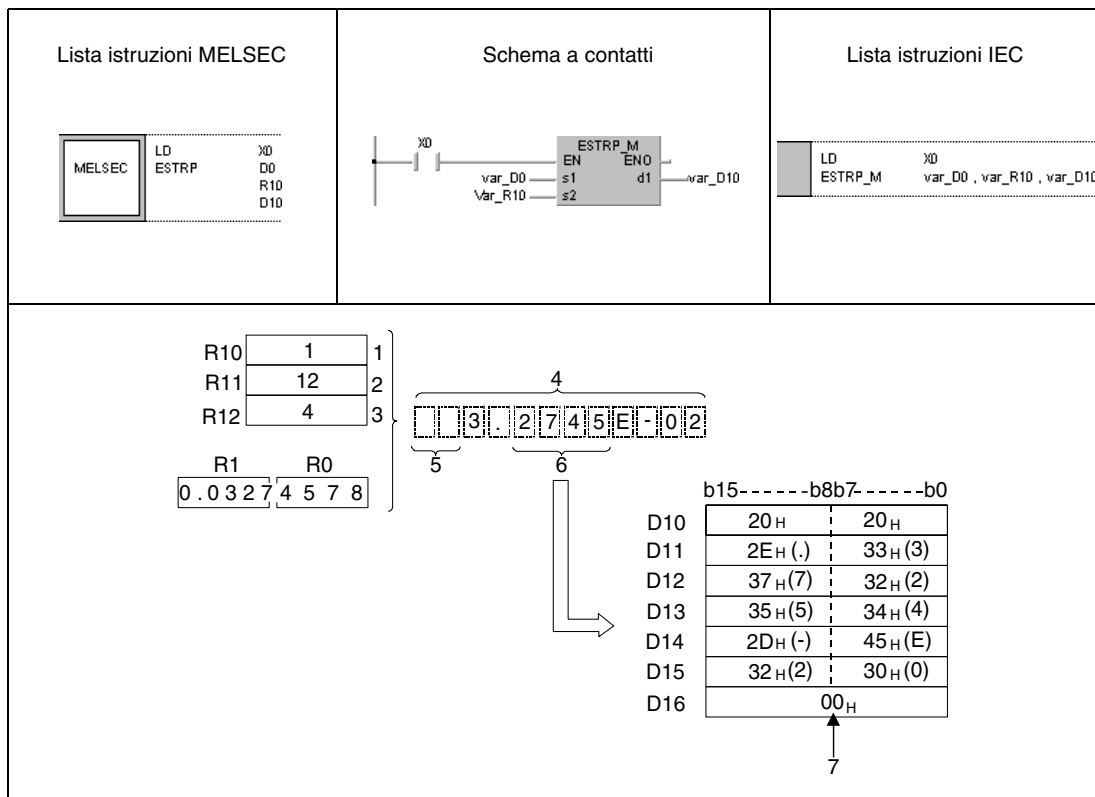
Con il fronte positivo di X0, il programma che segue converte il valore in virgola mobile (numero reale) contenuto negli operandi R0 e R1 nel formato specificato da R10 (var_R10 Array [1]) a R12 (var_R10 Array [3]) e memorizza il risultato da D0 a D3.



- 1 Numero delle cifre
- 2 Blank
- 3 Numero di posti decimali
- 4 Inserito automaticamente

Programma di esempio 2 ESTRP

Con il fronte positivo di X0, il programma che segue converte il valore in virgola mobile (numero reale) contenuto negli operandi D0 e D1 nel formato specificato da R10 (var_R10 Array [1]) a R12 (var_R10 Array [3]) e memorizza il risultato da D10 a D16.



- 1 Formato dati (rappresentazione esponenziale) (1)
- 2 Numero delle cifre
- 3 Numero di posti decimali
- 4 Numero delle cifre
- 5 Blank
- 6 Numero decimali nella parte decimale
- 7 Inserito automaticamente

7.11.12 EVAL, EVALP

CPU

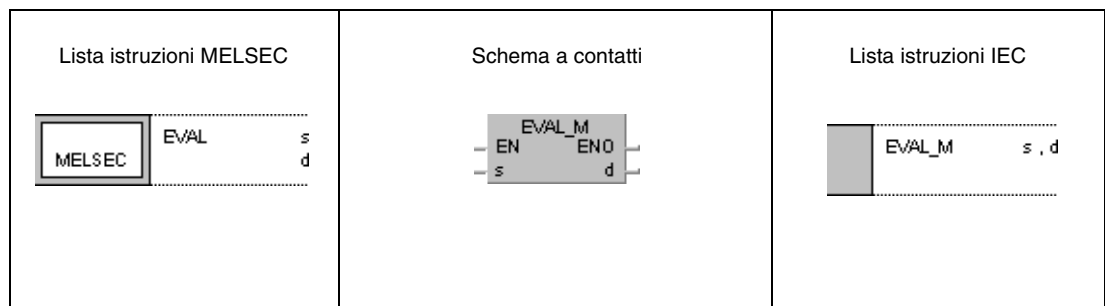
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

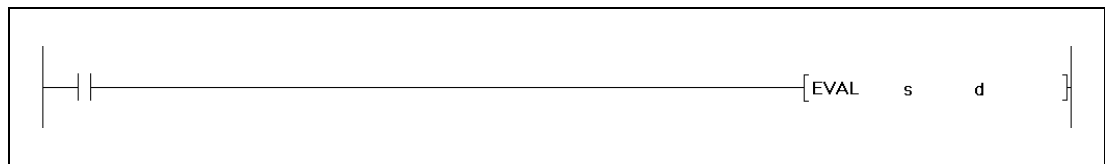
Operandi MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Indice file Zn	Costanti \$	Altro		
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	●	—	SM0	3
d	—	●	●	—	●	●	—	—	—		

GX IEC Developer



GX Developer



Variabili

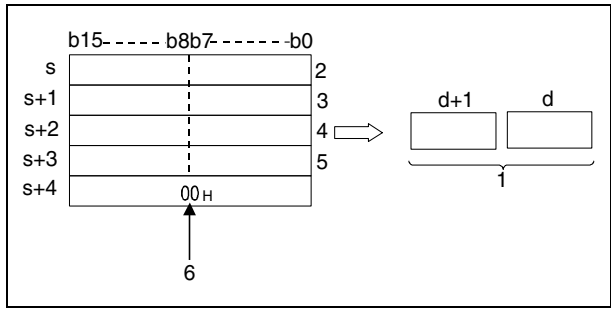
Dati impostati	Significato	Tipo dati
s	Stringa di caratteri da convertire in numero in virgola mobile (numero reale) o indirizzo del primo operando che contiene questo dato.	Stringa caratteri
d	Indirizzo del primo operando che memorizza il numero in virgola mobile (numero reale) convertito.	Numero reale

Funzioni Conversione di una stringa di caratteri in un dato in virgola mobile

EVAL Conversione di stringhe di caratteri

L'istruzione EVAL converte la stringa di caratteri indicata da s a s+4 in un numero decimale in virgola mobile (numero reale). Il risultato è memorizzato in d.

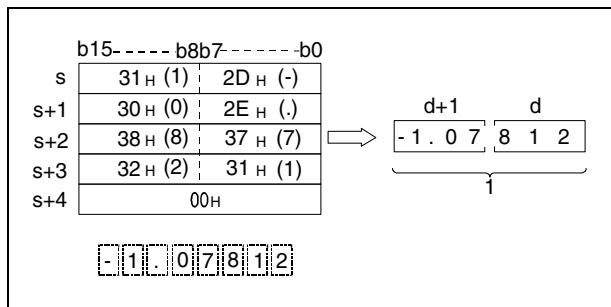
La stringa di caratteri può essere convertita nei formati virgola mobile decimale o esponenziale.



- ¹ Dato in virgola mobile decimale (numero reale)
- ² Codice ASCII del primo carattere/ codice ASCII del segno
- ³ Codice ASCII del terzo carattere/ Codice ASCII del secondo carattere/
- ⁴ Codice ASCII del quinto carattere/ Codice ASCII del quarto carattere/
- ⁵ Codice ASCII del settimo carattere/ Codice ASCII del sesto carattere

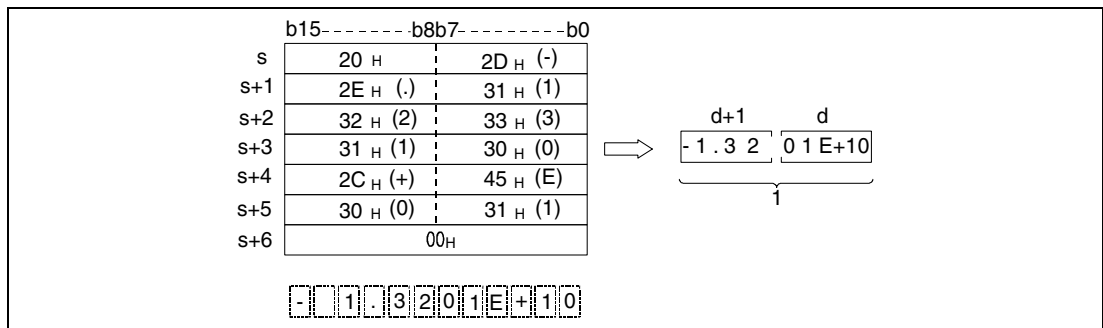
⁶ Indica la fine della stringa

Formato decimale



- ¹ Dato in virgola mobile decimale (numero reale)

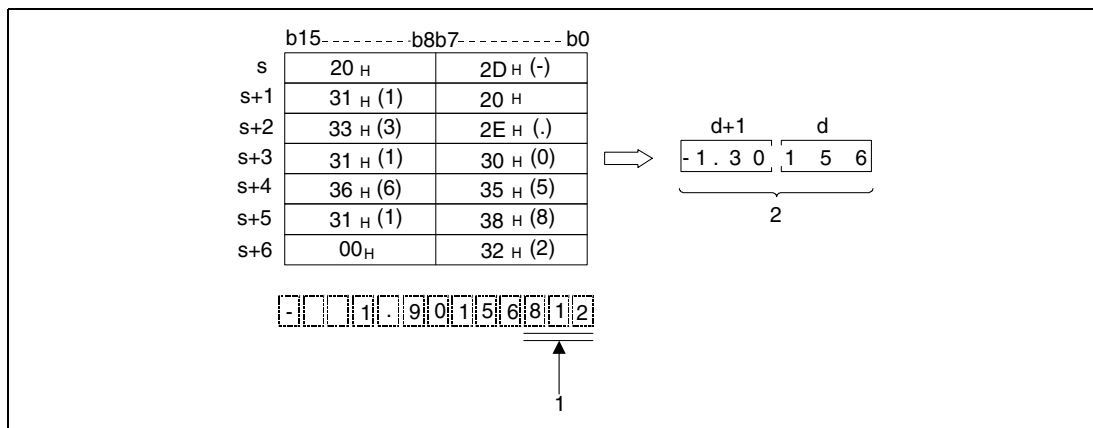
Formato esponenziale



- ¹ Dato in virgola mobile decimale (numero reale)

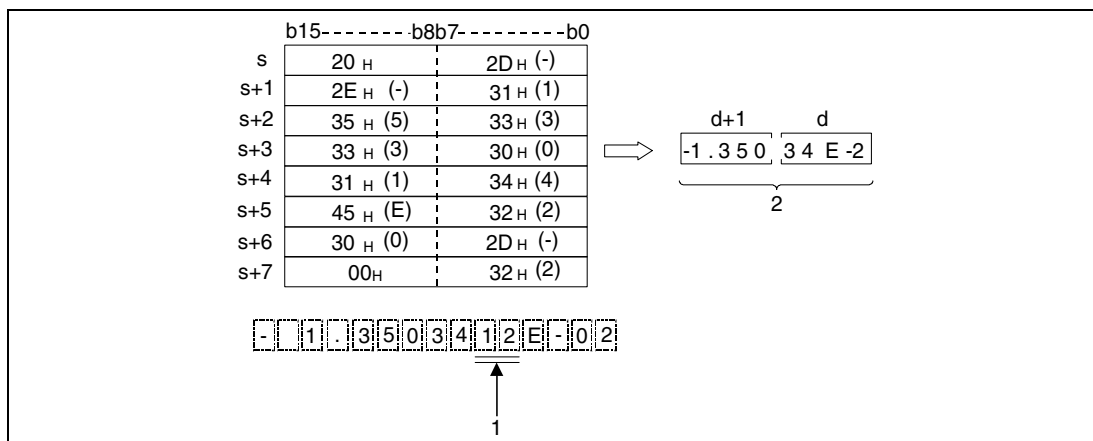
Nell'esempio che segue, sei cifre (escluso segno, punto decimale e cifre esponente del risultato) della stringa di caratteri a partire da s vengono convertite in un numero in virgola mobile decimale. Le cifre a partire dalla settima sono escluse dal risultato.

Formato decimale



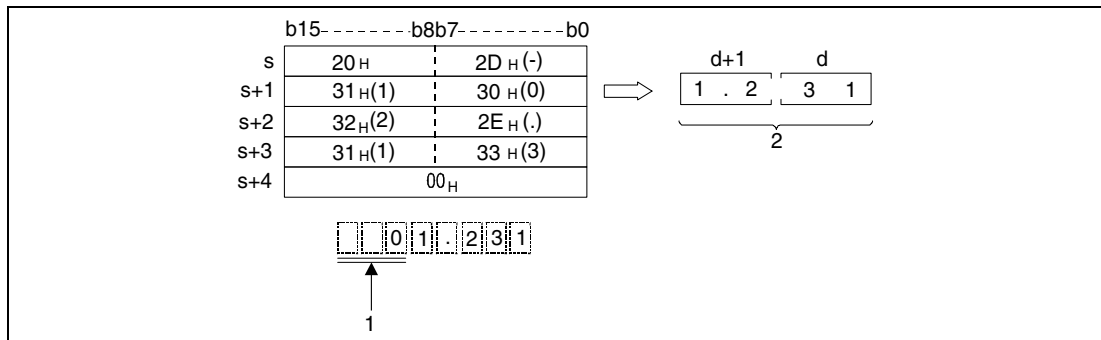
- ¹ Queste cifre non vengono considerate
- ² Dato in virgola mobile decimale (numero reale)

Formato esponenziale



- ¹ Queste cifre non vengono considerate
- ² Dato in virgola mobile decimale (numero reale)

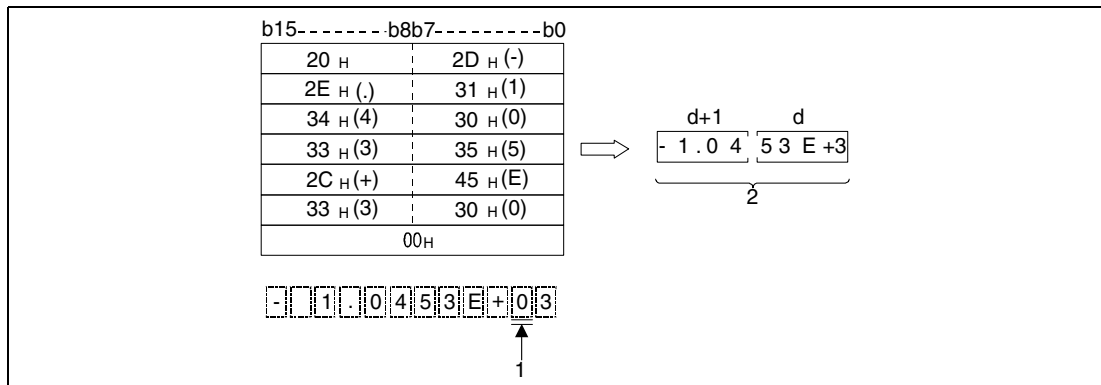
Blank (codice ASCII "20H") o zeri (codice ASCII "30H") iniziali nella stringa di caratteri a partire da s vengono ignorati nella conversione, tranne lo zero iniziale (ad es. 0.123).



¹ Questi caratteri vengono ignorati dalla conversione

² Dato in virgola mobile decimale (numero reale)

Se il codice ASCII "30H" (zero) è inserito fra il carattere "E" e la stringa di caratteri del formato esponenziale, questo carattere viene ignorato dalla conversione.



¹ Questi caratteri vengono ignorati dalla conversione

² Dato in virgola mobile decimale (numero reale)

Una stringa di caratteri da convertire può contenere al massimo 24 caratteri.

Errori di esecuzione

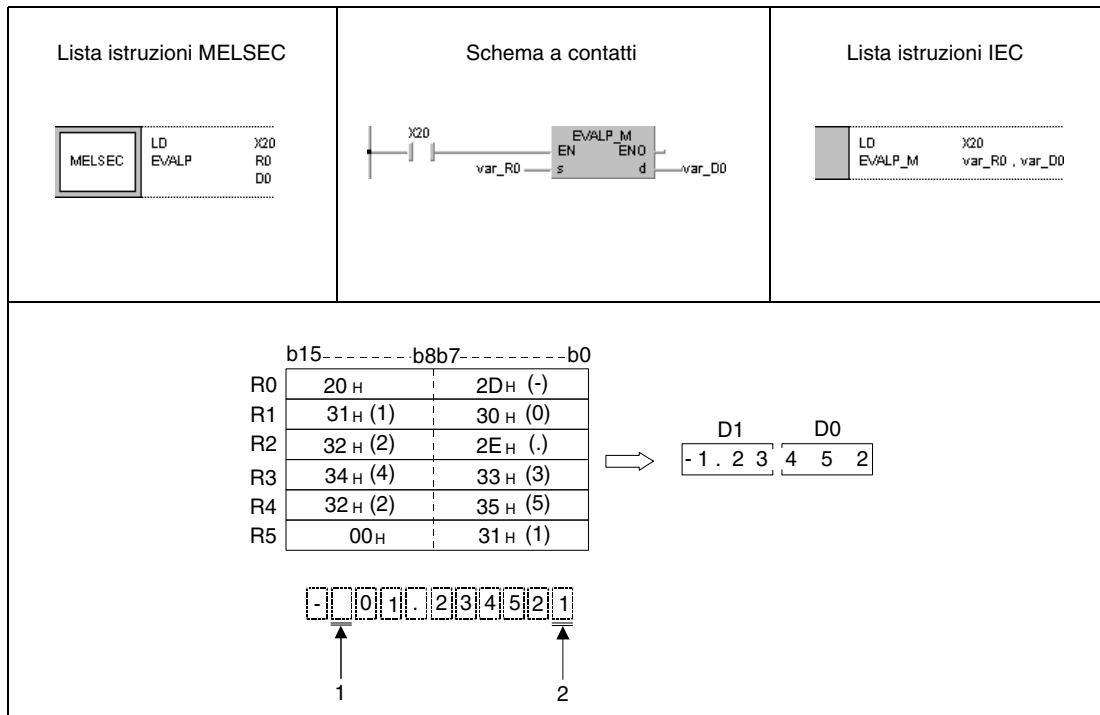
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- La stringa non inizia con il carattere "20H" (blank) o "2DH" (meno) (codice di errore 4100).
- Le cifre prima del punto decimale, o le cifre decimali, contengono caratteri con codici fuori dal campo da "30H" (0) a "39H" (9) (codice di errore 4100).
- Il carattere "2EH" compare più di una volta nella stringa (codice di errore 4100).
- La parte esponente contiene caratteri diversi da "45H (E), 2CH (+)" o "45H (E), 2DH (-)". Più di un esponente (codice di errore 4100).
- Il valore è 0 o supera il campo di valori ammessi da 1.0×2^{-127} a 1.0×2^{129} (codice di errore 4100).
- Il carattere di fine stringa "00H" si trova oltre il campo operando ammesso (error code 4100).
- Il numero di caratteri della stringa è 0 o superiore a 24.

Programma di esempio 1

EVALP

Con il fronte positivo di X20, il programma seguente converte la stringa di caratteri contenuta da R0 a R5 in un numero in virgola mobile decimale (numero reale) e memorizza il risultato in D0 e D1.

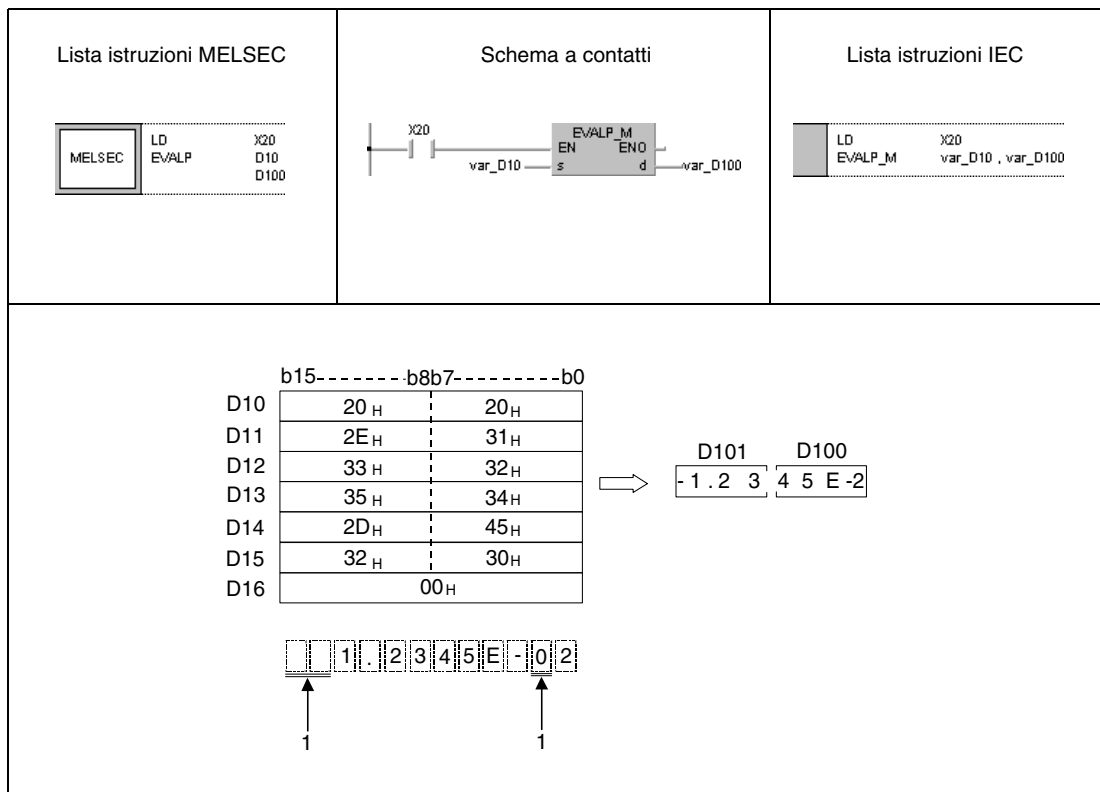


¹ Questa cifra non viene elaborata

² Questo numero viene troncato

Programma di esempio 2 EVALP

Con il fronte positivo di X20, il programma seguente converte la stringa di caratteri contenuta da D10 a D16 in un numero in virgola mobile (numero reale) e memorizza il risultato in D100 e D101.



¹ Queste cifre non vengono elaborate

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provocano messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.11.13 ASC, ASCP (serie Q e System Q)

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

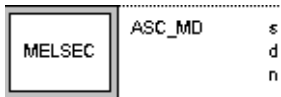


¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Indice file Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	—	SM0	4	
d	—	●	●	—	—	—	—	—			
n	●	●	●	●	●	●	●	—			

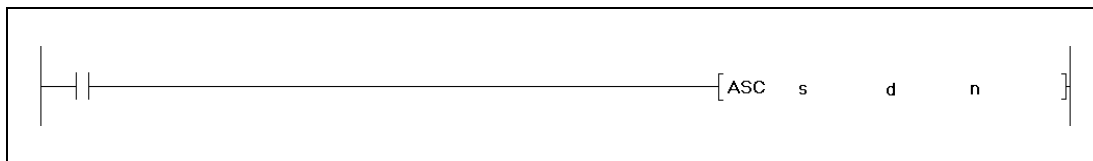
GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
--	--	---

Le istruzioni ASC e ASCP non possono essere usate con gli editor IEC. Queste istruzioni possono essere programmate solo con la lista istruzioni MELSEC.

Rimedio: Inserire il formato ASCII esadecimale direttamente nei registri target.

**GX
Developer**



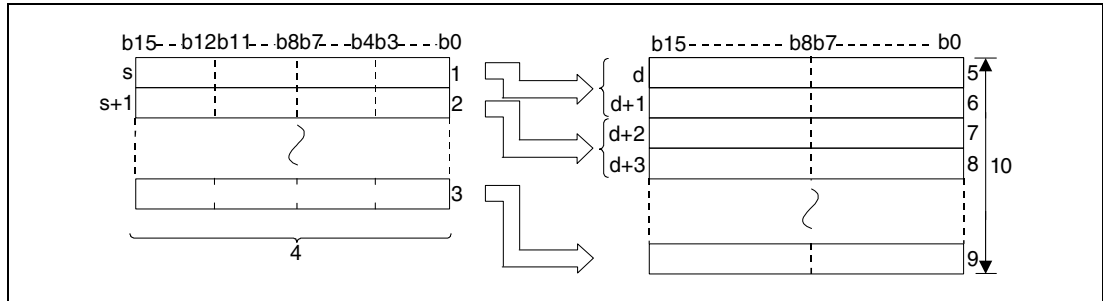
Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando contenente la stringa di caratteri da convertire in formato binario.	Stringa caratteri
d	Indirizzo del primo operando che memorizza il dato binario convertito.	BIN 16-bit
n	Numero di caratteri da convertire.	

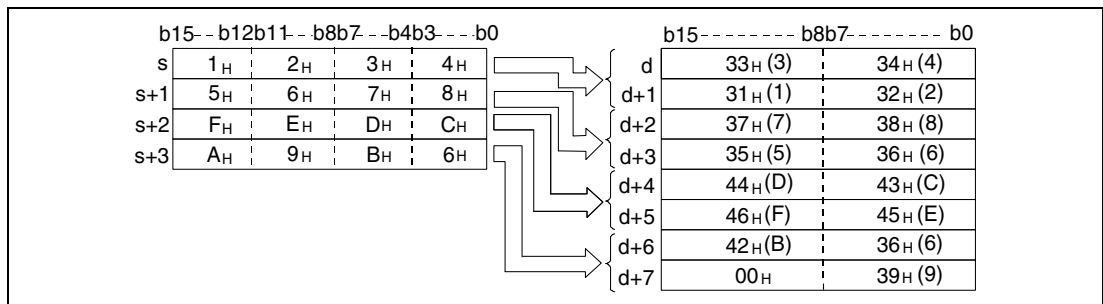
Funzioni Conversione di un dato BIN a 16 bit in codice ASCII

ASC/ASCP Istruzione di conversione

L'istruzione ASCII converte i dati a 16 bit memorizzati a partire da s in formato ASCII esadecimale, memorizzando il numero di caratteri specificati da n a partire da d.

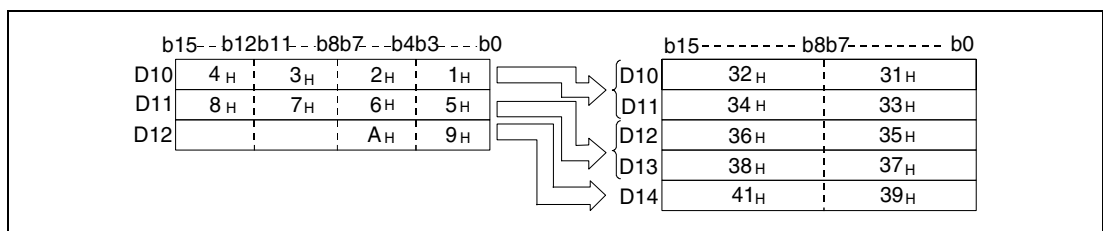


- ¹ Prima cifra/ seconda cifra/ terza cifra/ quarta cifra
- ² Prima cifra/ seconda cifra/ terza cifra/ quarta cifra
- ³ Prima cifra/ seconda cifra/ terza cifra/ quarta cifra
- ⁴ Dato binario
- ⁵ Codice ASCII della prima cifra/ codice ASCII della seconda cifra
- ⁶ Codice ASCII della terza cifra/ codice ASCII della quarta cifra
- ⁷ Codice ASCII della quinta cifra/ codice ASCII della sesta cifra
- ⁸ Codice ASCII della settima cifra/ codice ASCII della ottava cifra
- ⁹ Codice ASCII della nona cifra/ codice ASCII della decima cifra
- ¹⁰ Numero di cifre indicato da n

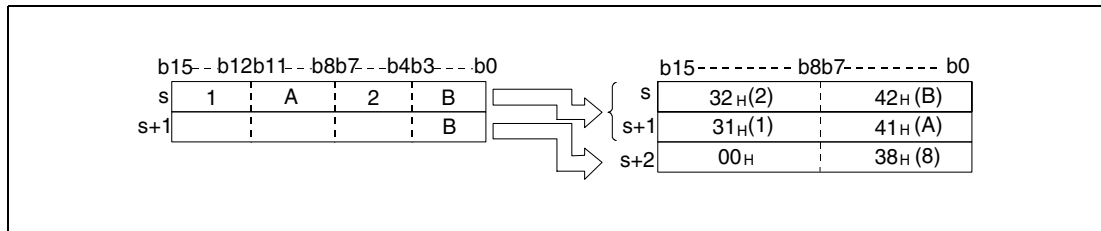


Il numero di caratteri specificato da n determina il campo di valori per gli operandi contenuti a partire da s. Gli operandi specificati a partire da s contengono i dati binari da convertire. La stringa di caratteri risultante viene memorizzata negli operandi a partire da d.

Il programma viene eseguito correttamente e senza segnalazione di errore, anche se l'area contenente i dati da convertire si sovrappone all'area per i dati ASCII convertiti.



Se n indica un numero di caratteri dispari, il carattere ASCII "00H" viene inserito automaticamente negli 8 bit più pesanti dell'ultimo indirizzo specificato per la memorizzazione della stringa.



Se il numero di caratteri indicato da n è zero, il programma non viene eseguito.

Errori di esecuzione

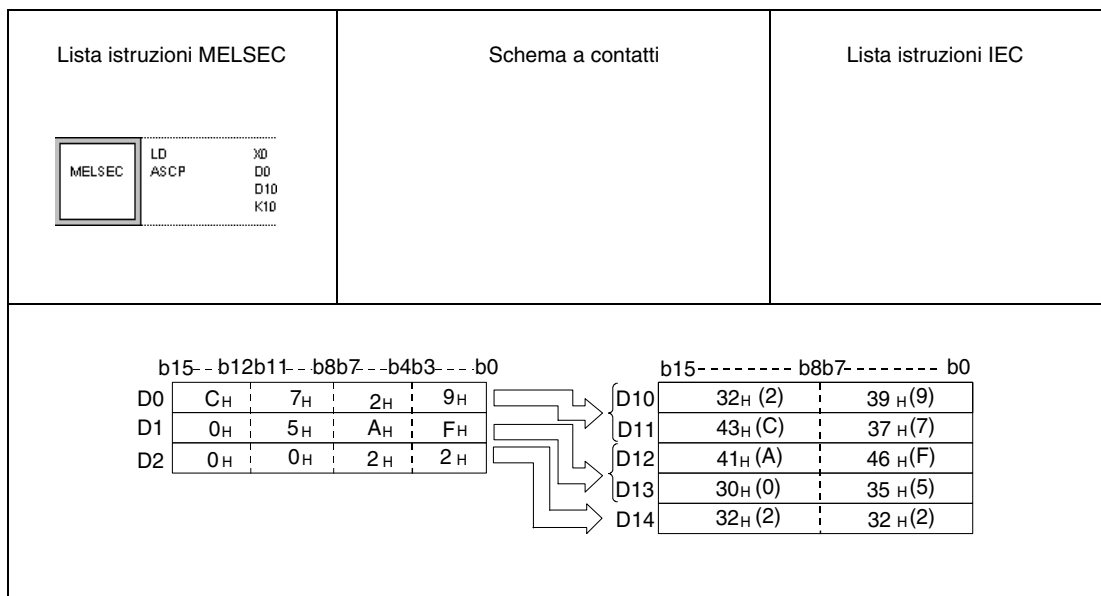
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il numero di caratteri specificato da n , e quindi il numero di registri necessari a partire da s , supera il campo operando ammesso (codice di errore 4101).
- Il numero di caratteri specificato da n , e quindi il numero di registri necessari a partire da d , supera il campo operando ammesso (codice di errore 4101).

Esempio di programma

ASCP

Con il fronte positivo di X0, il programma seguente legge il dato binario contenuto in D0 come valore esadecimale, e lo converte in una stringa di caratteri. Il risultato viene memorizzato da D10 a D14.



7.11.14 ASC (serie A)

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●		

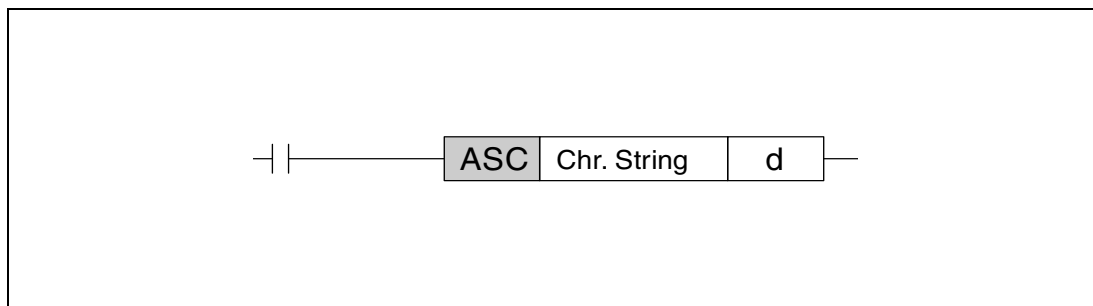
Operandi MELSEC A

	Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Riporto		Errore							
	Operandi a bit						Operandi a word (16-bit)						Costanti		Puntatore					Livello		M9012	M9010 M9011						
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V				K	H (16#)	P	I	N					
d							●	●	●	●	●														13 ¹	●			●

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

GX IEC Developer

GX Developer



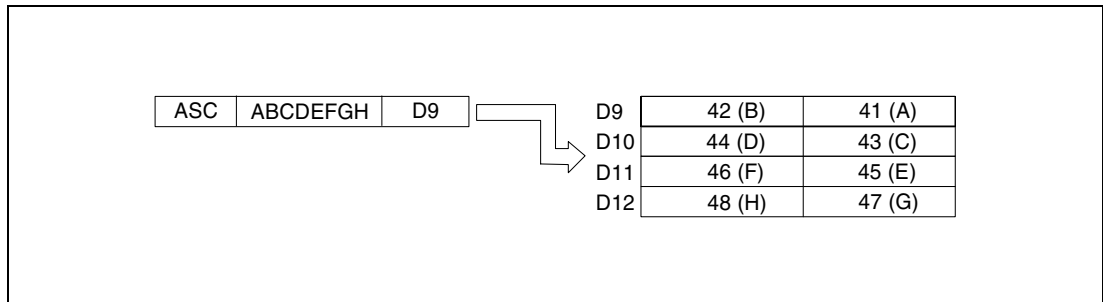
Variabili

Operando	Significato	Tipo dati
d	Operando che memorizza i caratteri convertiti.	BIN 16-bit

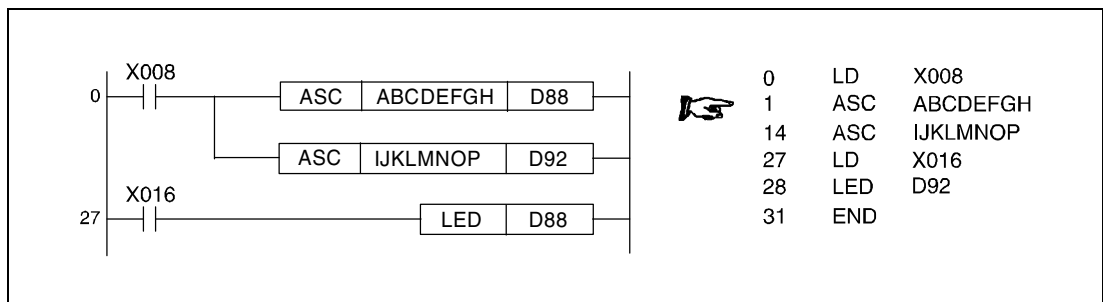
Funzioni Conversione di stringa di caratteri in codice ASCII**ASC Conversione di stringhe di caratteri alfanumeriche**

L'istruzione ASC converte stringhe di caratteri alfanumerici lunghe fino a 8 caratteri in codice ASCII. Il risultato è memorizzato a partire da d.

Il codice ASCII memorizzato può essere stampato con le istruzioni PR/PRC e visualizzato sul display a LED della CPU (se applicabile) con l'istruzione LED.

**Esempio di programma ASCP**

Dopo l'attivazione di X8, il programma seguente converte la stringa di caratteri "ABCDEFGHIJKLMNOP" in codice ASCII e memorizza il risultato da D88 a D91 e da D92 a D95. Dopo l'attivazione di X16, i dati ASCII da D88 a D95 vengono visualizzati sui LED del pannello frontale della CPU.



7.11.15 HEX, HEXP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	●

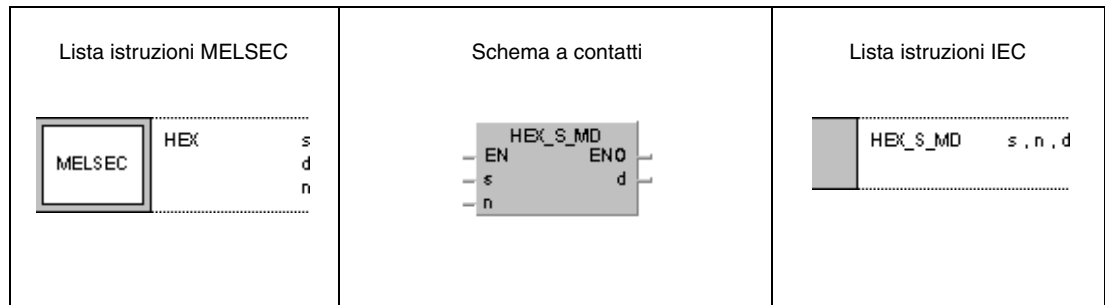
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

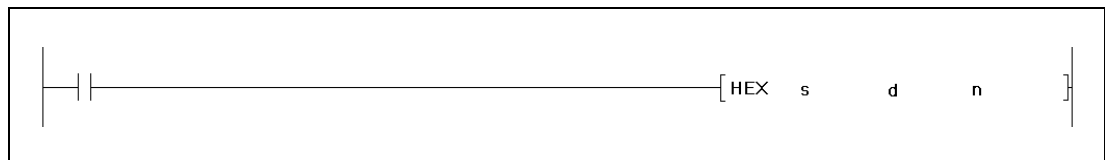
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Indice file Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	—	SM0	4	
d	—	●	●	—	—	—	—	—			
n	●	●	●	●	●	●	●	—			

GX IEC Developer



GX Developer

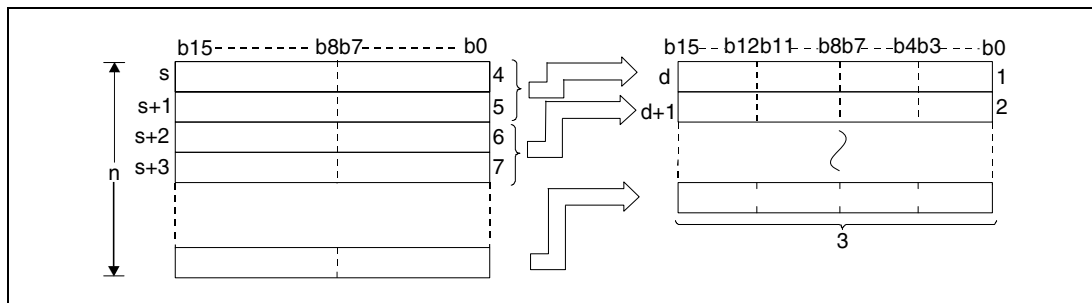


Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene il dato binario da convertire	Stringa caratteri
d	Indirizzo del primo operando che memorizza il dato binario convertito.	BIN 16-bit
n	Numero di caratteri da convertire.	

Funzioni Conversione di valori ASCII esadecimale in valori binari**HEX Conversione di valori ASCII esadecimale**

L'istruzione HEX converte i caratteri ASCII esadecimale a partire da s in valori binari. Il risultato è memorizzato a partire da d.



¹ Quarta cifra, terza cifra, seconda cifra, prima cifra

² Dato binario

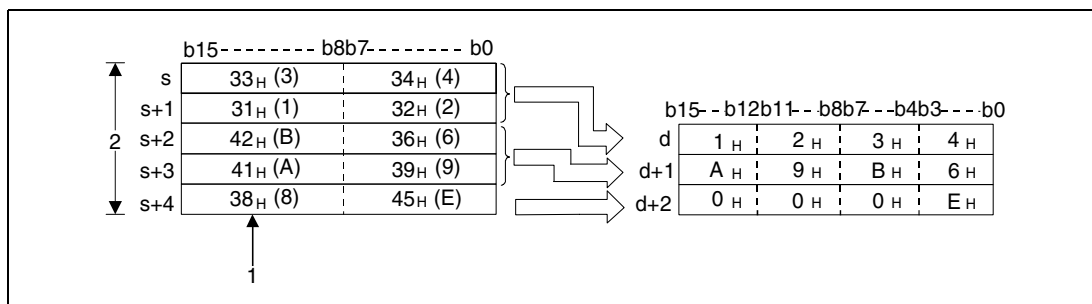
³ Codice ASCII della seconda cifra/ codice ASCII della prima cifra

⁴ Codice ASCII della quarta cifra/ codice ASCII della terza cifra

⁵ Codice ASCII della seconda cifra/ codice ASCII della prima cifra

⁶ Codice ASCII della quarta cifra/ codice ASCII della terza cifra

Il numero di caratteri in n è 9.

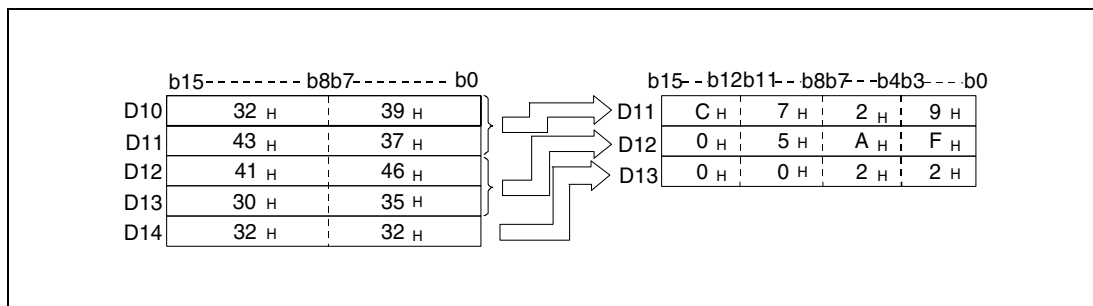


¹ Dato che la stringa contiene 9 caratteri, il carattere "38_H" non viene modificato o spostato.

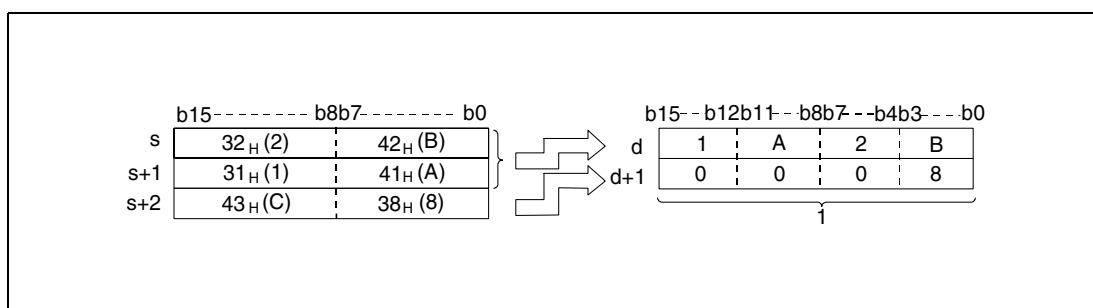
² n = 9

Il numero di caratteri indicato da n determina automaticamente il campo di valori della stringa di caratteri s e dei dati binari a partire da d.

Anche se il campo di valori del codice ASCII da convertire e quello dei valori binari convertiti si sovrappongono, questa istruzione elabora correttamente i dati.



Se il numero di caratteri in 4 non è divisibile per 4, viene automaticamente inserito uno zero dopo il numero di caratteri specificato, nel registro di indirizzo più alto che contiene il valore binario convertito.



¹ Il valore zero viene inserito automaticamente

Se il numero di caratteri indicato da n è zero, la conversione non viene eseguita.

Il codice ASCII contenuto a partire da s può variare da "30H" a "39H" e da "41H" a "46H".

Errori di esecuzione

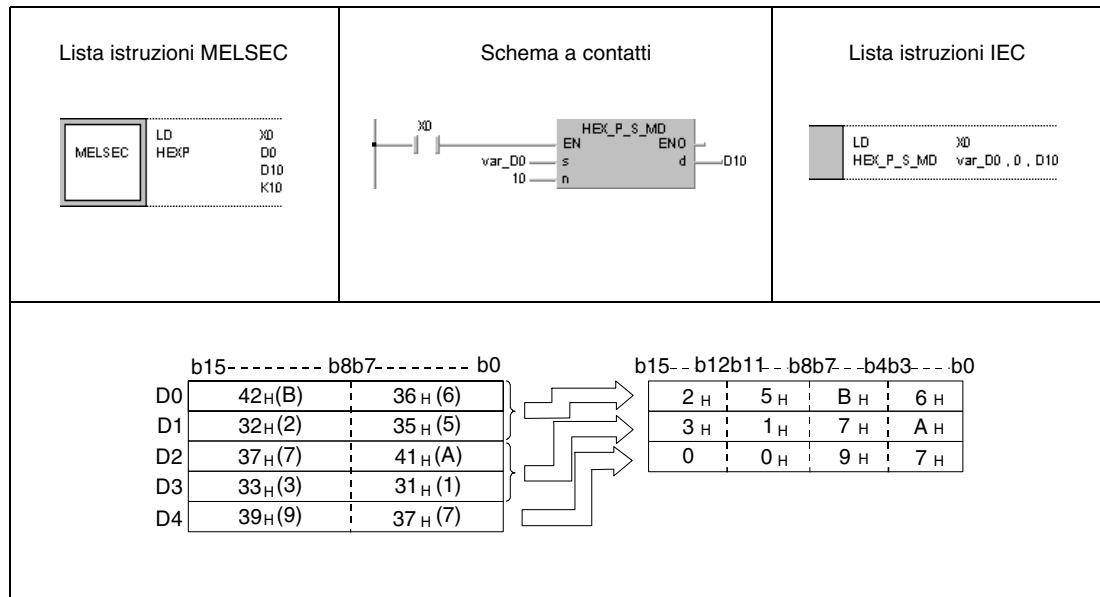
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il codice ASCII contenuto a partire da s non è compreso nel campo da "30H" a "39H" e da "41H" a "46H" (codice di errore 4100).
- Il numero di caratteri specificato da n, e quindi il numero di registri necessari a partire da s, supera il campo operando ammesso (codice di errore 4101).
- Il numero di caratteri specificato da n, e quindi il numero di registri necessari a partire da d, supera il campo operando ammesso (codice di errore 4101).
- Il valore di n è negativo.

Esempio di programma

HEXP

Con il fronte positivo di X0, il programma seguente converte la stringa di caratteri "6B52A71379" contenuta da D0 a D4 in un dato binario. Il risultato viene memorizzato da D10 a D14.

**NOTA**

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.11.16 RIGHT, RIGHTP, LEFT, LEFTP

CPU

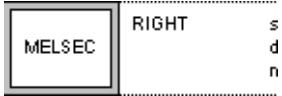
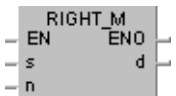
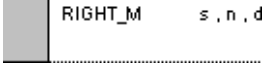
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

	Operandi utilizzabili										Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Indice file Zn	Costanti		Altro		
	Bit	Word		Bit	Word			K, H (16#)	\$			
s	—	●	●	—	—	—	—	—	●	—	SMO	4
d	—	●	●	—	—	—	—	—	—	—		
n	●	●	●	●	●	●	●	●	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
---	---	---

GX Developer

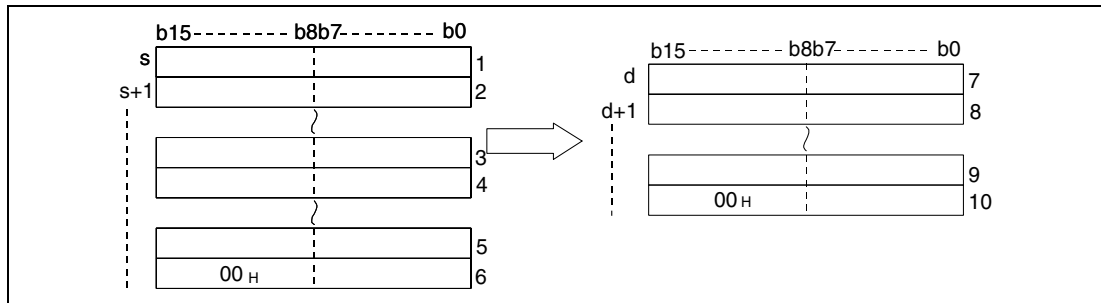


Variabili

Operando	Significato	Tipo dati
s	Indirizzo del primo operando che memorizza la stringa di caratteri.	Stringa caratteri
d	Indirizzo del primo operando che memorizza i caratteri specificati della stringa originale.	
n	Numero di caratteri memorizzati della parte sinistra o destra.	BIN 16-bit

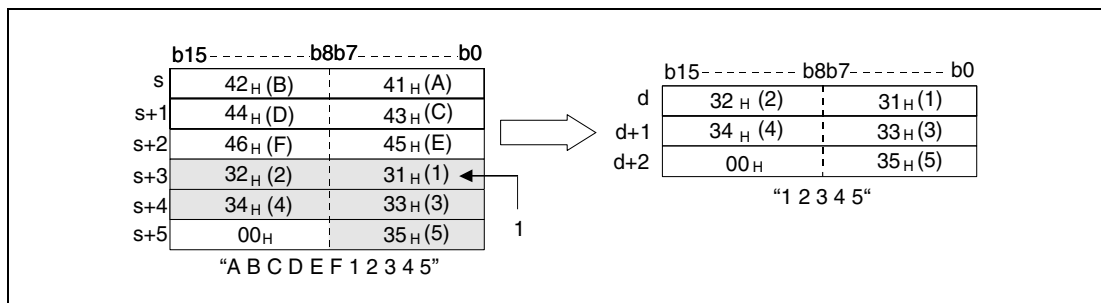
Funzioni Estrazione di dati dalla parte sinistra o destra di una stringa di caratteri**RIGHT Estrae caratteri dalla parte destra della stringa**

L'istruzione RIGHT memorizza n caratteri del lato destro (parte finale) della stringa di caratteri contenuta a partire da s. I caratteri sono memorizzati a partire da d.



- ¹ Codice ASCII del secondo carattere/ codice ASCII del primo carattere
- ² Codice ASCII del quarto carattere/ codice ASCII del terzo carattere
- ³ Codice ASCII dell'ultimo carattere meno n+2/ codice ASCII dell'ultimo carattere meno n+1
- ⁴ Codice ASCII dell'ultimo carattere meno n+4/ codice ASCII dell'ultimo carattere meno n+3
- ⁵ Codice ASCII dell'ultimo carattere meno 1/ codice ASCII dell'ultimo carattere meno 2
- ⁶ "00H"/ codice ASCII dell'ultimo carattere
- ⁷ Codice ASCII dell'ultimo carattere meno n+2/ codice ASCII dell'ultimo carattere meno n+1
- ⁸ Codice ASCII dell'ultimo carattere meno n+4/ codice ASCII dell'ultimo carattere meno n+3
- ⁹ Codice ASCII dell'ultimo carattere meno 1/ codice ASCII dell'ultimo carattere meno 2
- ¹⁰ "00H"/ codice ASCII dell'ultimo carattere

Con n = 5

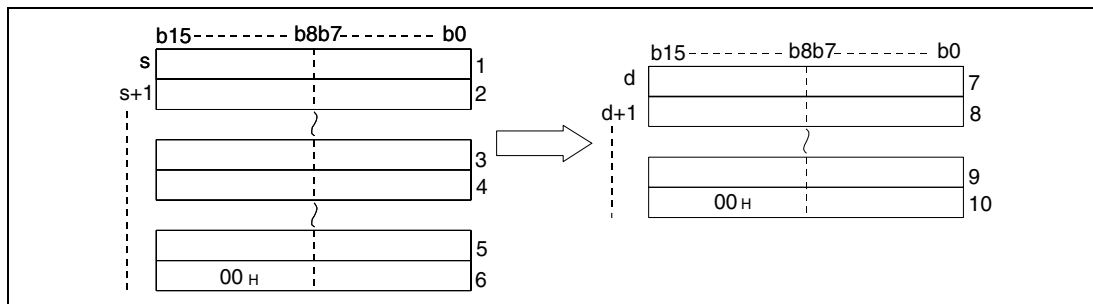


- ¹ Codice ASCII del quinto carattere

Se il numero di caratteri in n è zero, il codice di carattere "00H" viene memorizzato a partire da d.

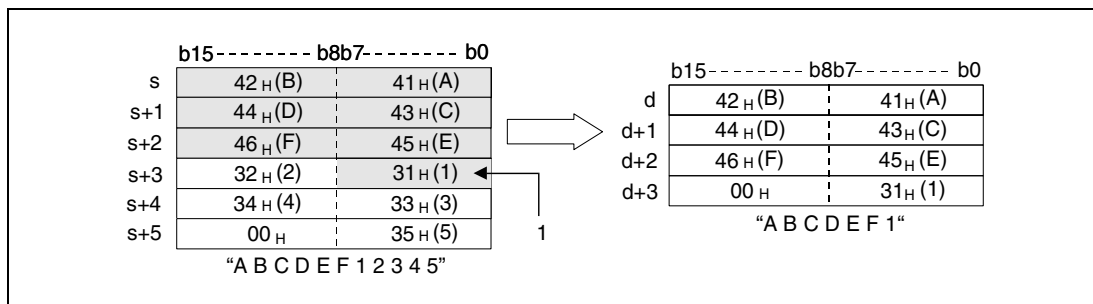
LEFT Estrae caratteri dalla parte sinistra della stringa

L'istruzione LEFT memorizza n caratteri del lato sinistro (parte iniziale) della stringa di caratteri contenuta a partire da s. I caratteri sono memorizzati a partire da d.



- ¹ Codice ASCII del secondo carattere/ codice ASCII del primo carattere
- ² Codice ASCII del quarto carattere/ codice ASCII del terzo carattere
- ³ Codice ASCII del carattere n-1/ codice ASCII del carattere n-2
- ⁴ Codice ASCII del carattere n+1/ codice ASCII dell'ennesimo carattere
- ⁵ "00H"/ codice ASCII dell'ultimo carattere
- ⁶ Codice ASCII del secondo carattere/ codice ASCII del primo carattere
- ⁷ Codice ASCII del quarto carattere/ codice ASCII del terzo carattere
- ⁸ Codice ASCII del carattere n-1/ codice ASCII del carattere n-2
- ⁹ "00H"/ codice ASCII dell'ennesimo carattere

Con n=7



- ¹ Codice ASCII del settimo carattere

Se il numero di caratteri in n è zero, il codice di carattere "00H" viene memorizzato a partire da d.

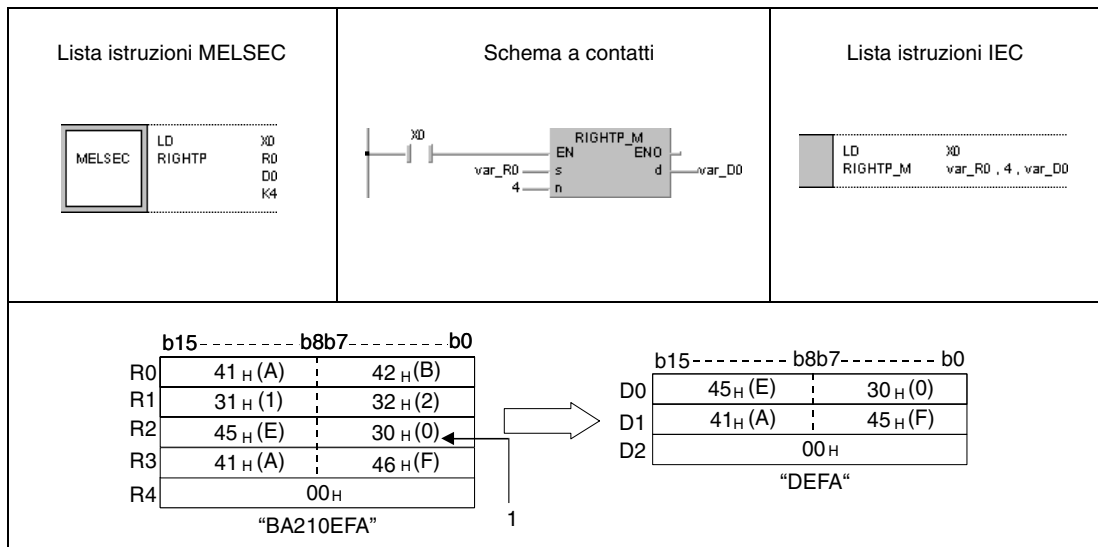
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il valore in n supera il numero di caratteri esistenti a partire da s (codice di errore 4101).
- L'area di dimensione n supera il campo operando ammesso per d. (codice di errore 4101).

Programma di esempio 1 RIGHTP

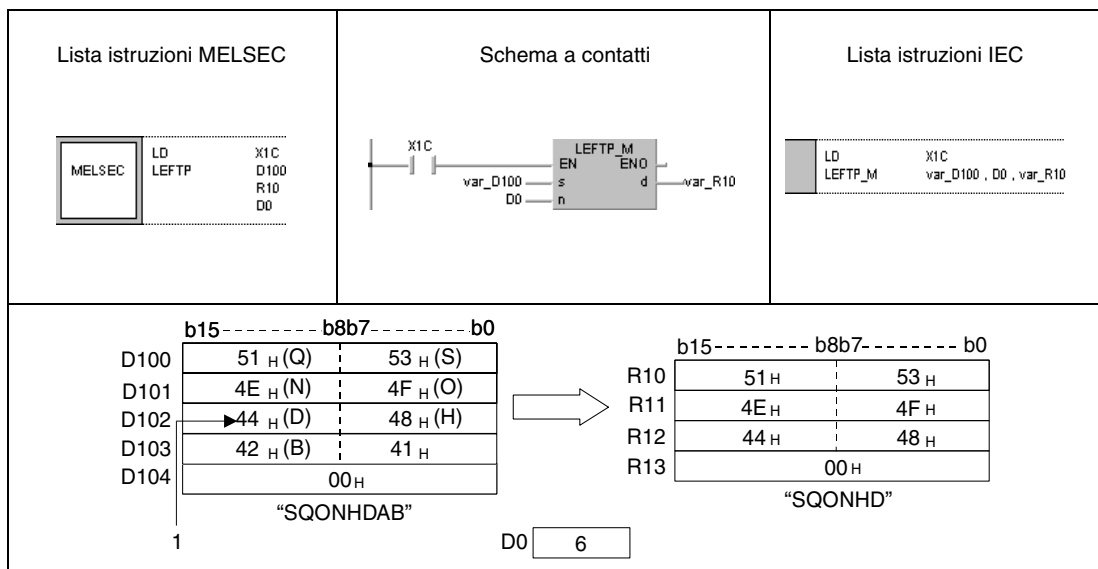
Con il fronte positivo di X0, il programma che segue estrae 4 caratteri dal lato destro della stringa contenuta da R0 a R4 e li memorizza da D0 a D2



¹ Codice ASCII del quarto carattere

Programma di esempio 2 LEFTP

Con il fronte positivo di X1C, il programma che segue estrae il numero di caratteri indicati da D0 dal lato sinistro della stringa contenuta da D100 a D104. Il risultato viene memorizzato da R10 a R13.



¹ Codice ASCII del sesto carattere

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provocano messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.11.17 MIDR, MIDRP, MIDW, MIDWP

CPU


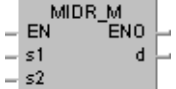
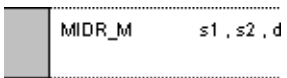
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

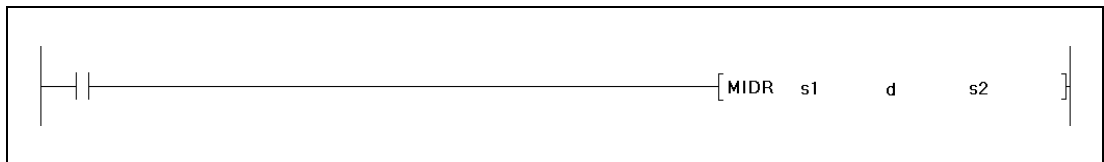
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Indice file Zn	Costanti \$			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	●	—	SM0	4
d	—	●	●	—	—	—	—	—	—		
s2	●	●	●	●	●	●	●	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
--	--	---

GX Developer



Variabili

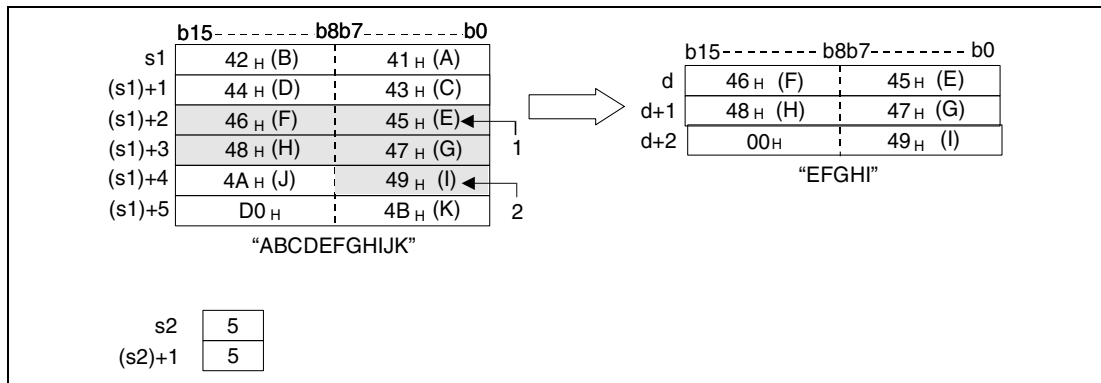
Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s1	Indirizzo del primo operando che contiene la stringa di caratteri.	Stringa caratteri	Stringa caratteri
d	Indirizzo del primo operando che memorizza il risultato dell'operazione.		
s2	Indirizzo dell'operando che memorizza il primo carattere e il numero di caratteri. (s2)+0: Registro del primo carattere (s2)+1: Numero di caratteri	BIN 16-bit	Array [1.0.2] di ANY16

Funzioni **Memorizzazione e spostamento di parti di stringhe di caratteri****MIDR** **Memorizzazione di parti specificate di una stringa di caratteri**

L'istruzione MIDR memorizza una parte specificata della stringa s nell'area a partire da quella indicata da d

Il primo carattere della parte da estrarre è indicato da s2 (Array_s2[1]) e viene contato iniziando dall'estremo sinistro della stringa (byte più leggero di s1).

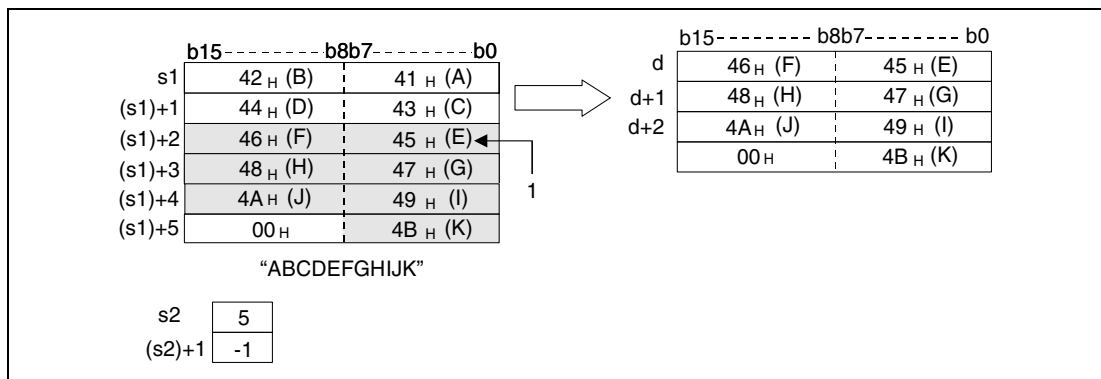
La lunghezza della parte da estrarre è specificato da s2+1 (Array_s2[2]).



¹ Posizione del quinto carattere (s2)

² Posizione dell'ultimo carattere da memorizzare

Se il numero di caratteri in (s2)+1 (Array_s2[2]) è zero, non viene eseguita nessuna operazione.



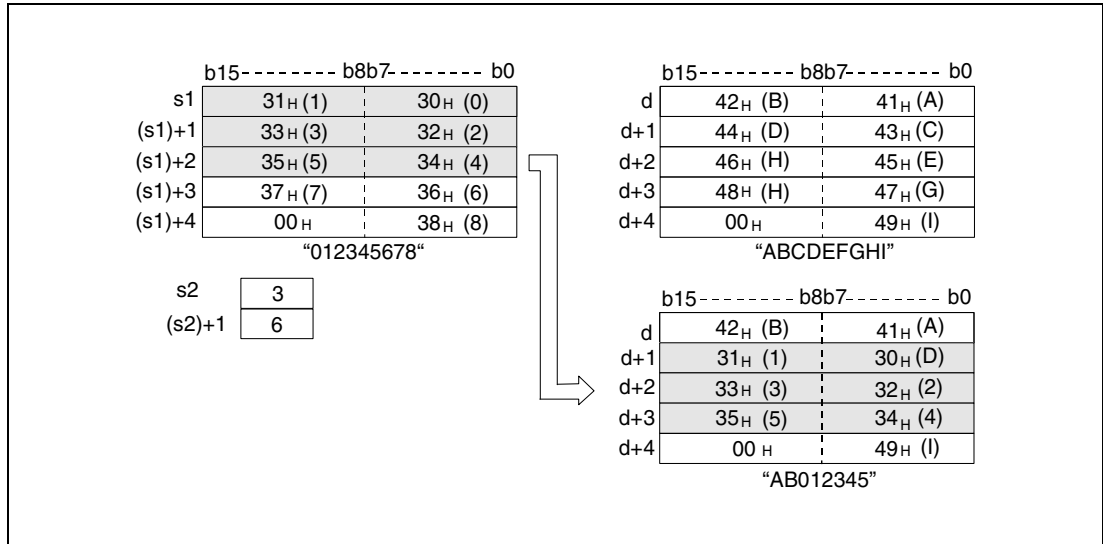
¹ Posizione del quinto carattere (s2)

MIDW Spostamento di parti definite di una stringa in una zona determinata

L'istruzione MIDW memorizza una parte di lunghezza specificata della stringa di caratteri indicata da s1 nell'area specificata da d e d+1.

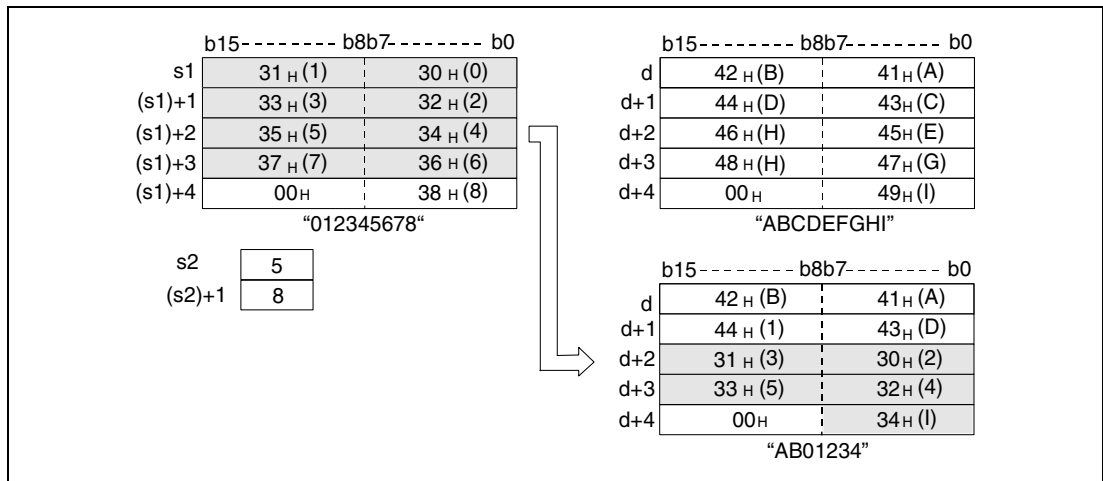
Il primo carattere della parte da memorizzare da d a d+n è indicato da s2 (Array_s[1]) e viene contato iniziando dall'estremo sinistro della stringa (byte più leggero di d).

La lunghezza della parte da estrarre è specificato da s2+1 (Array_s2[2]).

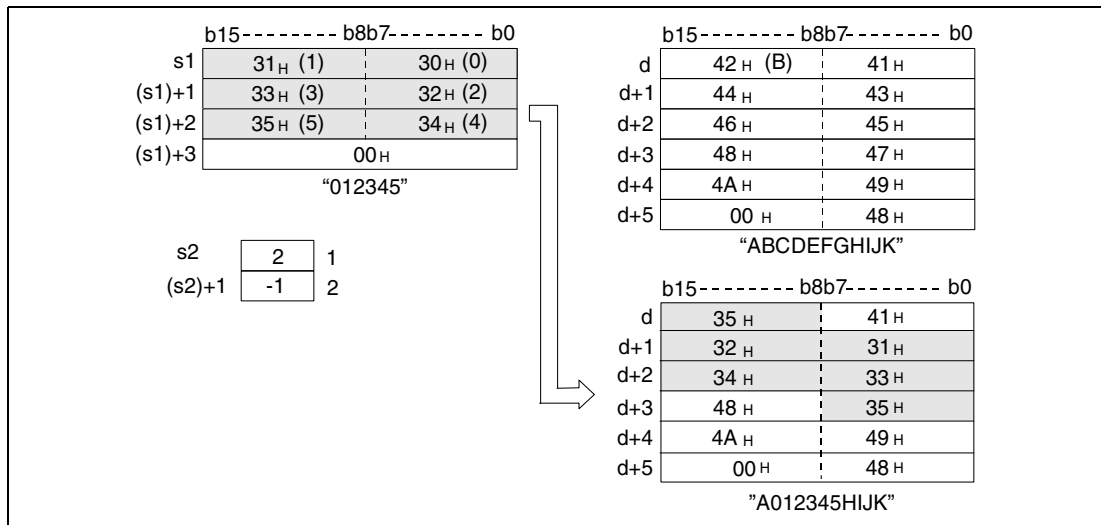


Se il numero di caratteri in (s2)+1 (Array_s2[2]) è zero, non viene eseguita nessuna operazione.

Se il numero di caratteri specificato in (s2)+1 (Array_s2[2]) supera l'area di memorizzazione indicata a partire da d, i caratteri in eccedenza vengono ignorati. Nella figura seguente, i caratteri da "35H" a "37H" non vengono memorizzati.



Se (s2)+1 (Array_s2[2]) contiene il valore -1, i caratteri vengono memorizzati a partire da s1.



Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

Per l'istruzione MIDR

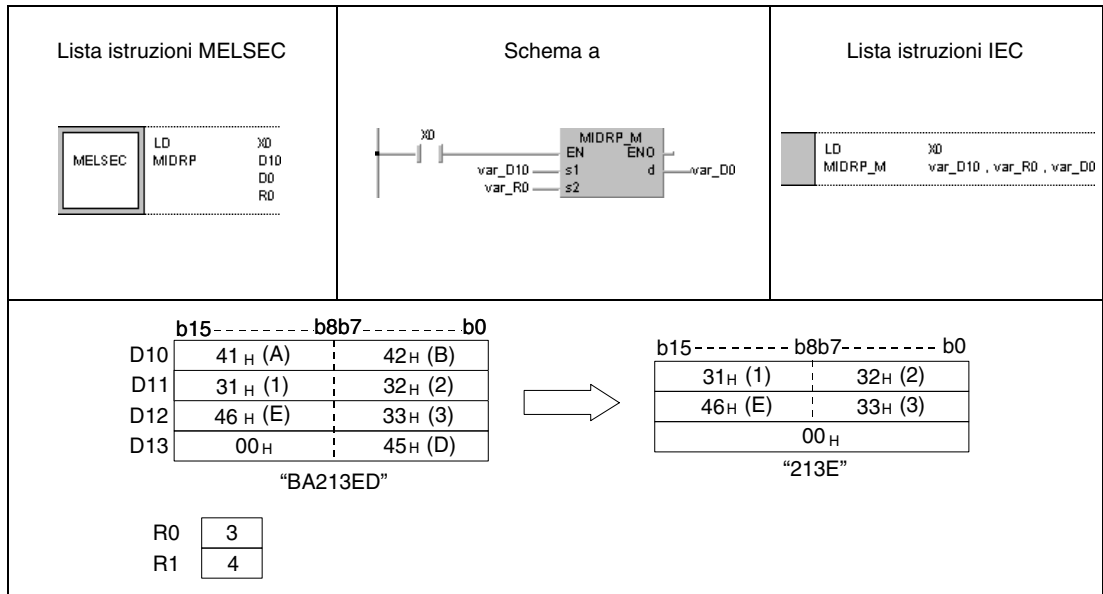
- L'indirizzo del primo carattere da estrarre contenuto in s2 (Array_s2[2]) supera il campo da s1 a (s1)+n (codice di errore 4101).
- L'indirizzo del primo carattere da estrarre contenuto in (s2)+1 (Array_s2[2]) supera il campo da d a d+n (codice di errore 4101).

Per l'istruzione MIDW

- L'indirizzo del primo carattere da estrarre contenuto in (s2) (Array_s2[1]) supera il campo da d a d+n (codice di errore 4101).
- L'indirizzo del primo carattere da estrarre contenuto in (s2)+1 (Array_s2[2]) supera il campo da s1 a (s1)+n (codice di errore 4101).

Programma di esempio 1 MIDRP

Con il fronte positivo di X0, il programma seguente estrae caratteri dalla stringa contenuta da D10 a D13 e li memorizza da D0 a D2. Il numero di caratteri da estrarre viene specificato in R1 (var_R0 Array [2]). La posizione iniziale nella stringa di origine viene specificata da R0 (var_R0 Array [1]).



7.11.18 INSTR, INSTRP

CPU

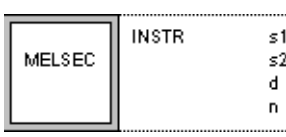
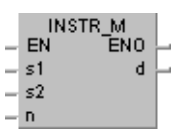
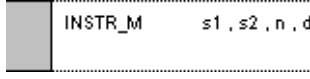
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

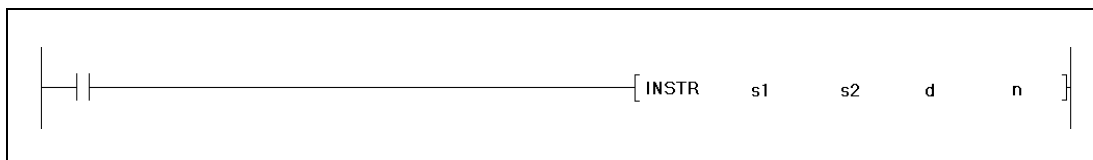
Operandi MELSEC Q

	Operandi utilizzabili										Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Indice file Zn	Costanti		Altro		
	Bit	Word		Bit	Word			K, H (16#)	\$			
s1	—	●	●	—	—	—	—	—	●	—	SM0	5
s2	—	●	●	—	—	—	—	—	●	—		
d	●	●	●	●	●	●	●	—	—	—		
n	●	●	●	●	●	●	●	●	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
---	---	--

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s1	Indirizzo del primo operando che memorizza la stringa di caratteri in cui eseguire la ricerca.	Stringa caratteri
s2	Indirizzo del primo operando che contiene la stringa da ricercare.	
d	Indirizzo del primo operando che memorizza il risultato della ricerca.	BIN 16-bit
n	Posizione iniziale della ricerca.	

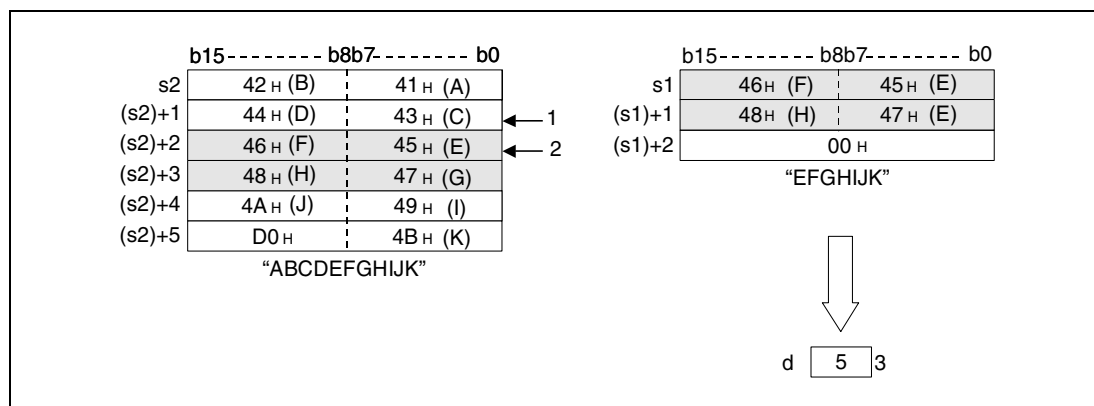
Funzioni Ricerca di stringhe di caratteri**INSTR Ricerca di stringhe di caratteri**

L'istruzione INSTR ricerca la stringa di caratteri specificata da s1 a (s1)+n all'interno della stringa specificata da s2 a (s2)+n.

La ricerca inizia dal carattere specificato da n.

Il primo carattere trovato viene memorizzato in d. Il carattere viene contato iniziando dall'estremo sinistro della stringa (byte leggero di s2).

Per n=3



¹ La ricerca inizia dal terzo carattere

² Primo carattere della stringa da ricercare

³ Risultato ricerca

Se la stringa ricercata non viene trovata, in d viene inserito uno zero.

Se il valore specificato da n è zero o negativo, non viene eseguita nessuna operazione.

Errori di esecuzione

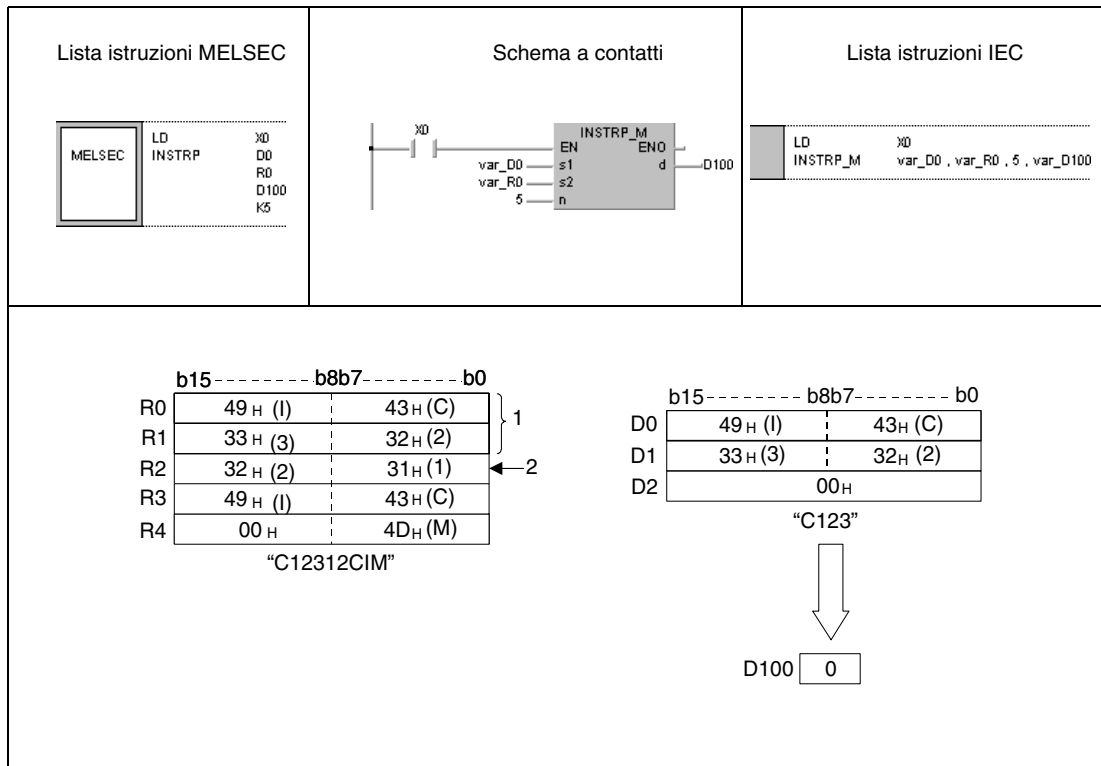
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- La posizione iniziale della ricerca indicata da n supera il campo da (s2) a (s2)+n. (codice di errore 4100).

Programma di esempio 1

INSTRP

Con il fronte positivo di X0, il programma seguente ricerca la stringa contenuta da D0 a D2 a partire dal quinto carattere della stringa contenuta a partire da R0. Il risultato (0) è memorizzato in D100.

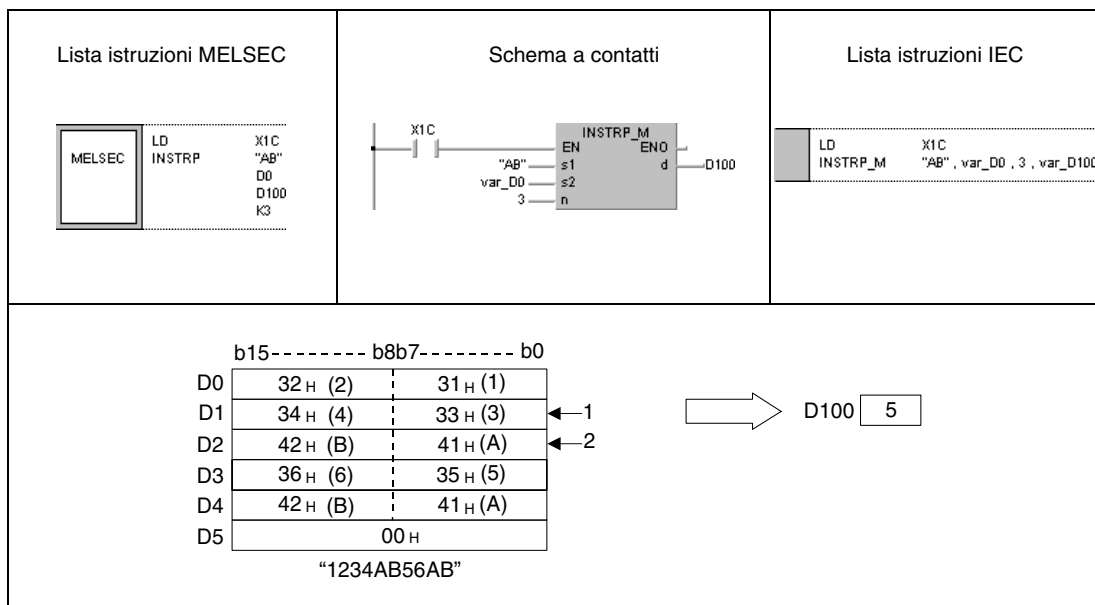


¹ La ricerca non viene eseguita in quest'area

² La ricerca inizia dal quinto carattere.

Programma di esempio 2 INSTRP

Con il fronte positivo di X0, il programma seguente ricerca la stringa "AB" a partire dal terzo carattere della stringa contenuta a partire da D0. Il risultato (5) è memorizzato in D100.



¹ La ricerca inizia dal terzo carattere.

² La stringa ricercata inizia dal quinto carattere.

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provocano messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.11.19 EMOD, EMODP

CPU

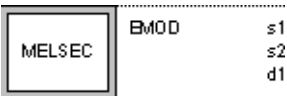
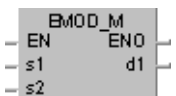
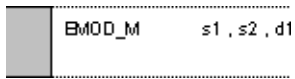
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

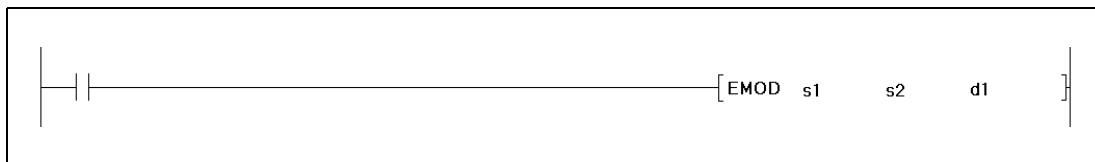
Operandi MELSEC Q

	Operandi utilizzabili										Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Indice file Zn	Costanti		Altro		
	Bit	Word		Bit	Word			K, H (16#)	E			
s1	—	●	●	—	●	●	—	—	●	—	SM0	4
s2	●	●	●	●	●	●	●	●	—	—		
d1	—	●	●	—	—	—	—	—	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
--	--	---

GX Developer



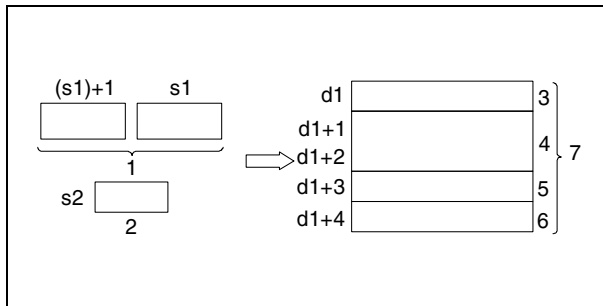
Variabili

Dati impostati	Significato	Tipo dati
s1	Dato virgola mobile (numero reale) o indirizzo del primo operando contenente il dato in virgola mobile	Numero reale
s2	Numero di cifre di cui viene spostato verso destra il punto decimale, oppure operando che contiene questo dato	BIN 16-bit
d1	Indirizzo del primo operando che contiene il numero in virgola mobile in formato BCD.	

Funzioni Conversione di un numero in virgola mobile in formato BCD

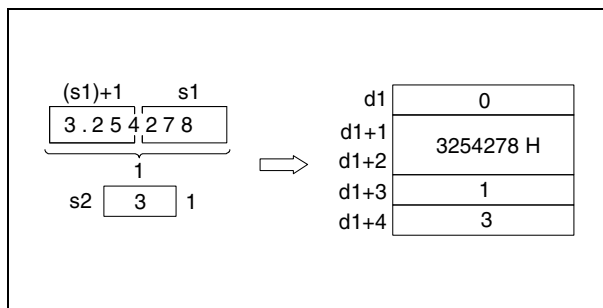
EMOD Conversione in formato BCD

L'istruzione EMOD calcola il formato BCD a partire dal numero in virgola mobile (numero reale) indicato da s1 e (s1)+1, considerando la posizione del punto decimale specificata da s2, partendo da destra. Il risultato viene memorizzato da d1 a (d1)+4.

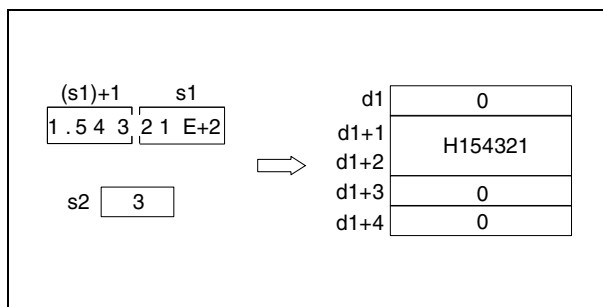
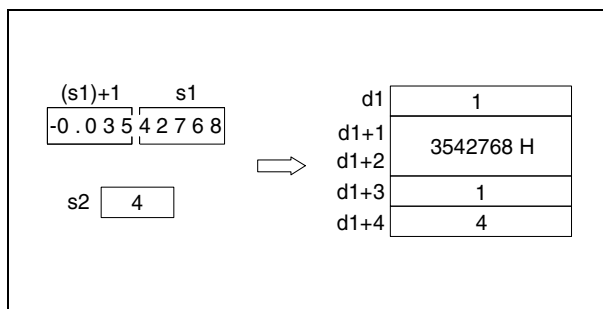


- ¹ Dato in virgola mobile (numero reale)
- ² Scorrimento verso destra del punto decimale
- ³ Bit del segno (0 = positivo / 1 = negativo)
- ⁴ 7 cifre BCD
- ⁵ Segno dell'esponente (0 = positivo / 1 = negativo)
- ⁶ Esponente BCD (valori da 0 a 38)
- ⁷ Numero virgola mobile in formato BCD

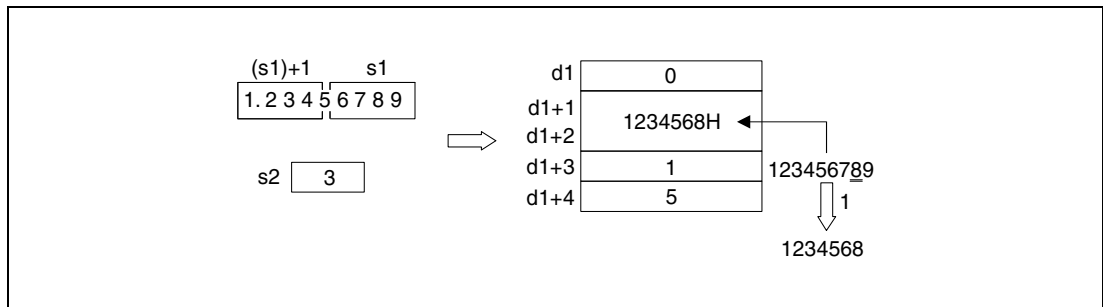
Le figure seguenti mostrano esempi di conversione.



- ¹ Dato in virgola mobile (numero reale)



Il numero in virgola mobile in s1 e (s1)+1 viene arrotondato alla settima cifra e memorizzato in (d1)+1 e (d1)+2.



¹ Arrotondamento

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il numero di cifre di scorrimento del punto decimale (s2) supera il campo da 0 a 7. (codice di errore 4100).
- Il valore inserito da d1 a (d1)+4 supera il campo operando ammesso (codice di errore 4101).

Esempio di programma

EMOD

Se X0 è attivo, il programma seguente converte il dato in virgola mobile (numero reale) contenuto in D0 e D1, considerando lo spostamento del punto decimale specificato in R10. Il risultato viene memorizzato da D100 a D104.

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.11.20 EREXP, EREXPP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

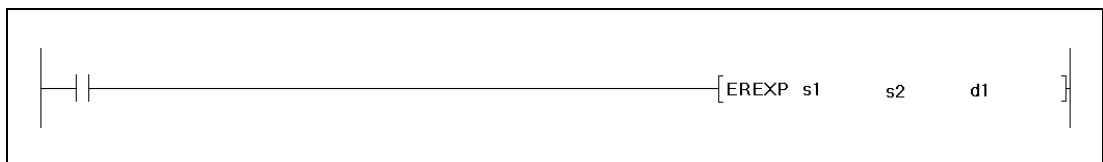
¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi
MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Indice file Zn	Costanti K, H (16#)	Altro U		
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	—	SM0	3
s2	●	●	●	●	●	●	●	●	—		
d1	—	●	●	—	●	●	—	—	—		

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
<div style="border: 1px solid black; padding: 5px; width: fit-content;">MELSEC</div> EREXP s1 s2 d1		EREXP_M s1, s2, d1

GX
Developer

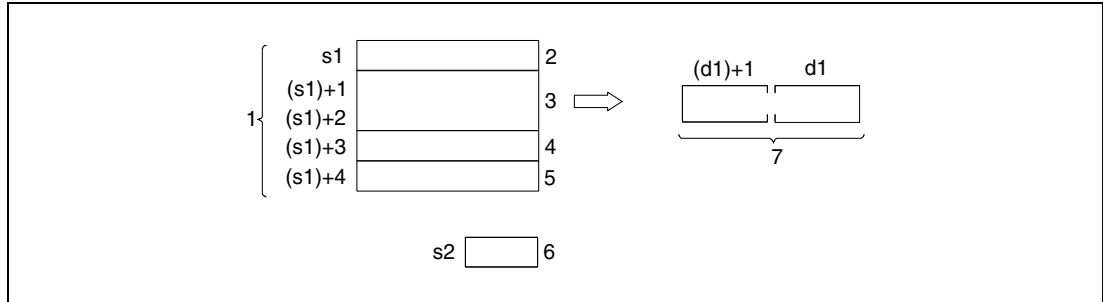
Variabili

Dati impostati	Significato	Tipo dati
s1	Indirizzo del primo operando che contiene il numero in virgola mobile in formato BCD.	BIN 16-bit
s2	Numero delle cifre decimali o operando che contiene questo dato.	
d1	Operando che contiene il dato in virgola mobile (numero reale).	Numero reale

Funzioni Conversione di un numero in virgola mobile in formato decimale

EREXP Conversione in formato decimale

L'istruzione EREXP calcola il formato decimale del numero in virgola mobile (numero reale) partendo dal dato virgola mobile in formato BCD contenuto da s1 a (s1)+4, considerando le cifre decimali indicate da s2. Il risultato viene memorizzato in d1 e (d1)+1.

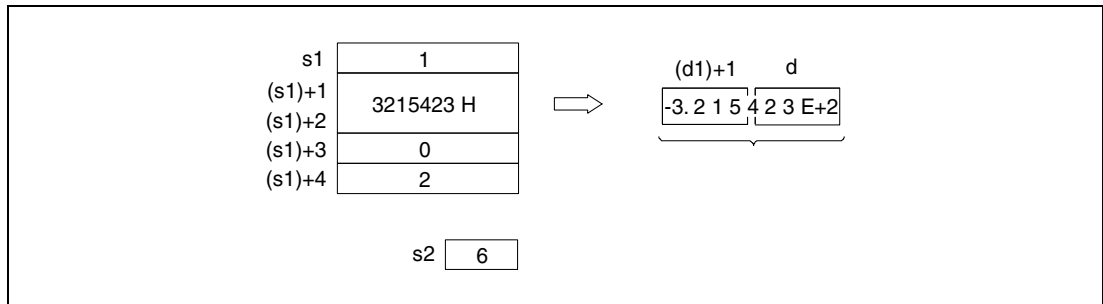


- ¹ Dato in virgola mobile in formato BCD
- ² Bit del segno (0 = positivo / 1 = negativo)
- ³ 7 cifre BCD
- ⁴ Segno dell'esponente (0 = positivo / 1 = negativo)
- ⁵ Esponente BCD (valori da 0 a 38)
- ⁶ Numero cifre decimali (valori da 0 a 7)
- ⁷ Dato in virgola mobile (numero reale)

Il segno in s1 e il segno dell'esponente in (s1)+3 sono a zero per valori positivi. Per valori negativi, il bit del segno è 1.

Il valore dell'esponente BCD in (s1)+4 può variare da 0 a 7.

Le cifre decimali in s2 possono variare da 0 a 7.



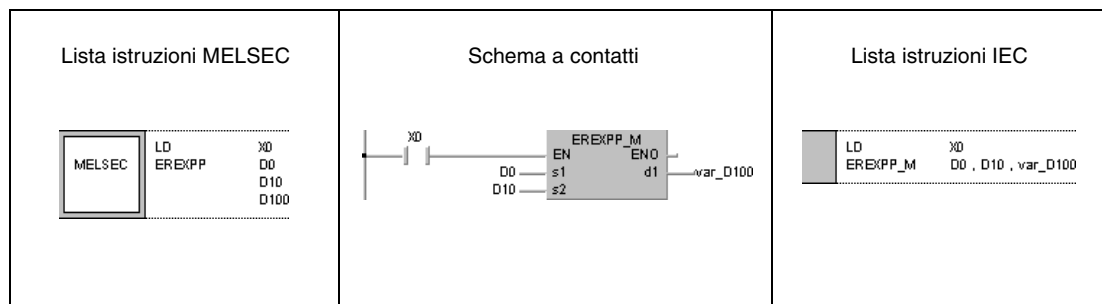
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il segno della destinazione in s1 non è 0 o 1 (codice di errore 4100).
- Il dato BCD in (s1)+1 e (s1)+2 contiene più di 8 cifre (codice di errore 4100).
- Il segno dell'esponente in (s1)+3 non è 0 o 1 (codice di errore 4100).
- Il dato dell'esponente in (s1)+4 supera il campo da 0 a 38 (codice di errore 4100).
- Il numero di cifre decimali in s2 supera il campo da 0 a 7 (codice di errore 4101).

Esempio di programma**EREXPP**

Con il fronte positivo di X0, il programma seguente calcola il valore in virgola mobile (numero reale) in formato decimale partendo dal valore virgola mobile in formato BCD contenuto da D0 a D4, considerando le cifre decimali indicate da D10. Il risultato viene memorizzato in D100 e D101.



7.12 Funzioni speciali

Funzione	Istruzione MELSEC in editor MELSEC	Istruzione MELSEC in editor IEC
Calcolo del seno	SIN	SIN_MD
		SIN_E_MD
	SINP	SIN_P_MD
		SIN_P_E_MD
Calcolo del coseno	COS	COS_MD
		COS_E_MD
	COSP	COS_P_MD
		COS_P_E_MD
Calcolo della tangente	TAN	TAN_MD
		TAN_E_MD
	TANP	TAN_P_MD
		TAN_P_E_MD
Calcolo arcoseno	ASIN	ASIN_MD
		ASIN_E_MD
	ASINP	ASIN_P_MD
		ASIN_P_E_MD
Calcolo arcocoseno	ACOS	ACOS_MD
		ACOS_E_MD
	ACOSP	ACOS_P_MD
		ACOS_P_E_MD
Calcolo arcotangente	ATAN	ATAN_MD
		ATAN_E_MD
	ATANP	ATAN_P_MD
		ATAN_P_E_MD
Conversione da gradi a radianti	RAD	RAD_MD
		RAD_E_MD
	RADP	RAD_P_MD
		RAD_P_E_MD
Conversione da radianti a gradi	DEG	DEG_MD
		DEG_E_MD
	DEGP	DEG_P_MD
		DEG_P_E_MD
Radice quadrata	SQR	SQR_MD
		SQR_E_MD
	SQRP	SQR_P_MD
		SQR_P_E_MD
Valore in virgola mobile come esponente di e	EXP	EXP_MD
		EXP_E_MD
	EXPP	EXP_P_MD
		EXP_P_E_MD

Funzione	Istruzione MELSEC in editor MELSEC	Istruzione MELSEC in editor IEC
Calcolo logaritmo (naturale)	LOG	LOG_MD
		LOG_E_MD
	LOGP	LOG_P_MD
		LOG_P_E_MD
Valore random	RND	RND_M
	RNDP	RNDP_M
Aggiorna valori random	SRND	SRND_M
	SRNDP	SRNDP_M
Calcolo radice quadrata da dato BCD a 4-cifre	BSQR	BSQR_MD
		BSQR_K_MD
	BSQRP	BSQR_P_MD
		BSQR_K_P_MD
Calcolo radice quadrata da dato BCD a 8-cifre	BDSQR	BDSQR_MD
		BDSQR_K_MD
	BDSQRP	BDSQR_P_MD
		BDSQR_K_P_MD
Calcolo seno da dato BCD	BSIN	BSIN_MD
		BSIN_K_MD
	BSINP	BSIN_P_MD
		BSIN_K_P_MD
Calcolo coseno da dato BCD	BCOS	BCOS_MD
		BCOS_K_MD
	BCOSP	BCOS_P_MD
		BCOS_K_P_MD
Calcolo tangente da dato BCD	BTAN	BTAN_MD
		BTAN_K_MD
	BTANP	BTAN_P_MD
		BTAN_K_P_MD
Calcolo arcoseno da dato BCD	BASIN	BASIN_MD
	BASINP	BASIN_P_MD
Calcolo arcocoseno da dato BCD	BACOS	BACOS_MD
	BACOSP	BACOS_P_MD
Calcolo arcotangente da dato BCD	BATAN	BATAN_MD
	BATANP	BATAN_P_MD

NOTA

Utilizzare istruzioni IEC con gli editor IEC

7.12.1 SIN, SINP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

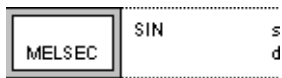
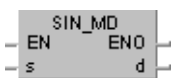
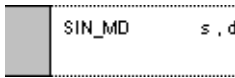
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

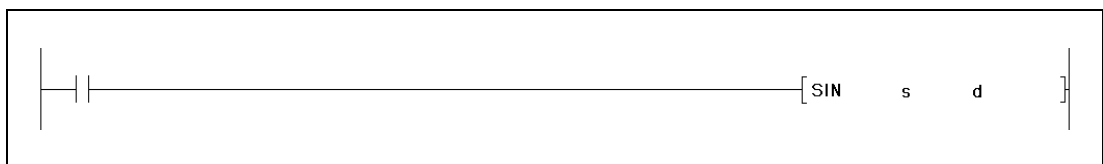
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Indice file Zn	Costanti E			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	●	●	—	●	—	3	
d	—	●	●	—	●	●	—	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
---	---	--

GX Developer

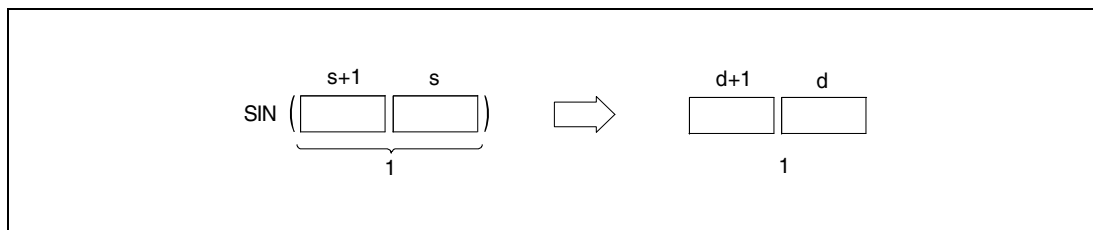


Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene il dato dell'angolo per l'istruzione SIN (seno).	Numero reale
d	Indirizzo del primo operando che memorizza il risultato dell'operazione.	

Funzioni **Calcolo del seno da valori in virgola mobile****SIN** **Calcolo del seno**

L'istruzione SIN calcola il valore del seno relativo all'angolo contenuto in s e s+1. Il risultato viene memorizzato in d e d+1.



¹ Valore in virgola mobile (numero reale)

L'angolo specificato da s e s+1 deve essere espresso in radianti (gradi $\times \pi/180$). La conversione da gradi a radianti viene descritta nella sezione relativa alle istruzioni RAD e DEG.

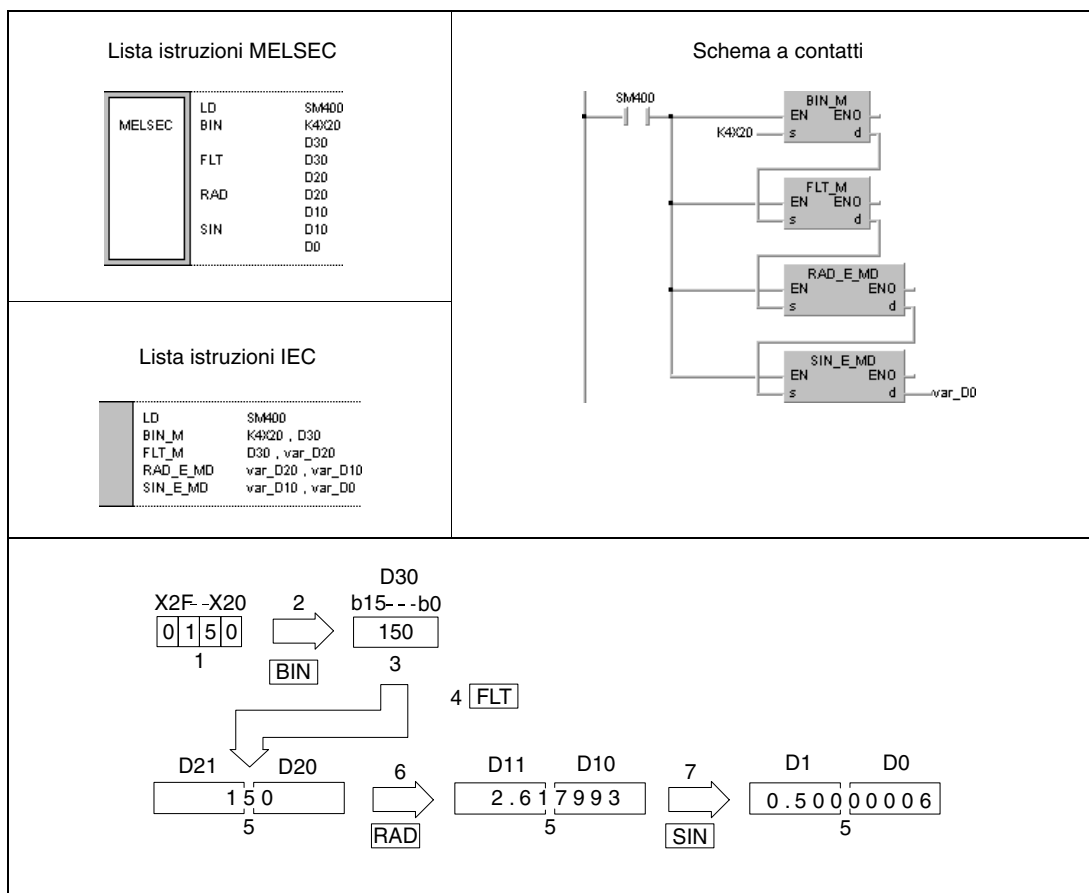
**Funzionamento
Errore**

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Solo per la CPU Q4AR: Se SM707 è OFF e l'operando specificato (s) contiene -0 (error code 4100).

Esempio di programma SIN

Se SM400 è attivo, il programma seguente calcola il valore del seno relativo al valore BCD a 4 cifre dell'angolo specificato da X20 a X2F. Il risultato viene memorizzato come valore in virgola mobile (numero reale) in D0 e D1.



- 1 Valore BCD
- 2 Conversione in formato BIN
- 3 Valore BIN
- 4 Conversione in formato virgola mobile
- 5 Valore in virgola mobile (numero reale)
- 6 Conversione in radianti
- 7 Calcolo valore del seno

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.12.2 COS, COSP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²


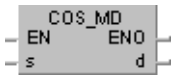
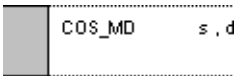
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

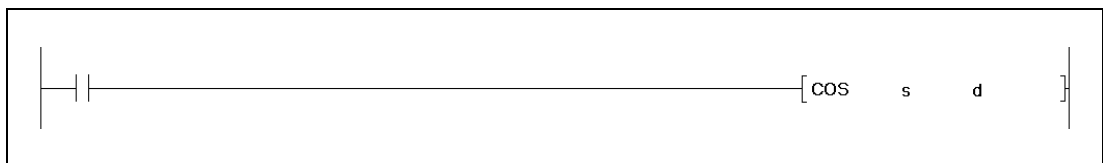
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Indice file Zn	Costanti E			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	●	●	—	●	—	3	
d	—	●	●	—	●	●	—	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
---	---	--

GX Developer



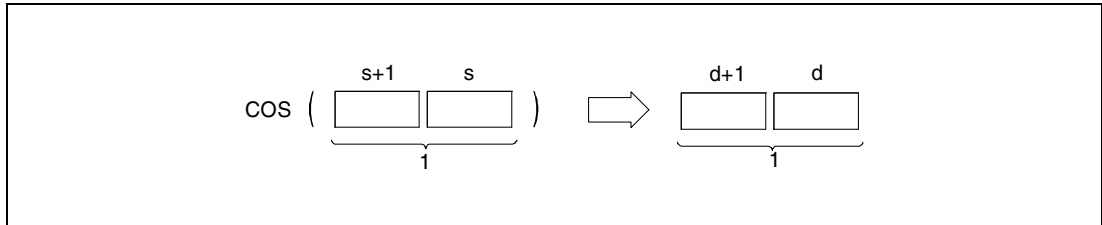
Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene il dato dell'angolo per l'istruzione COS (coseno).	Numero reale
d	Indirizzo del primo operando che memorizza il risultato dell'operazione.	

Funzioni **Calcolo del coseno da valori in virgola mobile**

COS **Calcolo del coseno**

L'istruzione COS calcola il valore del coseno relativo all'angolo contenuto in s e s+1. Il risultato viene memorizzato in d e d+1.



¹ Valore in virgola mobile (numero reale)

L'angolo specificato da s e s+1 deve essere espresso in radianti (gradi x $\pi/180$). La conversione da gradi a radianti viene descritta nella sezione relativa alle istruzioni RAD e DEG.

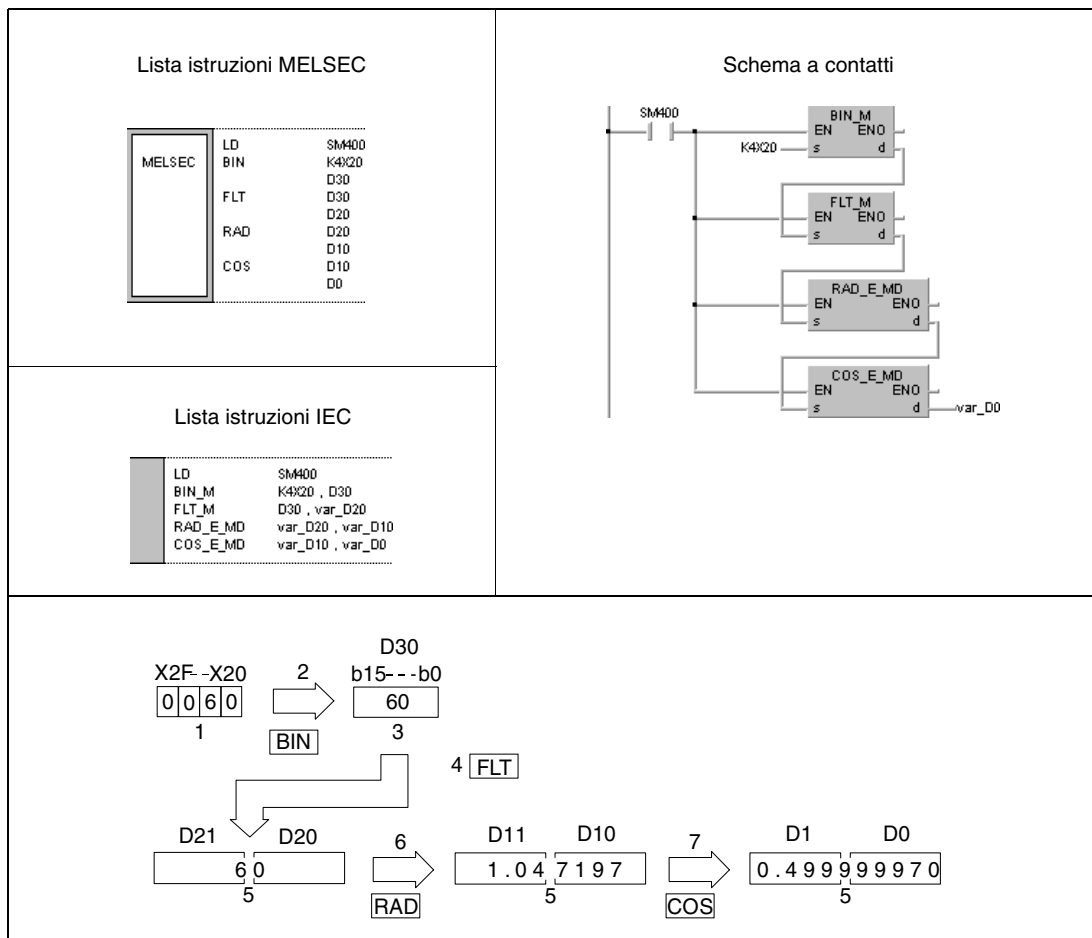
**Funzionamento
Errore**

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Solo per la CPU Q4AR: Se SM707 è OFF e l'operando specificato (s) contiene -0 (error code 4100).

Esempio di programma COS

Se SM400 è attivo, il programma seguente calcola il valore del coseno relativo al valore BCD a 4 cifre dell'angolo specificato da X20 a X2F. Il risultato viene memorizzato come valore in virgola mobile (numero reale) in D0 e D1.



- 1 Valore BCD
- 2 Conversione in formato BIN
- 3 Valore binario
- 4 Conversione in formato virgola mobile
- 5 Valore in virgola mobile (numero reale)
- 6 Conversione in radianti
- 7 Calcolo valore del coseno

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.12.3 TAN, TANP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

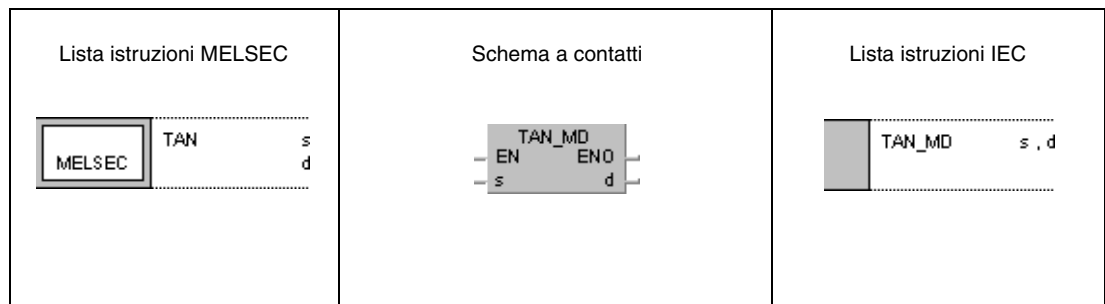
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

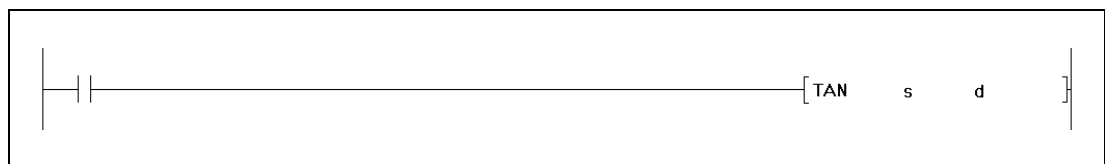
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Indice file Zn	Costanti E			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	●	●	—	●	—	SM0	3
d	—	●	●	—	●	●	—	—	—		

GX IEC Developer



GX Developer

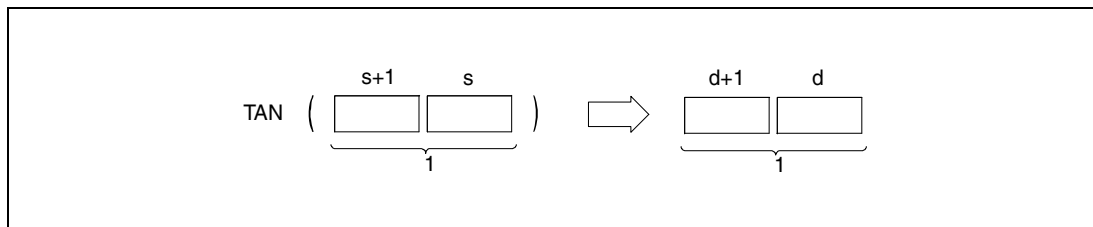


Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene il dato dell'angolo per l'istruzione TAN (tangente).	Numero reale
d	Indirizzo del primo operando che memorizza il risultato dell'operazione.	

Funzioni **Calcolo della tangente da valori in virgola mobile****TAN** **Calcolo della tangente**

L'istruzione TAN calcola il valore della tangente relativo all'angolo contenuto in s e s+1. Il risultato viene memorizzato in d e d+1.



¹ Valore in virgola mobile (numero reale)

L'angolo specificato da s e s+1 deve essere espresso in radianti (gradi $\times \pi/180$). La conversione da gradi a radianti viene descritta nella sezione relativa alle istruzioni RAD e DEG.

Se l'angolo contenuto in s e s+1 assume il valore $\pi/2$ rad o $(3/2)\times\pi$ rad, viene restituito un errore dal calcolo della conversione in radianti.

Errori di esecuzione

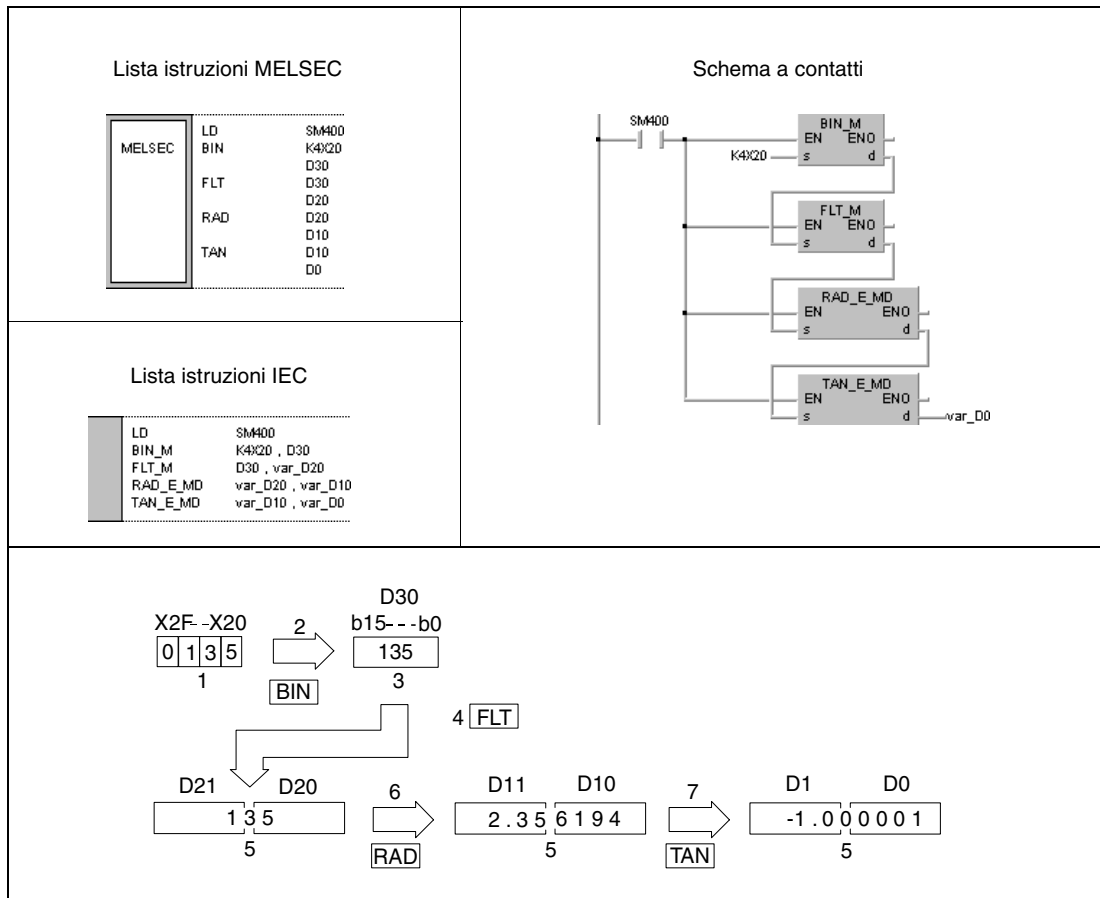
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il risultato dell'operazione è zero o non compreso nel campo da $\pm 2^{-127}$ a $\pm 2^{129}$ (codice di errore 4100).
- Solo per la CPU Q4AR: Se SM707 è OFF e l'operando specificato (s) contiene -0 (error code 4100).

Esempio di programma

TAN

Con il fronte positivo di SM400, il programma seguente calcola il valore della tangente relativo al valore BCD a 4-cifre dell'angolo specificato da X20 a X2F. Il risultato viene memorizzato come valore in virgola mobile (numero reale) in D0 e D1.



- 1 Valore BCD
- 2 Conversione in formato BIN
- 3 Valore binario
- 4 Conversione in formato virgola mobile
- 5 Valore in virgola mobile (numero reale)
- 6 Conversione in radianti
- 7 Calcolo valore della tangente

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.12.4 ASIN, ASINP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

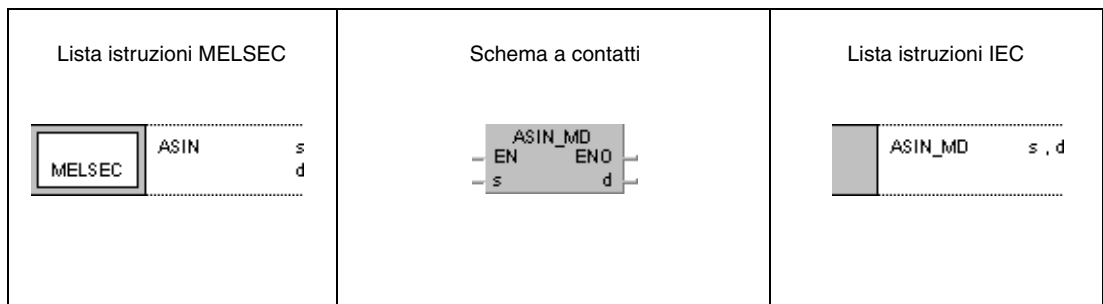
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

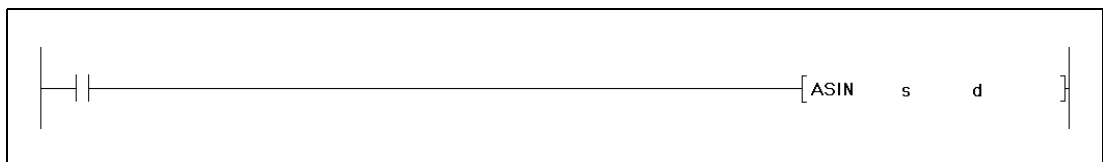
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Indice file Zn	Costanti E			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	●	●	—	●	—	SM0	3
d	—	●	●	—	●	●	—	—	—		

GX IEC Developer



GX Developer

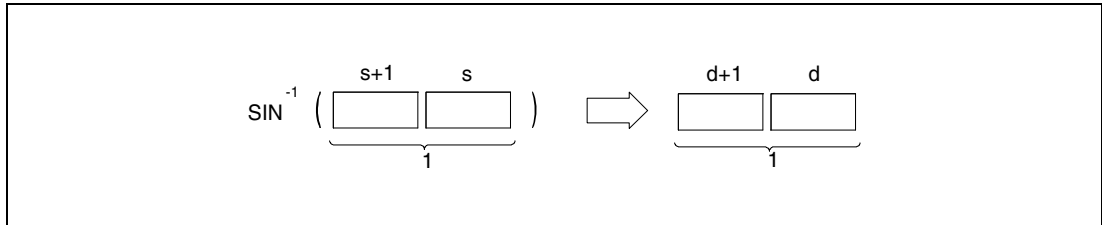


Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene il valore del seno per il calcolo dell'arcoseno.	Numero reale
d	Indirizzo del primo operando che memorizza il risultato dell'operazione.	

Funzioni **Calcolo del arcoseno da valori in virgola mobile****ASIN** **Calcolo arcoseno**

L'istruzione ASIN calcola il valore dell'angolo relativo al valore del seno contenuto in s e s+1. Il risultato viene memorizzato in d e d+1.



¹ Valore in virgola mobile (numero reale)

Il valore del seno in s e s+1 può variare nel campo da -1 a +1.

L'angolo specificato da s e s+1 deve essere espresso in radianti (gradi x $\pi/180$). La conversione da gradi a radianti viene descritta nella sezione relativa alle istruzioni RAD e DEG.

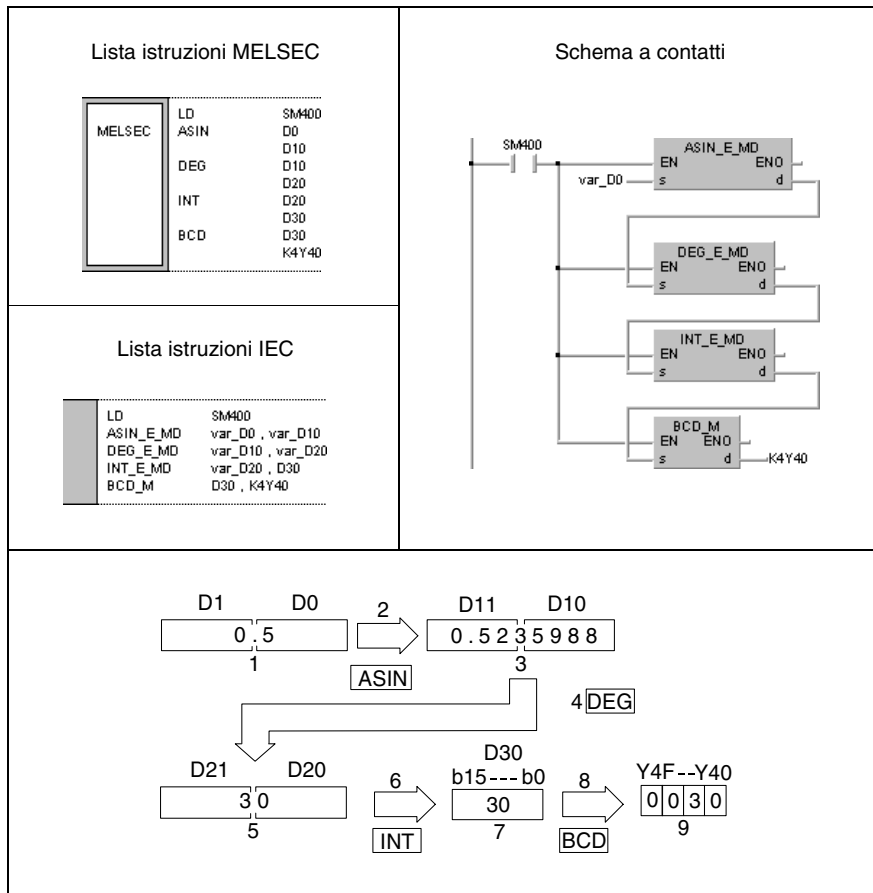
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il valore in s e s+1 supera il campo ammesso da -1 a +1 (codice di errore 4100).
- Solo per la CPU Q4AR: Se SM707 è OFF e l'operando specificato (s) contiene -0 (codice di errore 4100).

Esempio di programma ASIN

Se SM400 è attivo, il programma seguente calcola il valore dell'arcoseno dal valore in virgola mobile (numero reale) contenuto in D0 e D1. La misura in radianti dell'angolo risultante viene emessa sulle uscite da Y20 a Y4F, come valore BCD a 4 cifre.



- 1 Valore in virgola mobile (numero reale)
- 2 Calcolo arcoseno
- 3 Valore in virgola mobile (numero reale)
- 4 Conversione di misure angolari
- 5 Valore in virgola mobile (numero reale)
- 6 Conversione in formato BIN
- 7 Valore binario
- 8 Conversione in formato BCD
- 9 Valore BCD

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.12.5 ACOS, ACOSP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

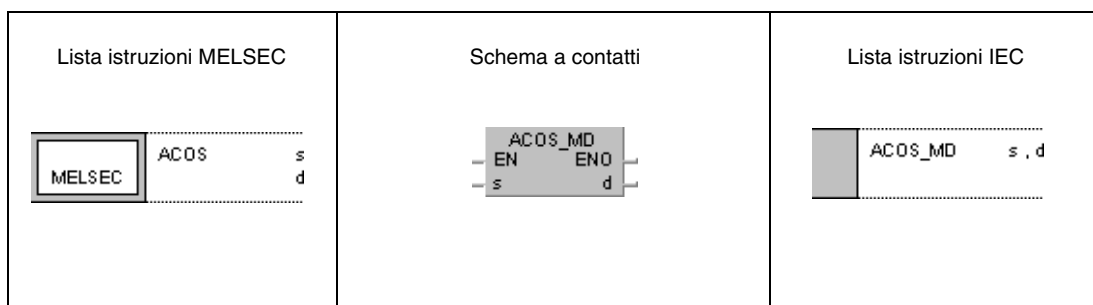
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

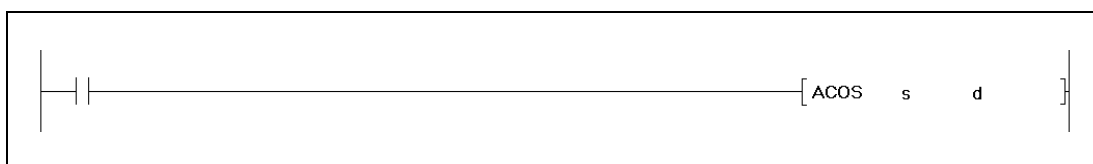
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Indice file Zn	Costanti E			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	●	●	—	●	—	SM0	3
d	—	●	●	—	●	●	—	—	—		

GX IEC Developer



GX Developer

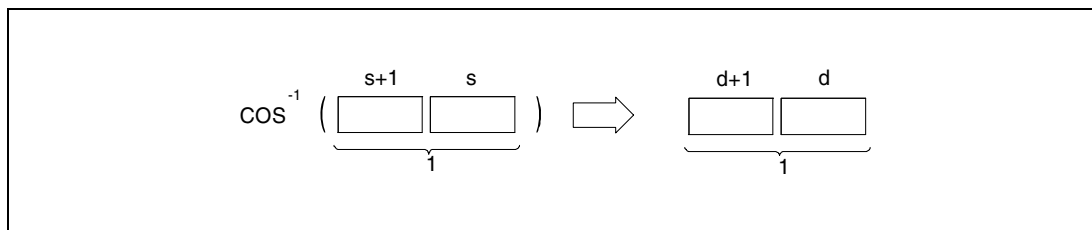


Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene il valore del coseno per il calcolo dell'arcocoseno.	Numero reale
d	Indirizzo del primo operando che memorizza il risultato dell'operazione.	

Funzioni **Calcolo del arcocoseno da valori in virgola mobile****ACOS** **Calcolo arcocoseno**

L'istruzione ACOS calcola il valore dell'angolo relativo al valore del coseno contenuto in s e s+1. Il risultato viene memorizzato in d e d+1.



¹ Valore in virgola mobile (numero reale)

Il valore del coseno in s e s+1 può variare nel campo da -1 a +1.

L'angolo specificato da s e s+1 deve essere espresso in radianti (gradi $\times \pi/180$). La conversione da gradi a radianti viene descritta nella sezione relativa alle istruzioni RAD e DEG.

Errori di esecuzione

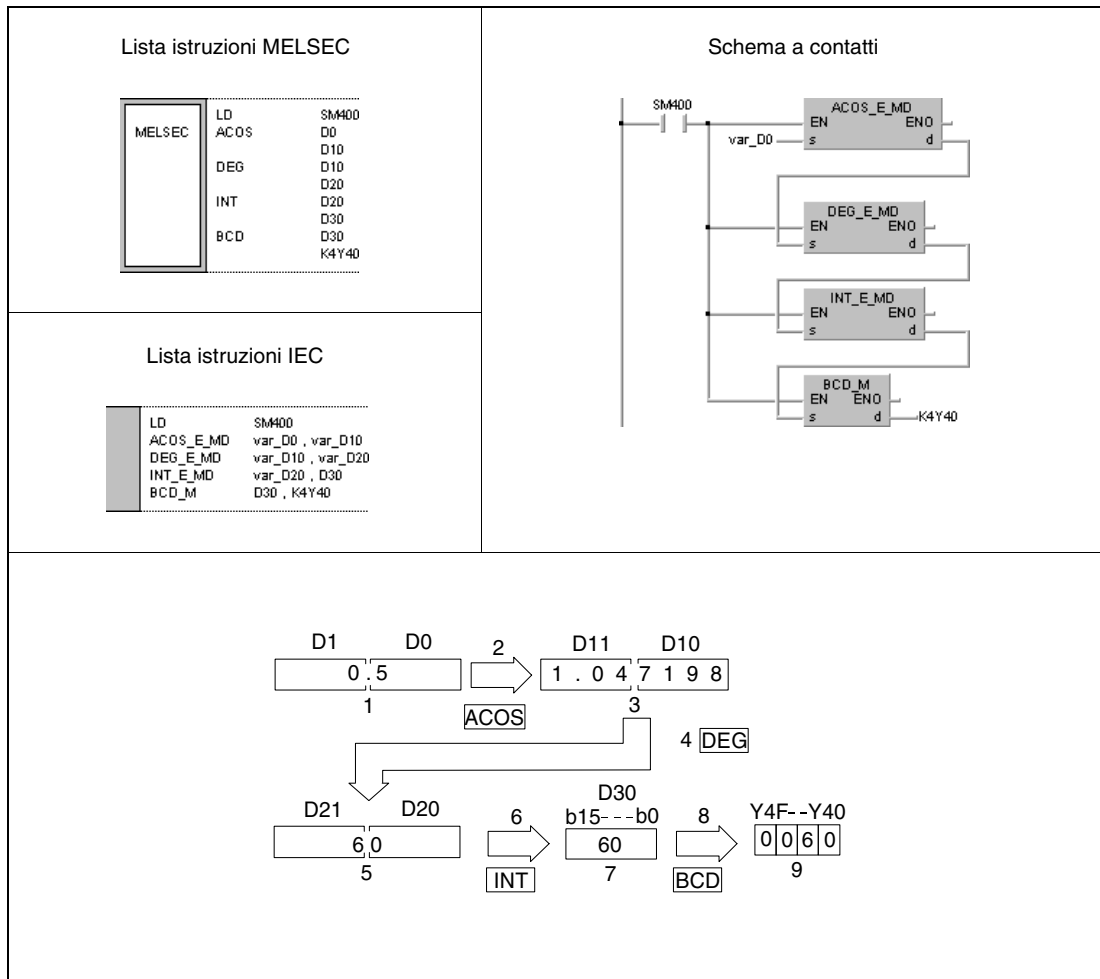
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il valore in s e s+1 supera il campo ammesso da -1 a +1 (codice di errore 4100).
- Solo per la CPU Q4AR: Se SM707 è OFF e l'operando specificato (s) contiene -0 (codice di errore 4100).

Esempio di programma

ACOS

Se SM400 è attivo, il programma seguente calcola il valore dell'arcocoseno dal valore in virgola mobile (numero reale) contenuto in D0 e D1. La misura in radianti dell'angolo risultante viene emessa sulle uscite da Y20 a Y4F, come valore BCD a 4 cifre.



- 1 Valore in virgola mobile (numero reale)
- 2 Calcolo arcocoseno
- 3 Valore in virgola mobile (numero reale)
- 4 Conversione di misure angolari
- 5 Valore in virgola mobile (numero reale)
- 6 Conversione in formato BIN
- 7 Valore binario
- 8 Conversione in formato BCD
- 9 Valore BCD

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.12.6 ATAN, ATANP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

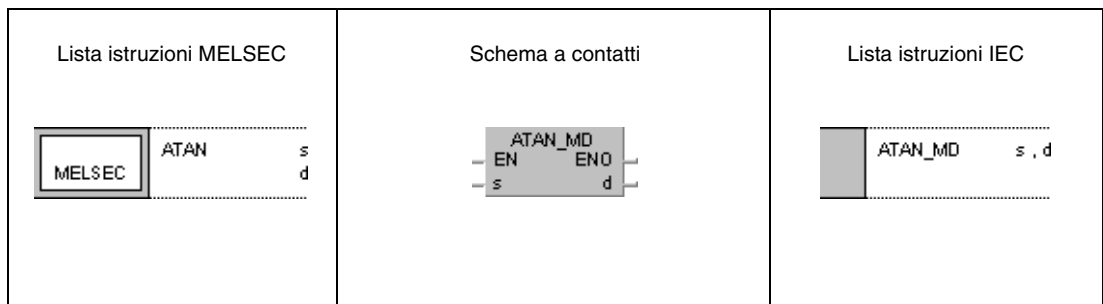
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

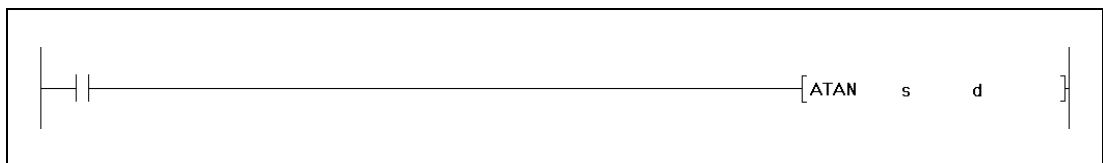
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Indice file Zn	Costanti E			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	●	●	—	●	—	3	
d	—	●	●	—	●	●	—	—	—		

GX IEC Developer



GX Developer



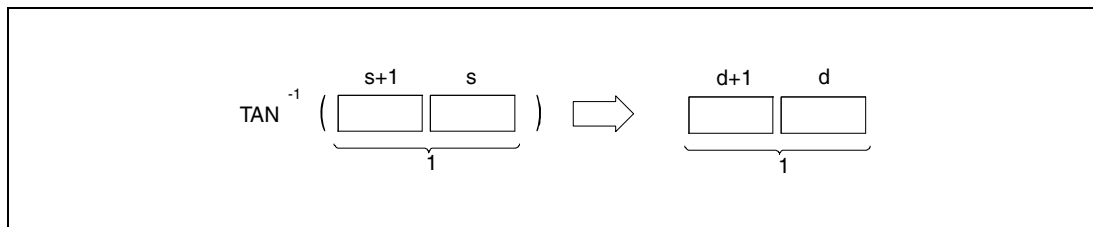
Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene il valore della tangente per il calcolo dell'arcotangente.	Numero reale
d	Indirizzo del primo operando che memorizza il risultato dell'operazione.	

Funzioni **Calcolo del arcotangente da valori in virgola mobile**

ATAN **Calcolo arcotangente**

L'istruzione ATAN calcola il valore dell'angolo relativo al valore della tangente contenuto in s e s+1. Il risultato viene memorizzato in d e d+1.



¹ Valore in virgola mobile (numero reale)

L'angolo specificato da s e s+1 deve essere espresso in radianti (gradi x $\pi/180$). La conversione da gradi a radianti viene descritta nella sezione relativa alle istruzioni RAD e DEG.

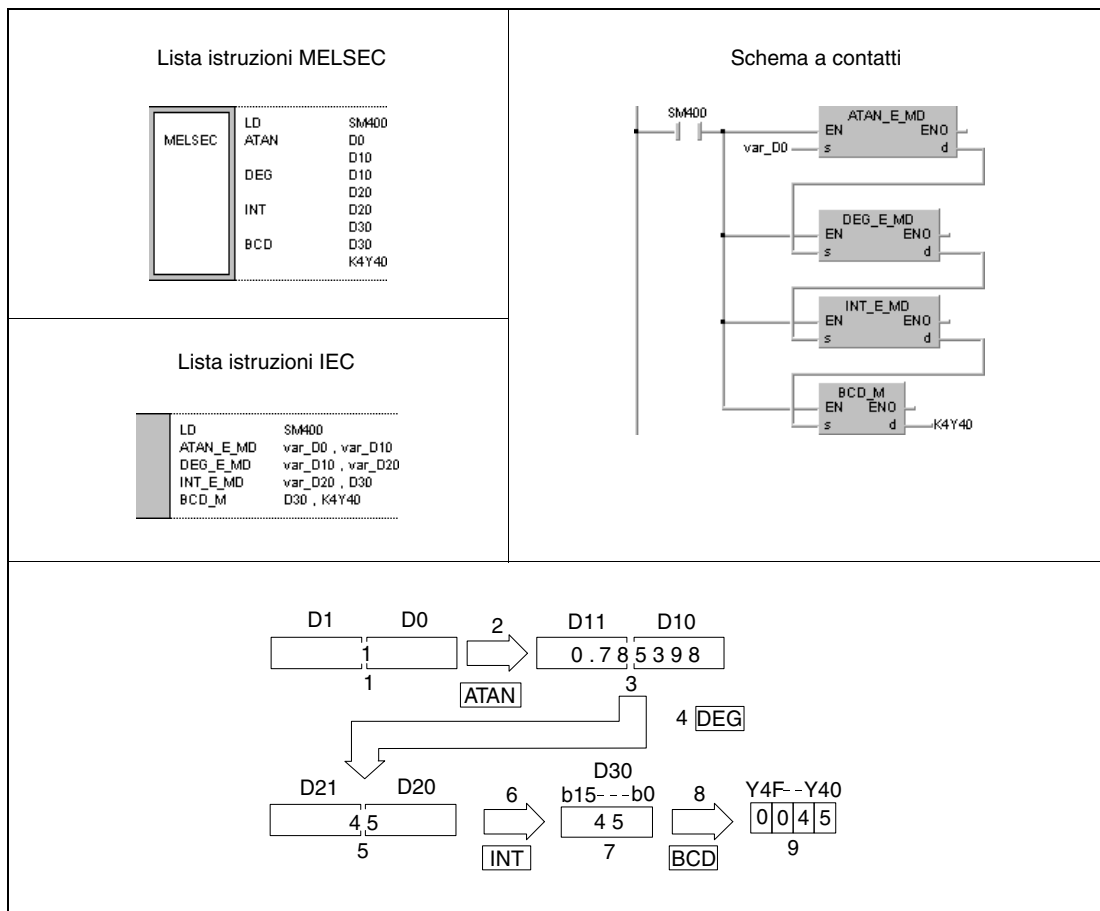
**Funzionamento
Errore**

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Solo per la CPU Q4AR: Se SM707 è OFF e l'operando specificato (s) contiene -0 (codice di errore 4100).

Esempio di programma ATAN

Se SM400 è attivo, il programma seguente calcola il valore dell'arcotangente dal valore in virgola mobile (numero reale) contenuto in D0 e D1. La misura in radianti dell'angolo risultante viene emessa sulle uscite da Y20 a Y4F, come valore BCD a 4 cifre.



- 1 Valore in virgola mobile (numero reale)
- 2 Calcolo arcotangente
- 3 Valore in virgola mobile (numero reale)
- 4 Conversione di misure angolari
- 5 Valore in virgola mobile (numero reale)
- 6 Conversione in formato BIN
- 7 Valore binario
- 8 Conversione in formato BCD
- 9 Valore BCD

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.12.7 RAD, RADP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²


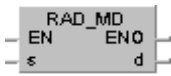
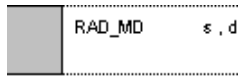
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

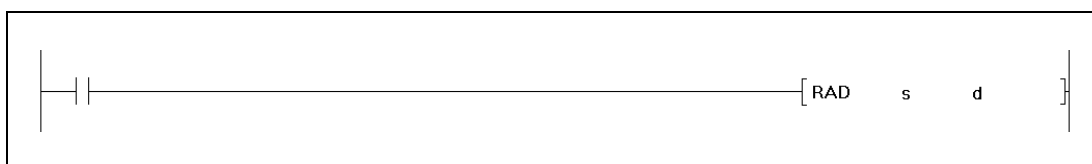
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Indice file Zn	Costanti E			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	●	●	—	●	—	3	
d	—	●	●	—	●	●	—	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
---	---	--

GX Developer

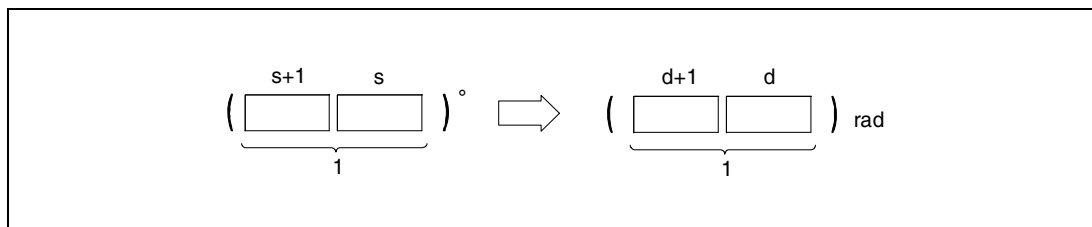


Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene il valore in gradi da convertire in radianti.	Numero reale
d	Indirizzo del primo operando che memorizza il risultato della conversione.	

Funzioni **Conversione da gradi a radianti con valori in virgola mobile****RAD** **Conversione da gradi a radianti**

L'istruzione RAD calcola il valore in radianti (rad) dal valore in gradi (°) contenuto in s e s+1. Il risultato viene memorizzato in d e d+1.



¹ Valore in virgola mobile (numero reale)

La conversione da gradi a radianti usa la seguente equazione:

$$\text{Valore in radianti} = \text{Valore in gradi} \times \pi / 180$$

**Funzionamento
Errore**

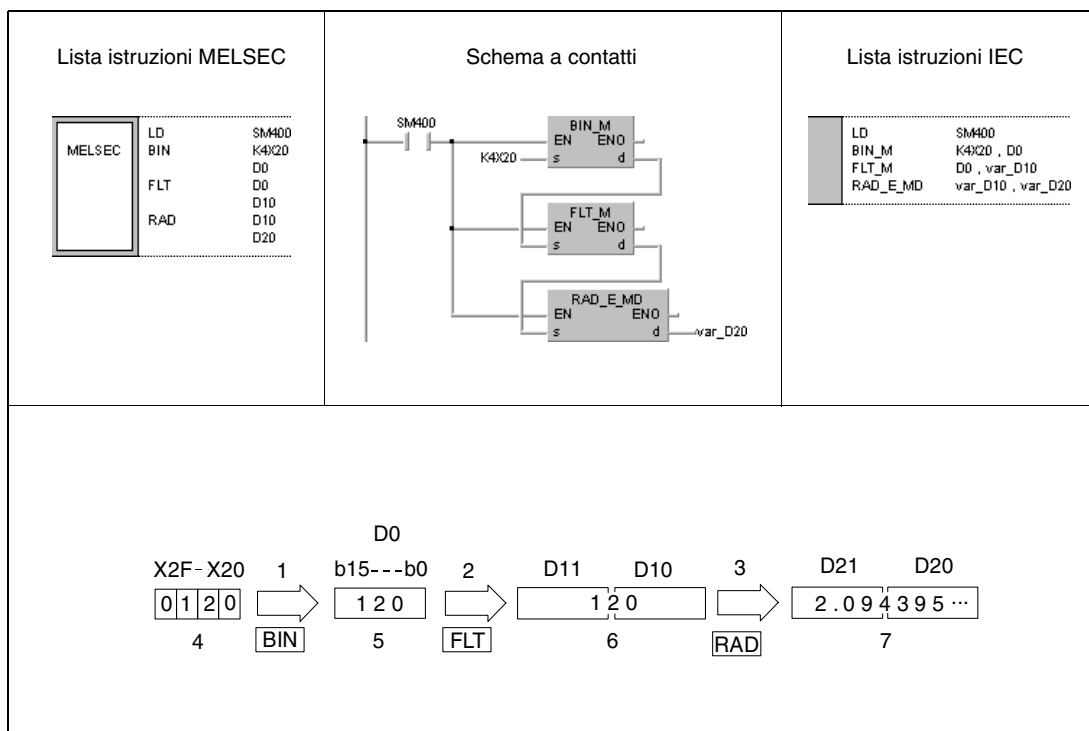
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Solo per la CPU Q4AR: Se SM707 è OFF e l'operando specificato (s) contiene -0 (codice di errore 4100).

Esempio di programma

RAD

Se SM400 è attivo, il programma seguente calcola il valore in radianti corrispondente al valore BCD a 4 cifre in gradi, contenuto da X20 a X2F. Il risultato viene memorizzato in D20 e D21 come valore in virgola mobile.



- 1 Conversione in formato BIN
- 2 Conversione in formato virgola mobile
- 3 Conversione in radianti
- 4 Valore BCD
- 5 Valore binario
- 6 Valore in virgola mobile (numero reale)
- 7 Valore in virgola mobile (numero reale)

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.12.8 DEG, DEGP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²


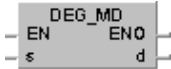
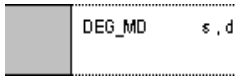
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

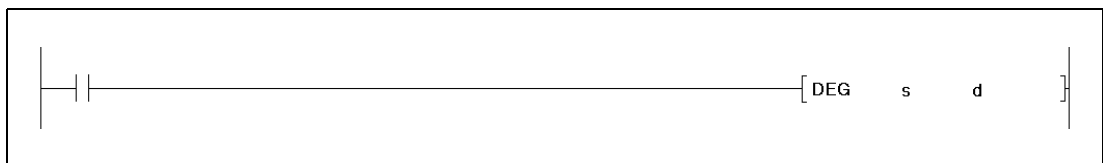
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Indice file Zn	Costanti E			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	●	●	—	●	—	3	
d	—	●	●	—	●	●	—	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
---	---	--

GX Developer

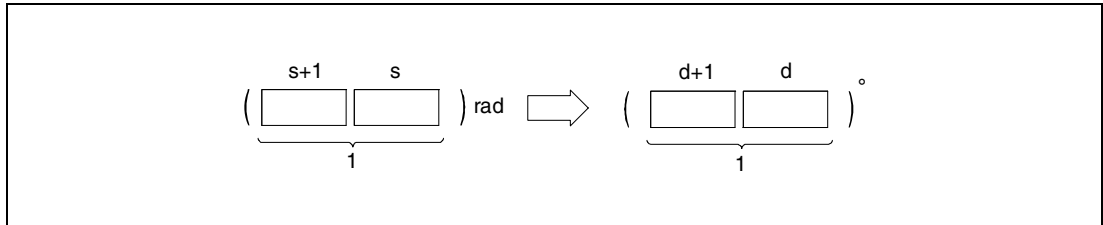


Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene il valore in radianti da convertire in gradi.	Numero reale
d	Indirizzo del primo operando che memorizza il risultato della conversione.	

Funzioni **Conversione da radianti a gradi con valori in virgola mobile****DEG** **Conversione da radianti a gradi**

L'istruzione DEG calcola il valore in gradi (°) dal valore in radianti (rad) contenuto in s e s+1. Il risultato viene memorizzato in d e d+1.



¹ Valore in virgola mobile (numero reale)

La conversione da radianti a gradi usa la seguente equazione:

$$\text{Valore in gradi} = \text{Valore in radianti} \times 180 / \pi$$

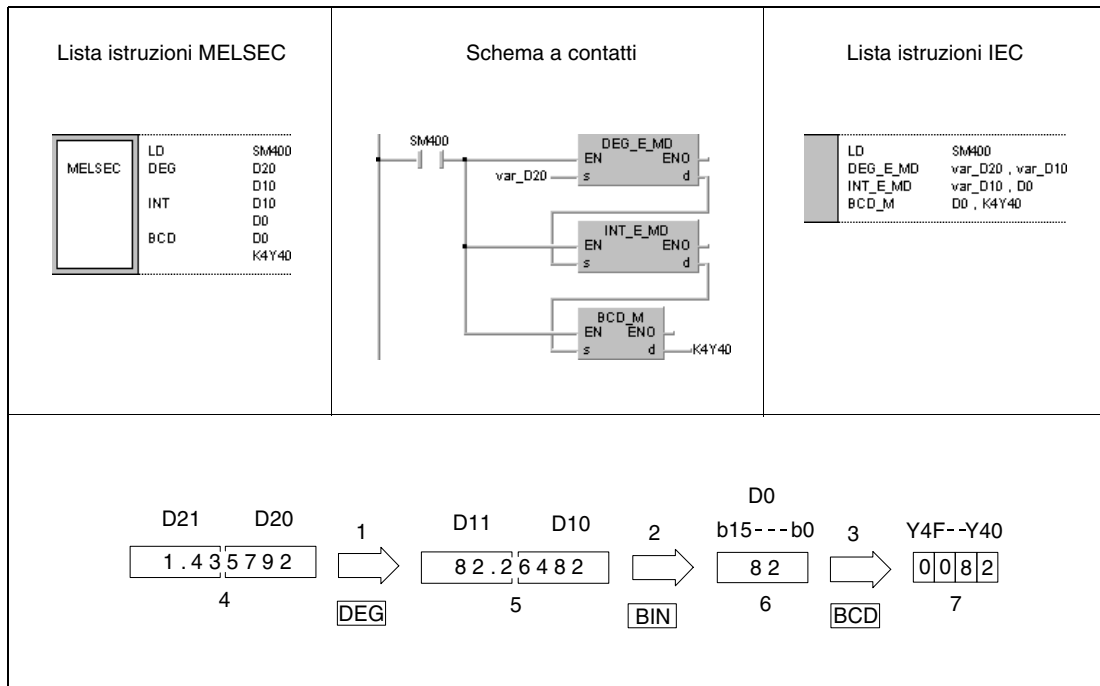
**Funzionamento
Errore**

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Solo per la CPU Q4AR: Se SM707 è OFF e l'operando specificato (s) contiene -0 (codice di errore 4100).

Esempio di programma DEG

Se SM400 è attivo, il programma seguente calcola il valore in gradi dal valore in radianti memorizzato in D20 e D21 come valore BCD a 4 cifre. Il risultato viene memorizzato in D20 e D21 come valore in virgola mobile.



- 1 Conversione in gradi
- 2 Conversione in formato BIN
- 3 Conversione in formato BCD
- 4 Valore in virgola mobile (numero reale)
- 5 Valore in virgola mobile (numero reale)
- 6 Valore binario
- 7 Valore BCD

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.12.9 SQR, SQRP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

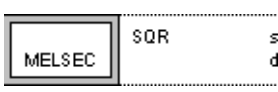
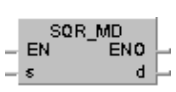
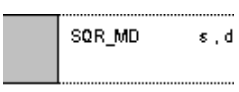
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

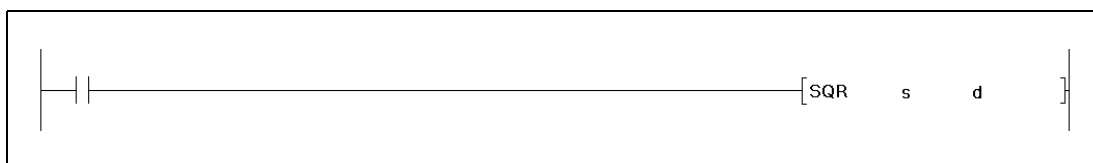
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Indice file Zn	Costanti E			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	●	●	—	●	—	SM0	3
d	—	●	●	—	●	●	—	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
---	---	--

GX Developer

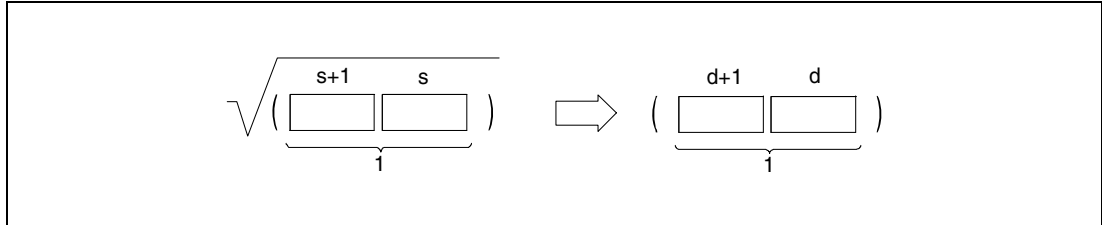


Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene il valore per il calcolo della radice quadrata.	Numero reale
d	Indirizzo del primo operando che memorizza il risultato dell'estrazione di radice.	

Funzioni **Calcolo della radice quadrata da valori in virgola mobile****SQR** **Calcolo radice quadrata**

L'istruzione SQR calcola il valore della radice quadrata relativo al valore in virgola mobile contenuto in s e s+1. Il risultato viene memorizzato in d e d+1.



¹ Valore in virgola mobile (numero reale)

In s e s+1 possono essere contenuti solo valori positivi.
(valori negativi non possono essere elaborati).

Errori di esecuzione

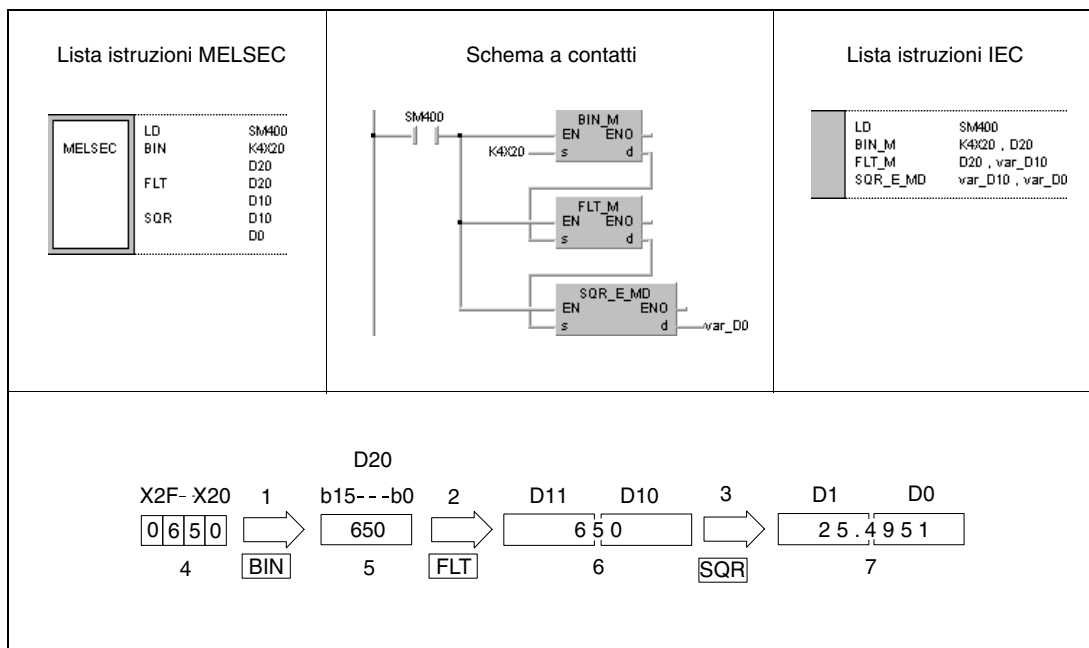
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il valore contenuto in s è negativo.
- Solo per la CPU Q4AR: Se SM707 è OFF e l'operando specificato (s) contiene -0 (codice di errore 4100).

Esempio di programma

SQR

Se SM400 è attivo, il programma seguente calcola la radice quadrata del valore BCD a 4 cifre specificato da X20 a X2F. Il risultato viene memorizzato in D0 e D1.



- 1 Conversione in formato BIN
- 2 Conversione in formato virgola mobile
- 3 Calcolo radice quadrata
- 4 Valore BCD
- 5 Valore binario
- 6 Valore in virgola mobile (numero reale)
- 7 Valore in virgola mobile (numero reale)

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.12.10 EXP, EXPP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

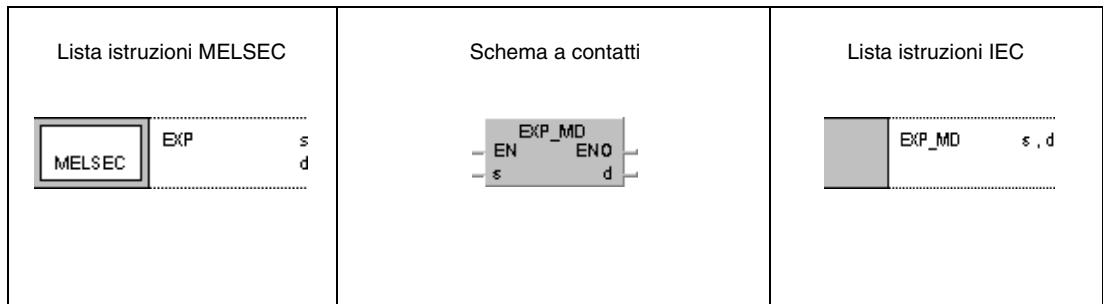
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

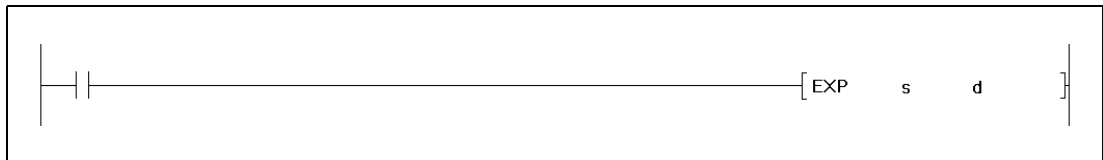
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Indice file Zn	Costanti E			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	●	●	—	●	—	SM0	3
d	—	●	●	—	●	●	—	—	—		

GX IEC Developer



GX Developer

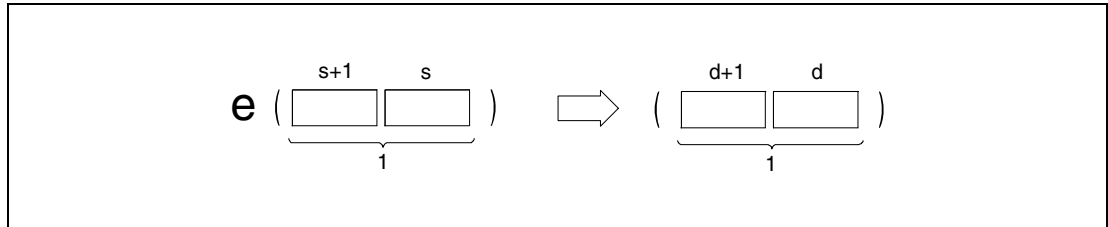


Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene il valore per l'istruzione EXP	Numero reale
d	Indirizzo del primo operando che memorizza il risultato dell'operazione.	

Funzioni Valore in virgola mobile come esponente della base e**EXP** Esponente di e

L'istruzione EXP calcola il valore dell'esponente corrispondente alla base e dal valore in virgola mobile contenuto in s e s+1. Il risultato viene memorizzato in d e d+1.



¹ Valore in virgola mobile (numero reale)

Il calcolo si basa sulla costante di Eulero: "e = 2.718281828".

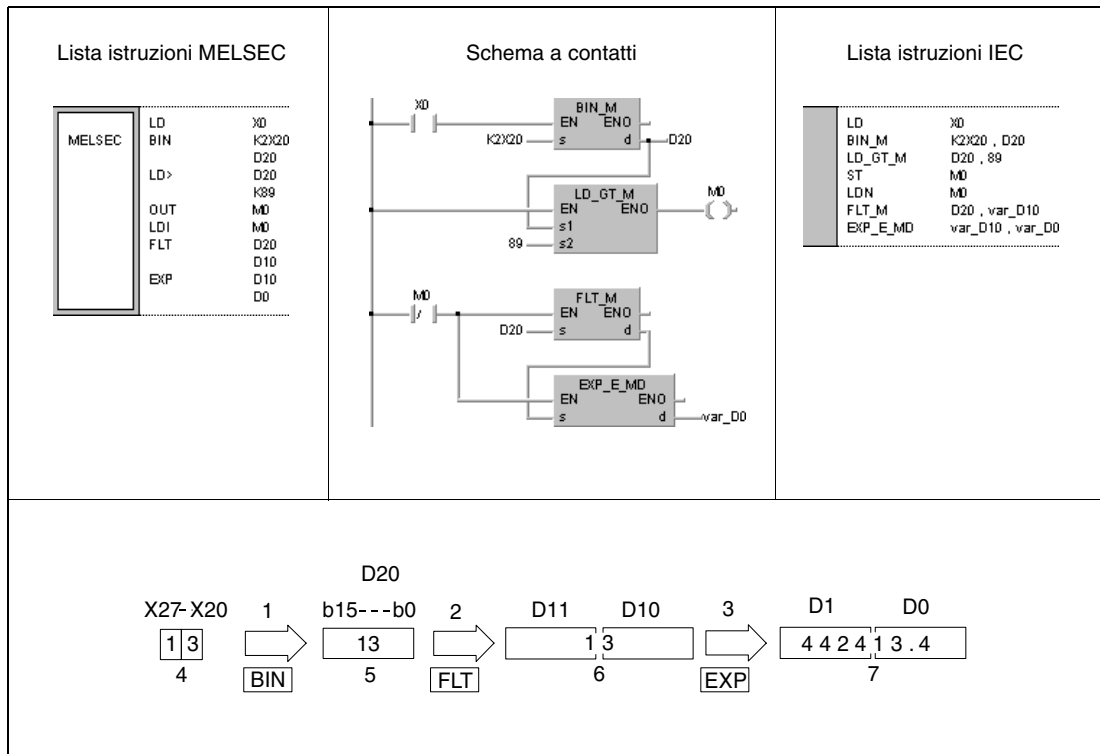
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il risultato del calcolo supera il campo di valori da 2^{-127} a 2^{129} (codice di errore 4100).
- Solo per la CPU Q4AR: Se SM707 è OFF e l'operando specificato (s) contiene -0 (codice di errore 4100).

Esempio di programma EXP

Il programma seguente calcola il risultato della funzione esponenziale in base e con il valore BCD a 2 cifre contenuto da X20 a X27. Il risultato viene memorizzato in D0 e D1 come valore in virgola mobile.



- 1 Conversione in formato BIN
- 2 Conversione in formato virgola mobile
- 3 Calcolo esponenziale
- 4 Valore BCD
- 5 Valore binario
- 6 Valore in virgola mobile (numero reale)
- 7 Valore in virgola mobile (numero reale)

NOTA

Il risultato del calcolo non deve superare $2^{129} \ln = 89.41598$. Se il valore BCD è maggiore di 90, viene restituito un messaggio di errore da SM0.

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.12.11 LOG, LOGP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

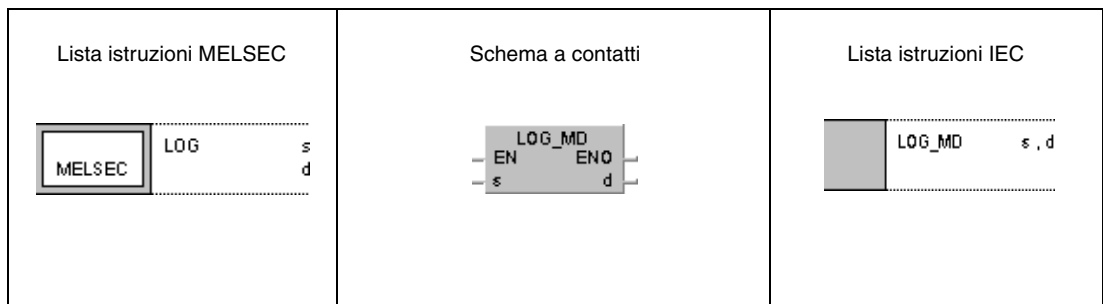
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

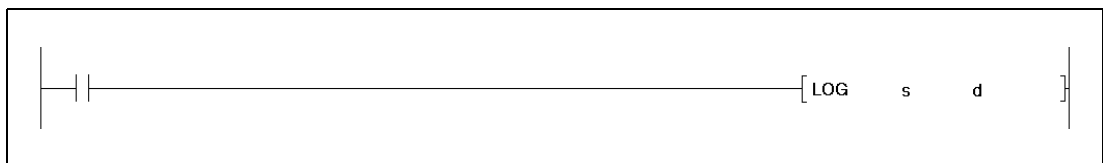
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Indice file Zn	Costanti E			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	●	●	—	●	—	SM0	3
d	—	●	●	—	●	●	—	—	—		

GX IEC Developer



GX Developer

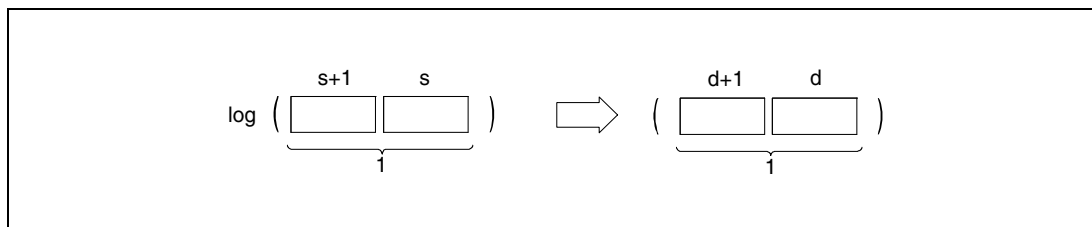


Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene il valore per l'istruzione LOG.	Numero reale
d	Indirizzo del primo operando che memorizza il risultato dell'operazione.	

Funzioni **Calcolo del logaritmo (ln) da valori in virgola mobile****LOG** **Calcolo del logaritmo (ln)**

L'istruzione LOG calcola il valore del logaritmo naturale relativo al valore in virgola mobile contenuto in s e s+1. Il risultato viene memorizzato in d e d+1.



¹ Valore in virgola mobile (numero reale)

In s e s+1 possono essere contenuti solo valori positivi. Valori negativi non possono essere elaborati.

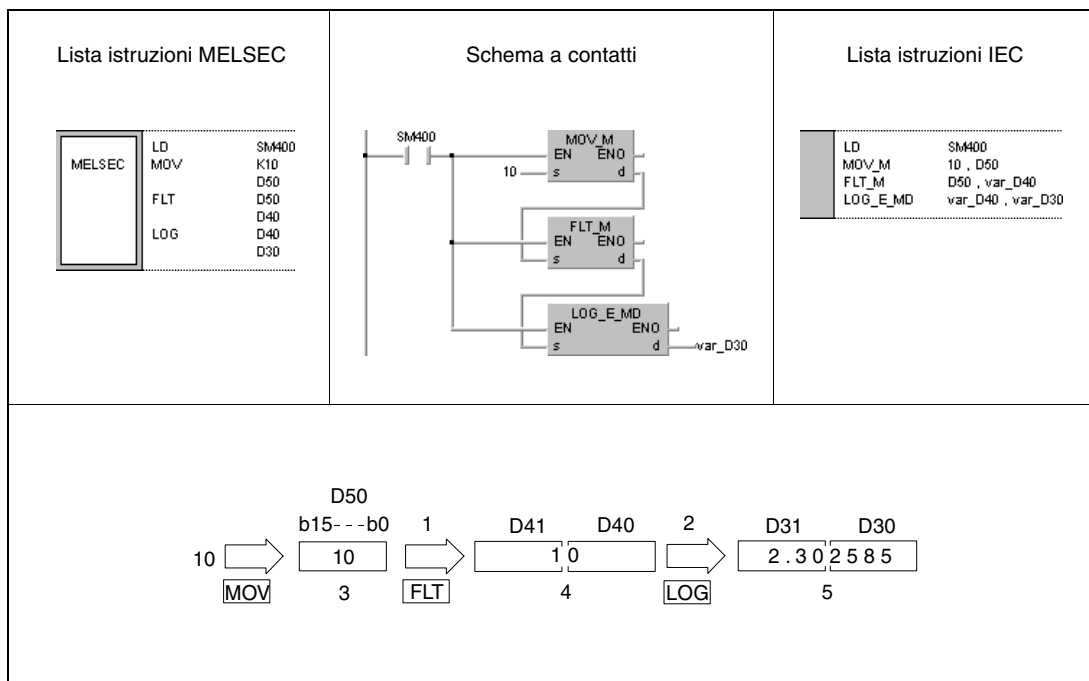
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il valore specificato in s è negativo (codice di errore 4100).
- Il risultato del calcolo supera il campo di valori da 2^{-127} a 2^{129} (codice di errore 4100).
- Solo per la CPU Q4AR: Se SM707 è OFF e l'operando specificato (s) contiene -0 (codice di errore 4100).

Esempio di programma LOG

Il programma seguente calcola il logaritmo naturale del valore 10. Il risultato viene memorizzato in D30 e D31.



- 1 Conversione in formato virgola mobile
- 2 Calcolo del logaritmo
- 3 Valore binario
- 4 Valore in virgola mobile (numero reale)
- 5 Valore in virgola mobile (numero reale)

NOTA L'istruzione LOG calcola il logaritmo naturale (in base e). La formula seguente converte il logaritmo naturale nel logaritmo in base 10:

$$\log_{10} X = 0.43429 \times \log_e X$$

NOTA Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.12.12 RND, RNDP, SRND, SRNDP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

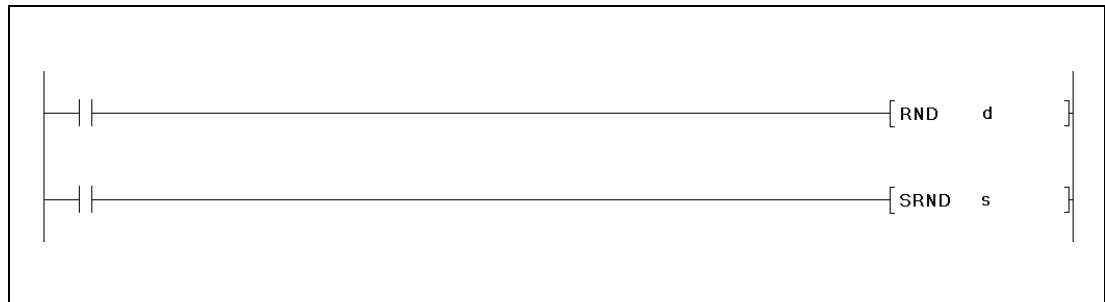
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Indice file Zn	Costanti K, H, 16#			Altro
	Bit	Word		Bit	Word						
s					●			—	—	—	3

GX IEC Developer

<p>Lista istruzioni MELSEC</p> <table border="1" style="margin-left: 20px;"> <tr> <td style="border: 1px solid black; padding: 2px;">MELSEC</td> <td style="padding: 2px;">RND</td> <td style="padding: 2px;">d</td> </tr> <tr> <td></td> <td style="padding: 2px;">SRND</td> <td style="padding: 2px;">s</td> </tr> </table>	MELSEC	RND	d		SRND	s	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p> <table border="1" style="margin-left: 20px;"> <tr> <td style="padding: 2px;">RND_M</td> <td style="padding: 2px;">d</td> </tr> <tr> <td style="padding: 2px;">SRND_M</td> <td style="padding: 2px;">s</td> </tr> </table>	RND_M	d	SRND_M	s
MELSEC	RND	d										
	SRND	s										
RND_M	d											
SRND_M	s											

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
d	Indirizzo del primo operando che memorizza il dato random.	BIN 16-bit
s	Serie dei valori random o indirizzo del primo operando che contiene questo dato.	

Funzioni **Generazione e aggiornamento serie valori random**

RND **Generazione valore random**

L'istruzione RND genera un valore random nel campo da 0 a 32767 e lo memorizza in d.

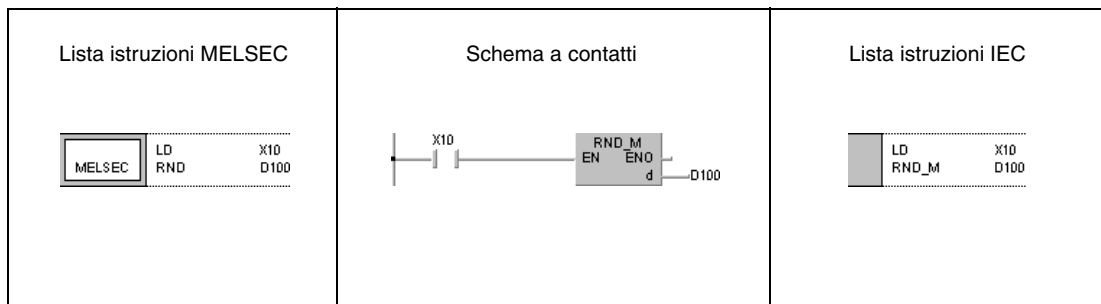
SRND **Aggiorna serie valori random**

L'istruzione SRND aggiorna la serie di valori random memorizzati in s.

Programma di esempio 1

RND

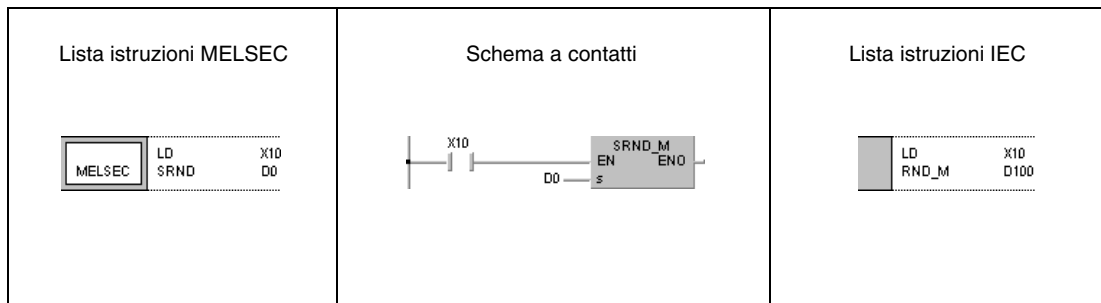
Mentre X10 è attivo, il programma seguente memorizza il valore random generato in D100.



Programma di esempio 2

SRND

Mentre X10 è attivo, il programma seguente aggiorna la serie di valori random in D0.



7.12.13 BSQR, BSQRP, BDSQR, BDSQRP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²


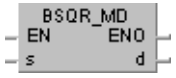
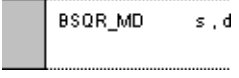
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

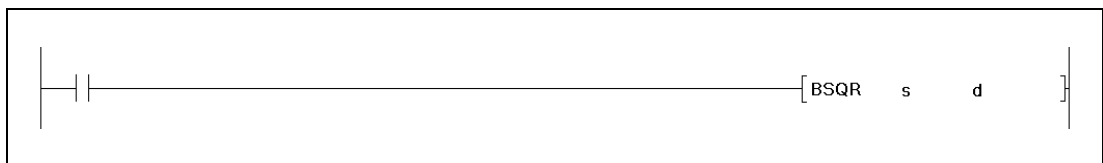
Operandi MELSEC Q

	Operandi utilizzabili							Errore	Numero di passi		
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Indice file Zn			Costanti K, H, (16#)	Altro
	Bit	Word		Bit	Word						
s					●			●	—	SM0	3
d					●			—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
---	---	--

GX Developer



Variabili

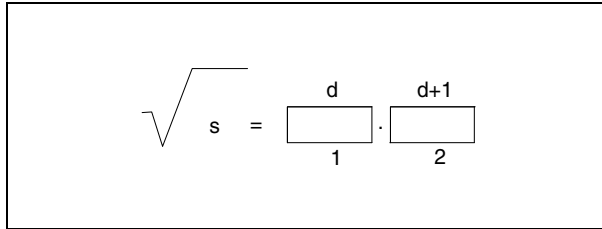
Dati impostati	Significato	Tipo dati
s	Dato per il calcolo della radice quadrata o indirizzo del primo operando che lo contiene.	BCD 4-/ 8-cifre
d	Indirizzo del primo operando che memorizza il risultato dell'estrazione di radice.	BCD 4-cifre

Funzioni

Calcolo radice quadrata di dati BCD a 4 o 8 cifre

BSQR Calcolo radice quadrata di dato BCD a 4 cifre

L'istruzione BSQR calcola la radice quadrata di s e memorizza il risultato in d e d+1.



¹ Parte intera
² Posti decimali

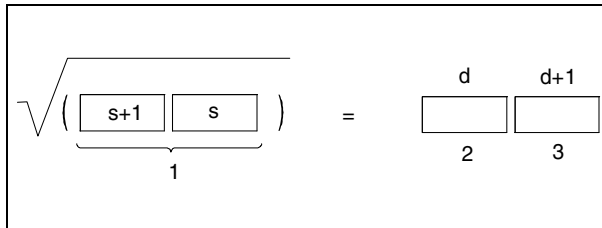
Il dato contenuto in s deve essere un valore BCD con un massimo di 4 cifre. Il campo di valori da 0 a 9999 non deve essere superato.

Il risultato calcolato, memorizzato in e d d+1, non deve superare il campo da 0 a 9999.

Il risultato viene calcolato con una precisione di 5 cifre ed arrotondato ad un valore da 4 cifre.

BDSQR Calcolo radice quadrata di dato BCD a 8 cifre

L'istruzione BDSQR calcola la radice quadrata di s e s+1 e memorizza il risultato in d e d+1.



¹Dato a doppia parola
²Parte intera
³Posti decimali

Il dato contenuto in s e s+1 deve essere un valore BCD con un massimo di 8 cifre. Il campo di valori da 0 a 99999999 non deve essere superato.

Il risultato calcolato, memorizzato in e d d+1, non deve superare il campo da 0 a 9999.

Il risultato viene calcolato con una precisione di 5 cifre ed arrotondato ad un valore da 4 cifre.

Errori di esecuzione

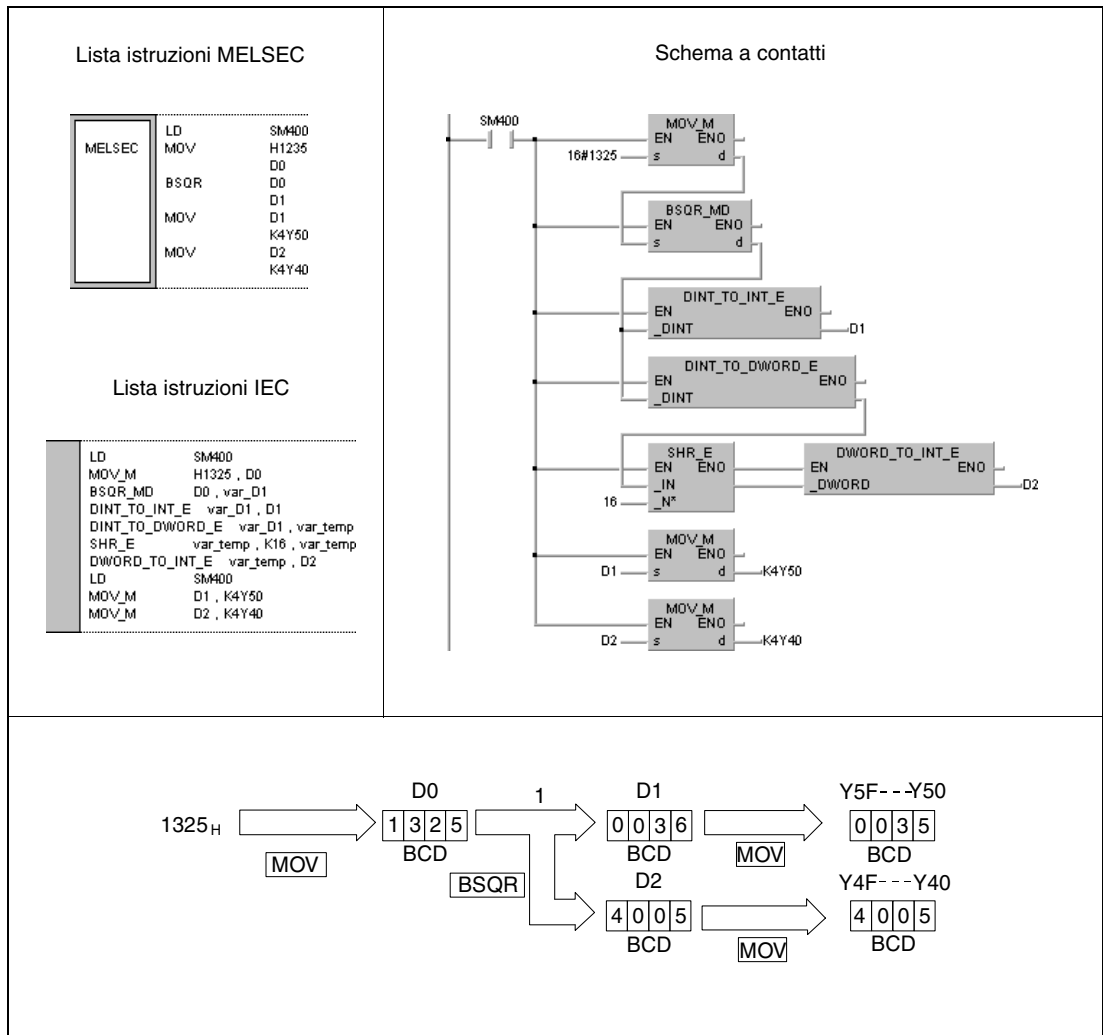
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il dato contenuto in s (s+1) non è un dato BCD (codice di errore 4100).

Programma di esempio 1

BSQR

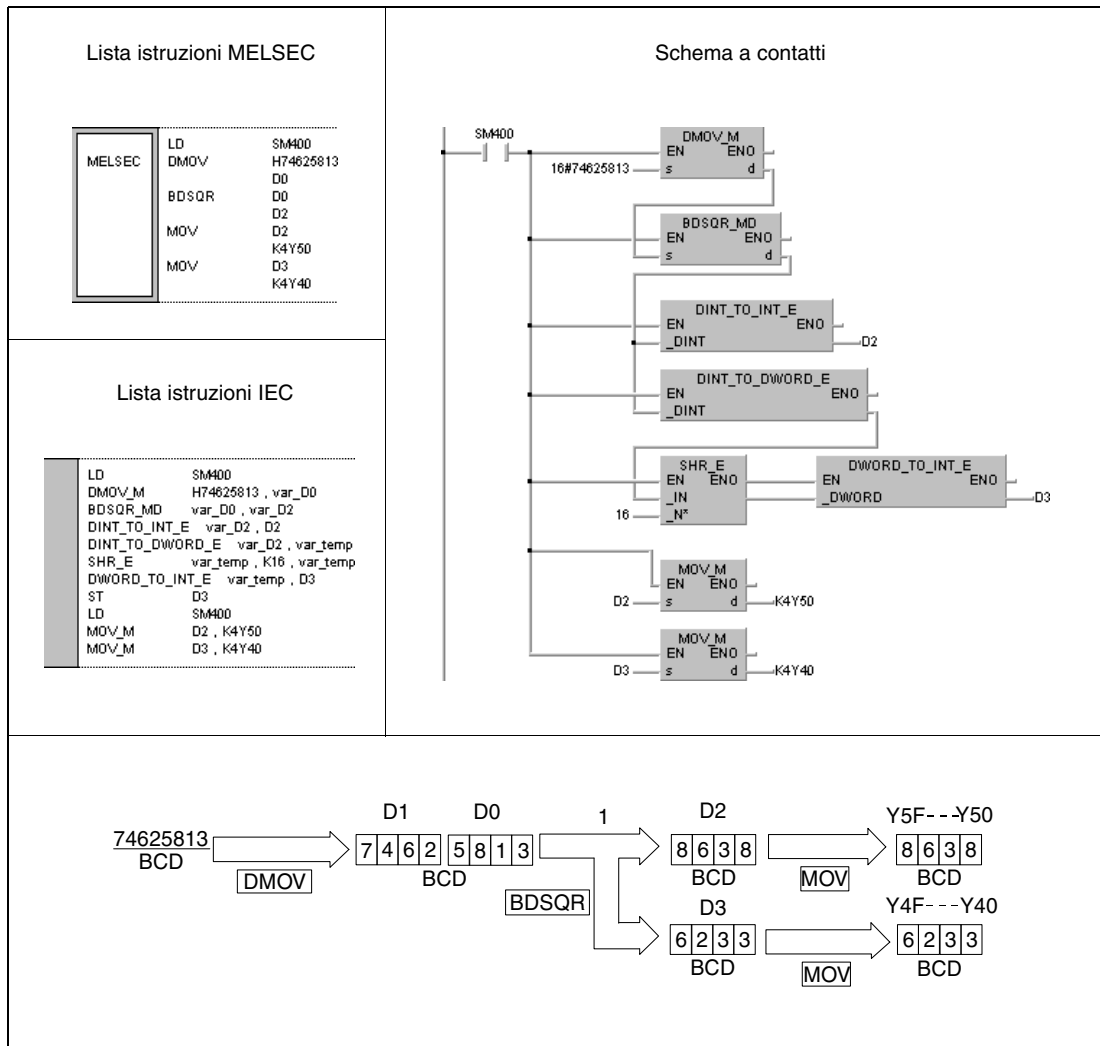
Se SM400 è attivo, il programma che segue calcola la radice quadrata del valore BCD 1325 ed invia la parte intera del risultato come valore BCD a 4 cifre sulle uscite da Y50 a Y5F. Le cifre decimali sono inviate come valore BCD sulle uscite da Y40 a Y4F.



¹ Calcolo radice quadrata

Programma di esempio 2 BDSQR

Se SM400 è attivo, il programma che segue calcola la radice quadrata del valore BCD 74625813 ed invia la parte intera del risultato come valore BCD a 4 cifre sulle uscite da Y50 a Y5F. Le cifre decimali sono inviate come valore BCD sulle uscite da Y40 a Y4F.



¹ Calcolo radice quadrata

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provocano messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.12.14 BSIN, BSINP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

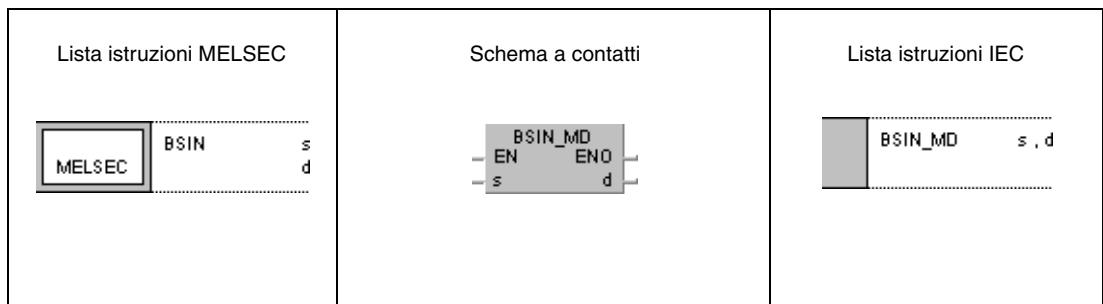
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

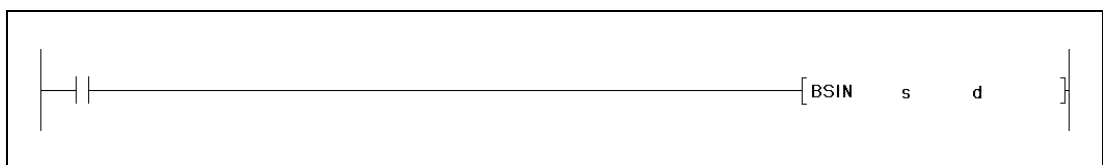
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Indice file Zn	Costanti K, H, (16#)			Altro
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	●	—	SM0	3	
d	—	●	●	—	—	—	—	—			

GX IEC Developer



GX Developer

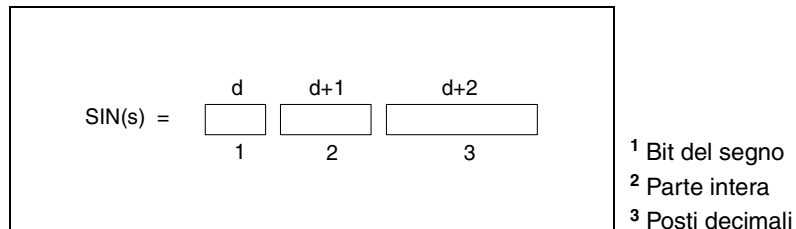


Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene il dato dell'angolo per l'istruzione BSIN (seno).	Valore BCD a 4 cifre
d	Indirizzo del primo operando che memorizza il risultato del calcolo.	

Funzioni **Calcolo seno da dato BCD****BSIN** **Calcolo del seno**

L'istruzione BSIN calcola il valore del seno dell'angolo contenuto in s. Il segno del risultato viene memorizzato in d. Il valore del risultato viene memorizzato in d+1 e d+2.



Il valore s deve essere un valore BCD nel campo da 0° a 360°.

Il segno del risultato in d è 0 se il valore è positivo e 1 se il valore è negativo.

Il risultato in d+1 e d+2 può variare nel campo da -1,000 a 1,000 in formato BCD.

Il calcolo del risultato viene arrotondato alla quinta cifra.

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

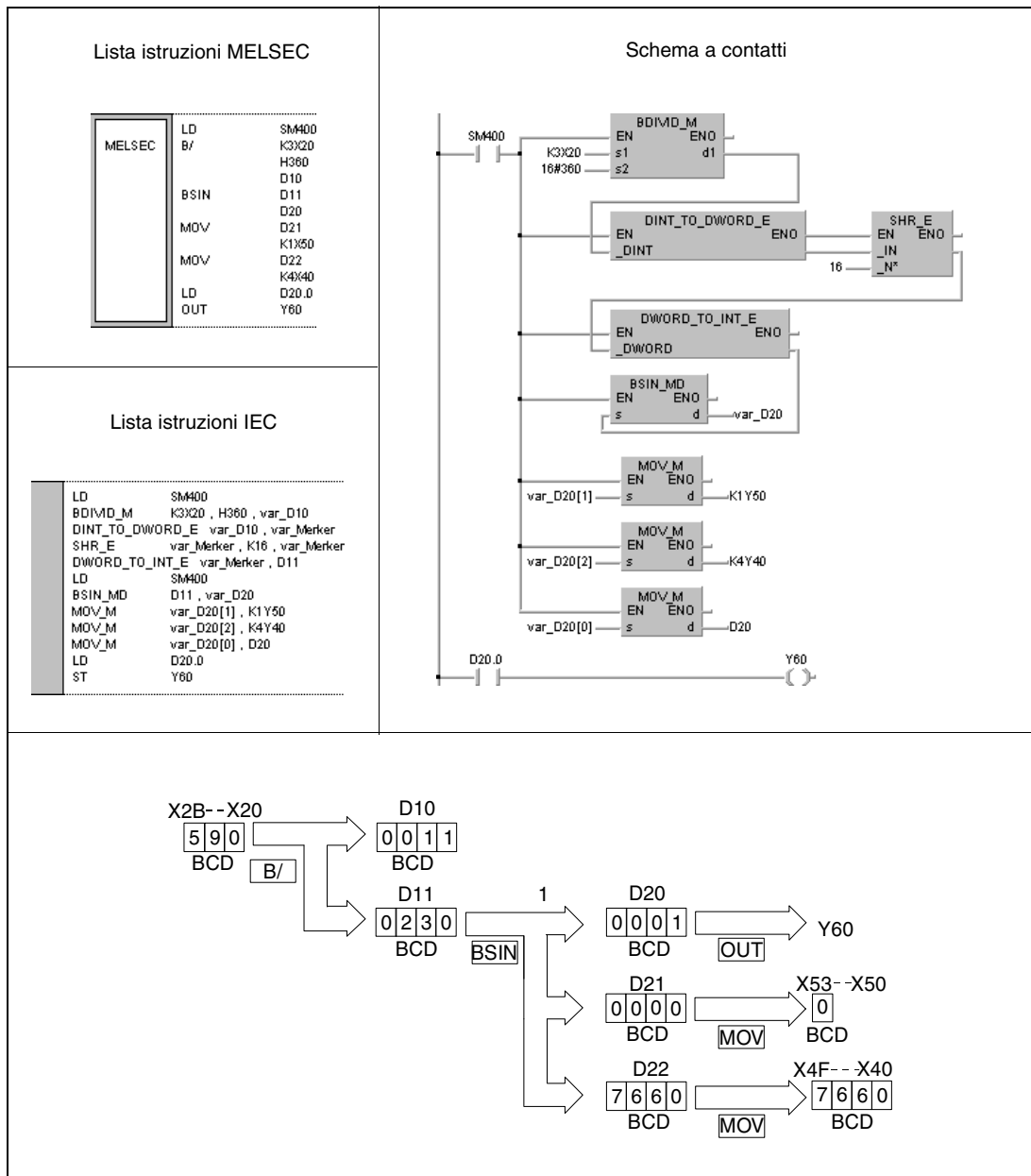
- Il dato contenuto in s non è un dato BCD (codice di errore 4100).
- Il dato specificato in s supera il campo di valori da 0° a 360° (codice di errore 4100).

Esempio di programma BSIN

Se SM400 è attivo, il programma seguente calcola il valore del seno relativo al valore BCD a 3 cifre specificato da X20 a X2B. Se il valore da X20 a X2B supera 360 (gradi), viene corretto per riportarlo nel campo ammesso da 0° a 360°.

Il segno viene inviato su Y60. La parte intera viene inviata sulle uscite da Y50 a Y53 come valore BCD a 1 cifra.

Le cifre decimali sono inviate come valore BCD a 4 cifre sulle uscite da Y40 a Y4F.



¹ Calcolo del seno

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.12.15 BCOS, BCOSP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

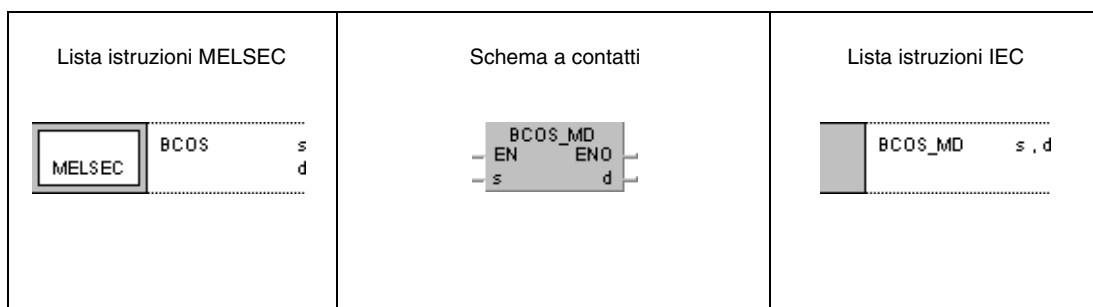
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

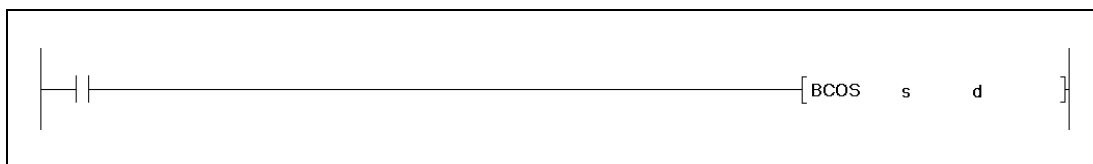
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Indice file Zn	Costanti K, H, (16#)			Altro
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	●	—	SM0	3	
d	—	●	●	—	—	—	—	—			

GX IEC Developer



GX Developer

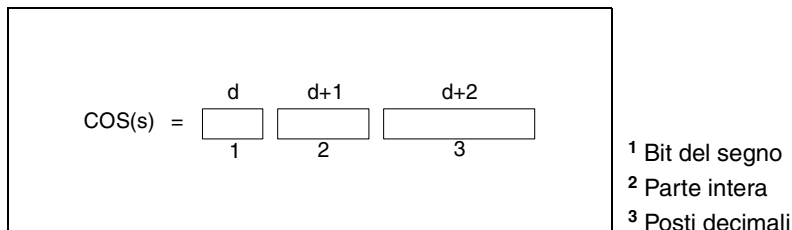


Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene il dato dell'angolo per l'istruzione BCOS (coseno).	Valore BCD a 4 cifre
d	Indirizzo del primo operando che memorizza il risultato del calcolo.	

Funzioni **Calcolo coseno da dato BCD****BCOS** **Calcolo del coseno**

L'istruzione BCOS calcola il valore del coseno dell'angolo contenuto in s. Il segno del risultato viene memorizzato in d. Il valore del risultato viene memorizzato in d+1 e d+2.



Il valore s deve essere un valore BCD nel campo da 0° a 360°.

Il segno del risultato in d è 0 se il valore è positivo e 1 se il valore è negativo.

Il risultato in d+1 e d+2 può variare nel campo da -1,000 a 1,000 in formato BCD.

Il calcolo del risultato viene arrotondato alla quinta cifra.

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il dato contenuto in s non è un dato BCD (codice di errore 4100).
- Il dato specificato in s supera il campo di valori da 0° a 360° (codice di errore 4100).

Esempio di programma BCOS

Se SM400 è attivo, il programma seguente calcola il valore del coseno relativo al valore BCD a 3 cifre specificato da X20 a X2B. Se il valore da X20 a X2B supera 360 (gradi), viene corretto per riportarlo nel campo ammesso da 0° a 360°.

Il segno viene inviato su Y60. La parte intera viene inviata sulle uscite da Y50 a Y53 come valore BCD a 1 cifra.

Le cifre decimali sono inviate come valore BCD a 4 cifre sulle uscite da Y40 a Y4F.

Lista istruzioni MELSEC

MELSEC	LD	SM400
	B/	K3X20
		H360
		D10
	BCOS	D11
		D20
	MOV	D21
		K1X50
	MOV	D22
		K4X40
	LD	D20.0
	OUT	Y60

Lista istruzioni IEC

```

LD      SM400
BDIVD_M K3X20, H360, var_D10
DINT_TO_DWORD_E var_D10, var_Marker
SHR_E    var_Marker, K16, var_Marker
DWORD_TO_INT_E var_Marker, D11
LD      SM400
BCOS_MD D11, var_D20
MOV_M   var_D20[0], K1Y50
MOV_M   var_D20[1], K4Y40
MOV_M   var_D20[2], D20
LD      D20.0
ST      Y60
    
```

Schema a contatti

¹ Calcolo del coseno

NOTA Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.12.16 BTAN, BTANP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²


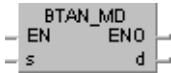
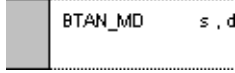
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

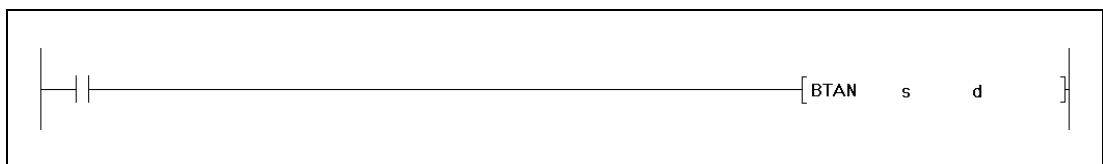
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Indice file Zn	Costanti K, H, (16#)			Altro
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	●	—	SM0	3	
d	—	●	●	—	—	—	—	—			

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
---	---	--

GX Developer

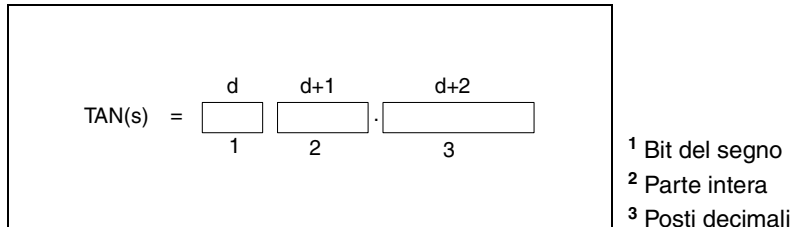


Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene il dato dell'angolo per l'istruzione BTAN (tangente).	Valore BCD a 4 cifre
d	Indirizzo del primo operando che memorizza il risultato del calcolo.	

Funzioni **Calcolo tangente da dato BCD****BTAN** **Calcolo della tangente**

L'istruzione BTAN calcola il valore della tangente dell'angolo contenuto in s. Il segno del risultato viene memorizzato in d. Il valore del risultato viene memorizzato in d+1 e d+2.



Il valore s deve essere un valore BCD nel campo da 0° a 360°.

Il segno del risultato in d è 0 se il valore è positivo e 1 se il valore è negativo.

Il risultato in d+1 e d+2 può variare nel campo da -57.2900 a 57.2900 in formato BCD.

Il calcolo del risultato viene arrotondato alla quinta cifra.

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

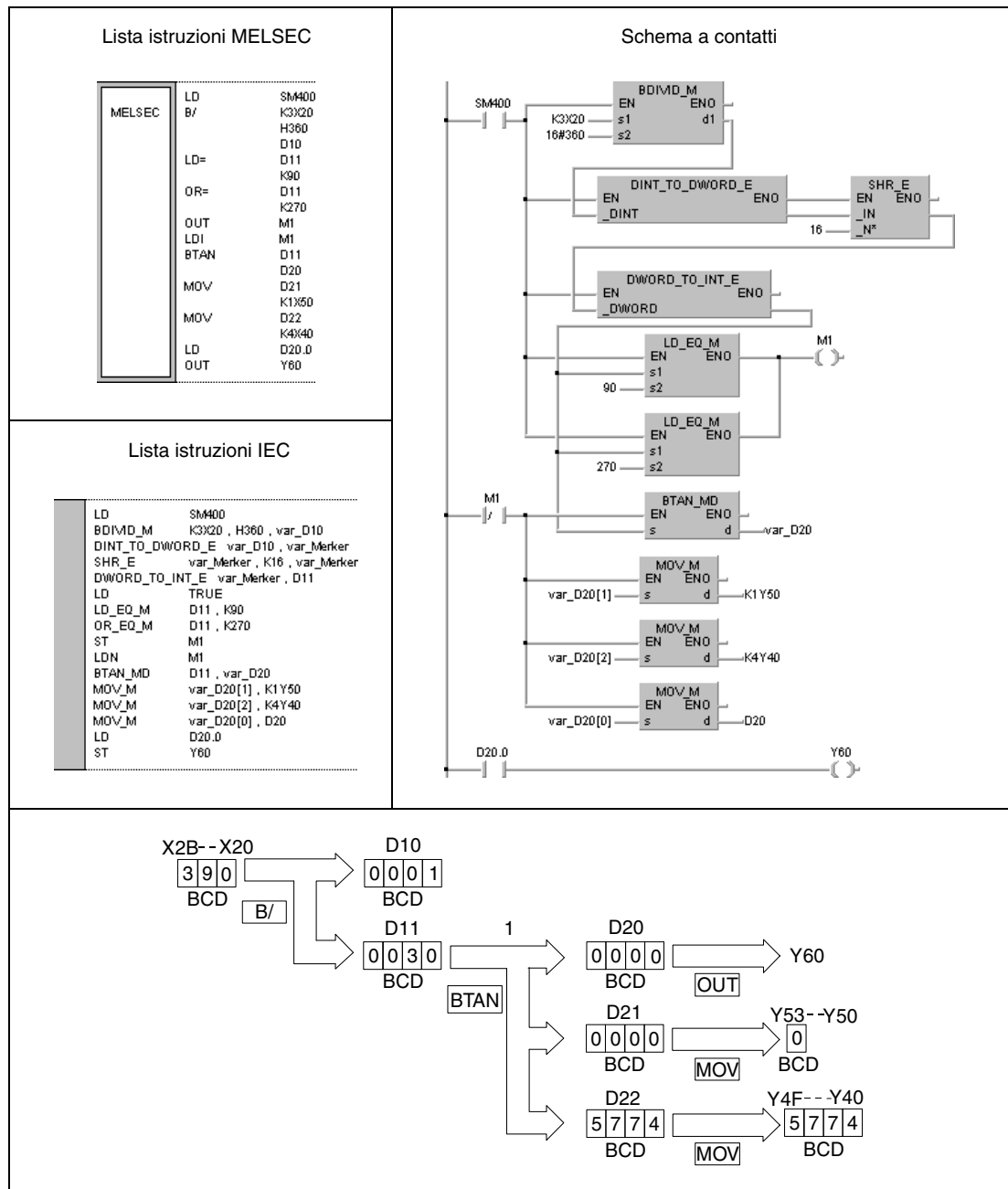
- Il dato contenuto in s non è un dato BCD (codice di errore 4100).
- Il dato specificato in s supera il campo di valori da 0° a 360° (codice di errore 4100).
- Il valore specificato in s è 90° o 270° (codice di errore 4100).

Esempio di programma BTAN

Se SM400 è attivo, il programma seguente calcola il valore della tangente relativo al valore BCD a 3 cifre specificato da X20 a X2B. Se il valore da X20 a X2B supera 360 (gradi), viene corretto per riportarlo nel campo ammesso da 0° a 360°.

Il segno viene inviato su Y60. La parte intera viene inviata sulle uscite da Y50 a Y53 come valore BCD a 1 cifra.

Le cifre decimali sono inviate come valore BCD a 4 cifre sulle uscite da Y40 a Y4F.



¹ Calcolo della tangente

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.12.17 BASIN, BASINP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

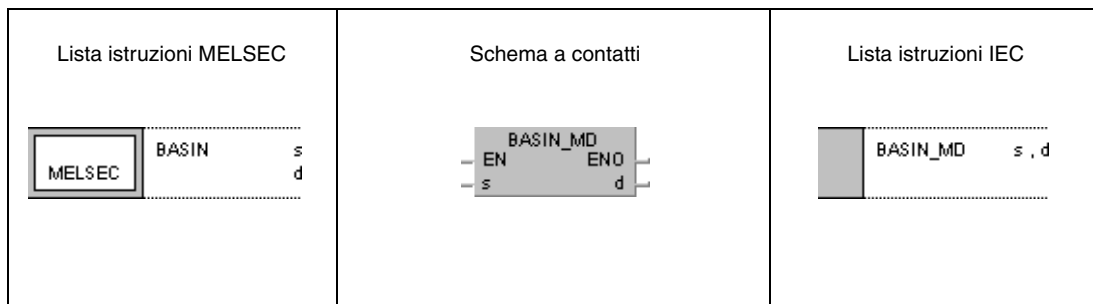
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

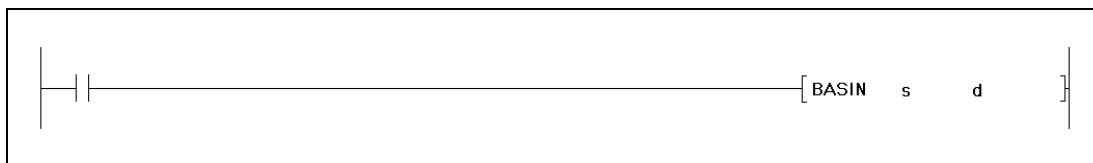
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Indice file Zn	Costanti K, H, (16#)			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	—	SM0	3	
d	●	●	●	●	●	●	—	—			

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene il valore del seno per l'istruzione BASIN. (arcoseno).	Valore BCD a 4 cifre
d	Indirizzo del primo operando che memorizza il risultato del calcolo.	

Funzioni **Calcolo arcoseno da dato BCD****BASIN** **Calcolo arcoseno**

L'istruzione BASIN calcola il valore dell'angolo relativo al valore del seno contenuto in s, s+1 e s+2. Il risultato viene memorizzato in d.

$$\text{SIN}^{-1} = \left(\begin{array}{c} \text{s} \\ \boxed{} \\ 1 \end{array} \begin{array}{c} \text{s+1} \\ \boxed{} \\ 2 \end{array} \cdot \begin{array}{c} \text{s+2} \\ \boxed{} \\ 3 \end{array} \right) = d$$

¹Bit del segno

²Parte intera

³Posti decimali

Il segno del risultato in s è 0 se il valore è positivo e 1 se il valore è negativo.

La parte intera prima del punto decimale ed il numero di decimali devono essere valori BCD nel campo da 0 a 10000.

Il valore del risultato in d deve essere un valore BCD nel campo da 0° a 90° o da 270° a 360°.

Il calcolo del risultato viene arrotondato alla quinta cifra.

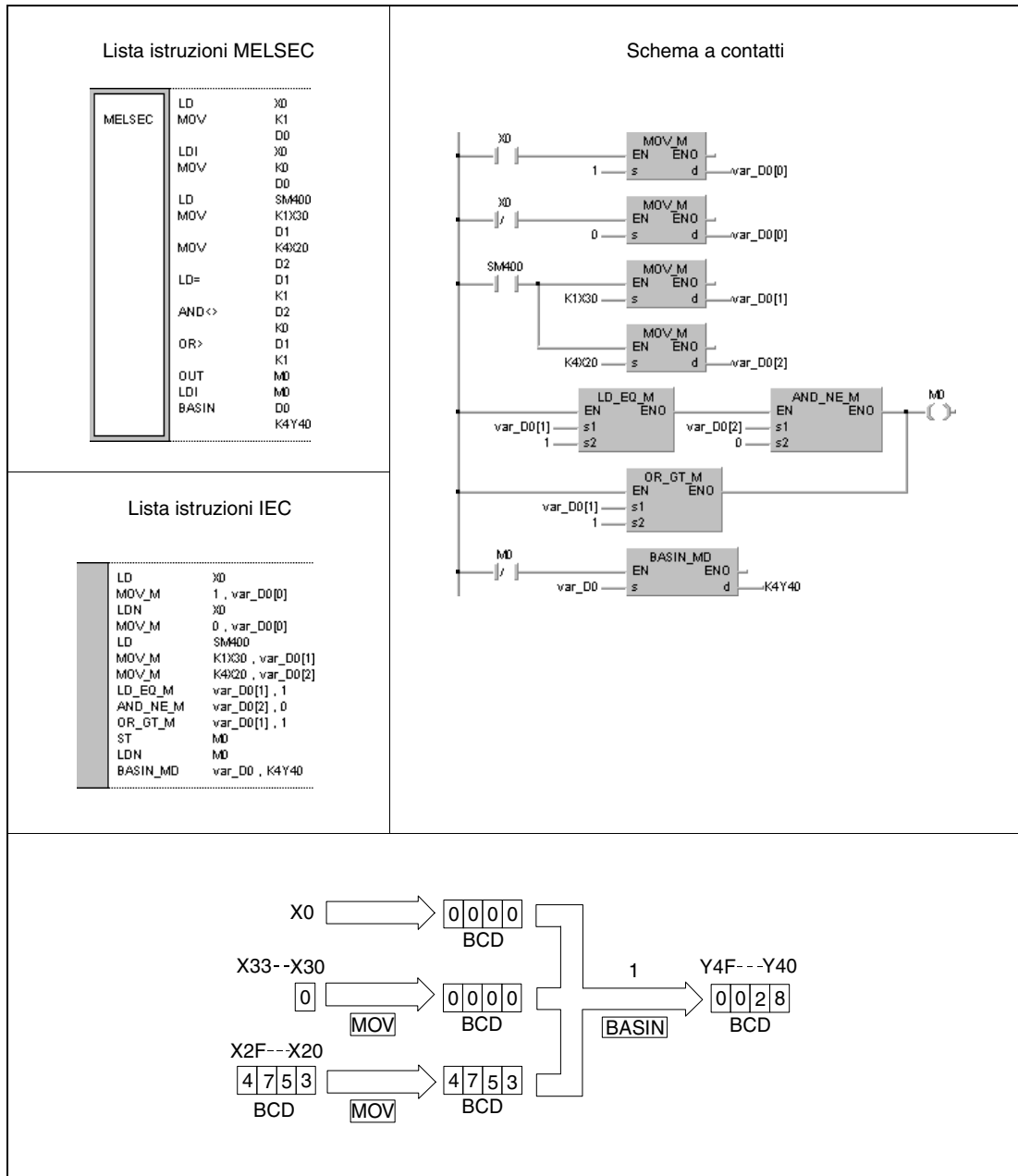
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il dato contenuto da s a s+2 non è un dato BCD (codice di errore 4100).
- Il dato specificato da s a s+2 supera il campo ammesso da -1.0000 a 1.0000 (codice di errore 4100).

Esempio di programma **BASIN**

Se SM400 è attivo, il programma seguente calcola il valore dell'arcoseno in base al bit del segno su X0 (1 = positivo, 0 = negativo), la parte intera BCD a 1 cifra da X30 a X33 e le cifre decimali come valore BCD a 4 cifre, da X20 a X2F. Il valore dell'angolo risultante viene inviato come valore BCD a 4 cifre, sulle uscite da Y40 a Y4F.



¹ Calcolo arcoseno

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.12.18 BACOS, BACOSP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

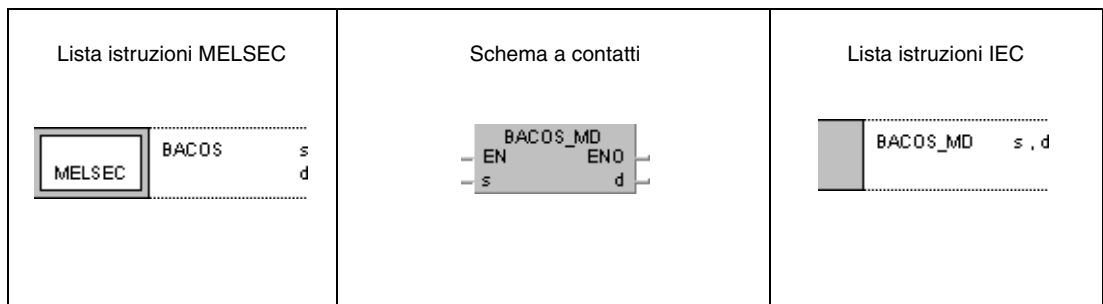
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

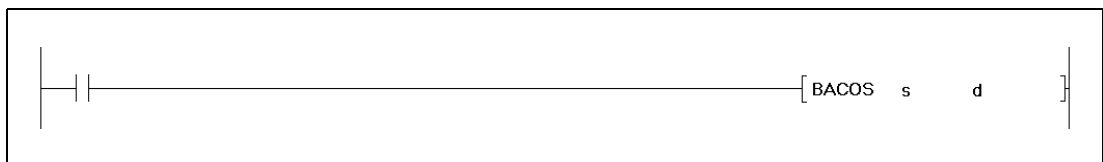
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Indice file Zn	Costanti K, H, (16#)			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	—	—	SM0	3
d	●	●	●	●	●	●	—	—			

GX IEC Developer



GX Developer

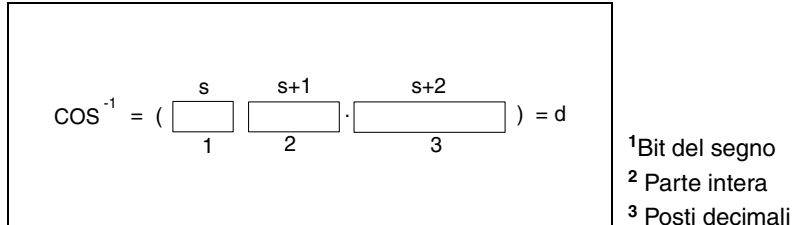


Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene il valore del coseno per l'istruzione BACOS. (arcocoseno).	Valore BCD a 4 cifre
d	Indirizzo del primo operando che memorizza il risultato del calcolo.	

Funzioni **Calcolo arcocoseno da dato BCD****BACOS** **Calcolo arcocoseno**

L'istruzione BACOS calcola il valore dell'angolo relativo al valore del coseno contenuto in s, s+1 e s+2. Il risultato viene memorizzato in d.



Il segno del risultato in s è 0 se il valore è positivo e 1 se il valore è negativo.

La parte intera prima del punto decimale ed il numero di decimali devono essere valori BCD nel campo da 0 a 10000.

Il valore del risultato in d deve essere un valore BCD nel campo da 0° A 180.

Il calcolo del risultato viene arrotondato alla quinta cifra.

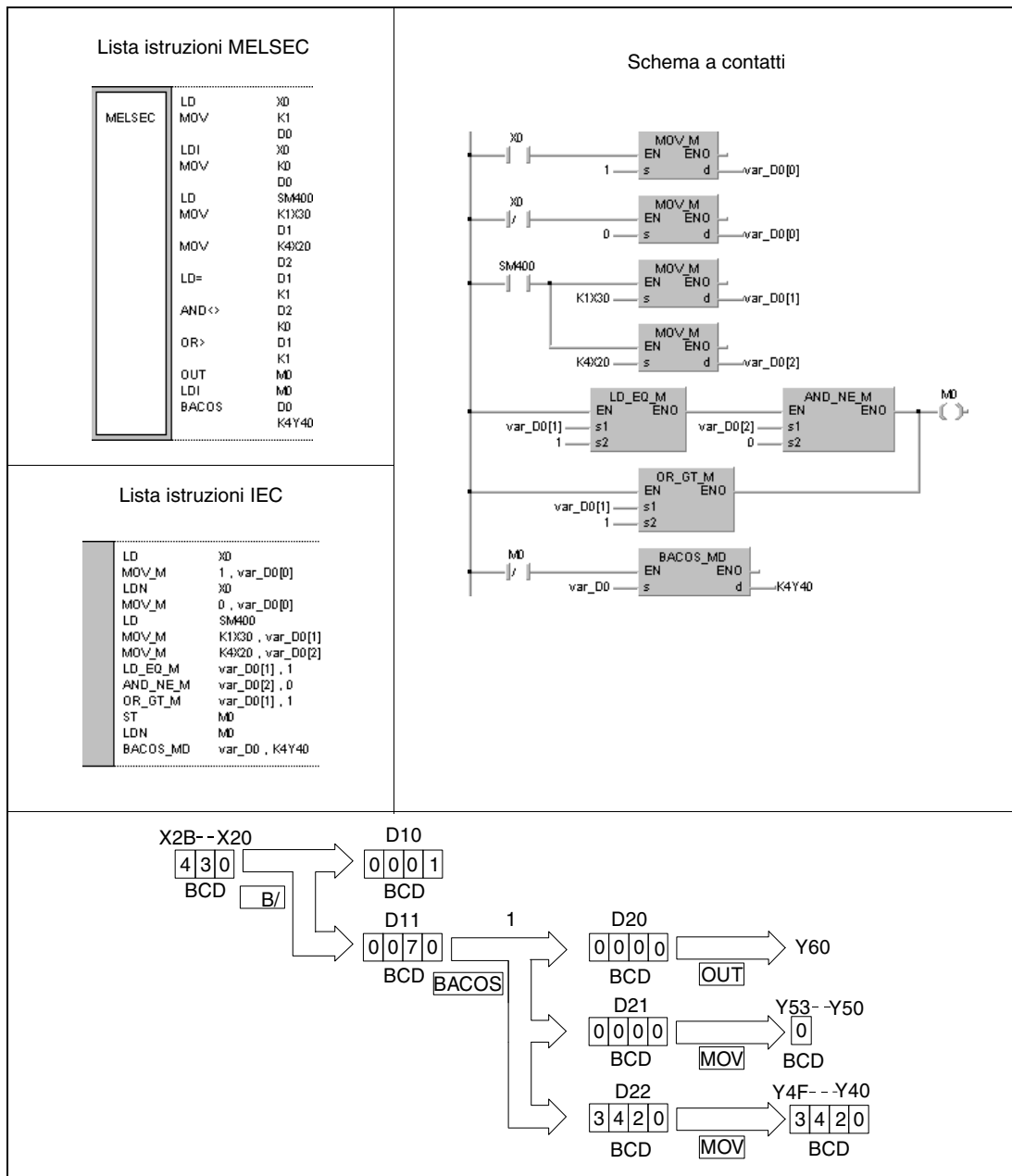
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il dato contenuto da s a s+2 non è un dato BCD (codice di errore 4100).
- Il dato specificato da s a s+2 supera il campo ammesso da -1000 a 1000 (codice di errore 4100).

Esempio di programma BACOS

Se SM400 è attivo, il programma seguente calcola il valore dell'arcocoseno in base al bit del segno su X0 (1 = positivo, 0 = negativo), la parte intera BCD a 1 cifra da X30 a X33 e le cifre decimali come valore BCD a 4 cifre, da X20 a X2F. Il valore dell'angolo risultante viene inviato come valore BCD a 4 cifre, sulle uscite da Y40 a Y4F.



¹ Calcolo arcocoseno

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.12.19 BATAN, BATANP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

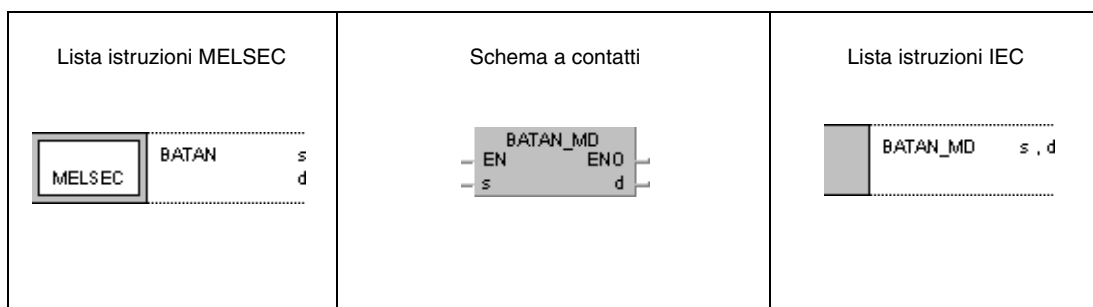
¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

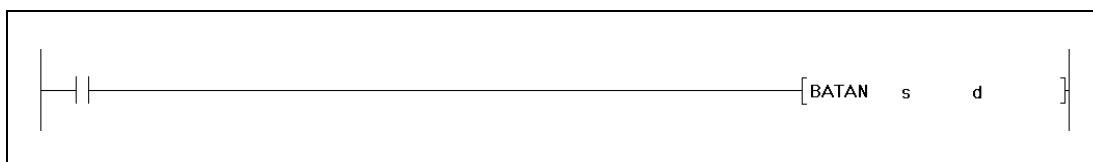
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Indice file Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	—	—	SM0	3
d	●	●	●	●	●	●	—	—			

GX IEC Developer



GX Developer



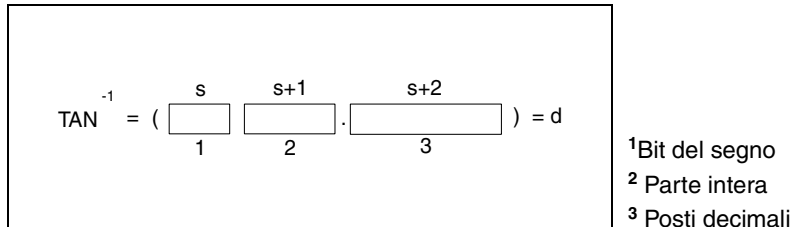
Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene il valore della tangente per l'istruzione BATAN (arcotangente).	Valore BCD a 4 cifre
d	Indirizzo del primo operando che memorizza il risultato del calcolo.	

Funzioni Calcolo arcotangente da dato BCD

BATAN Calcolo arcotangente

L'istruzione BATAN calcola il valore dell'angolo relativo al valore della tangente contenuto in s, s+1 e s+2. Il risultato viene memorizzato in d.



Il segno del risultato in s è 0 se il valore è positivo e 1 se il valore è negativo.

La parte intera prima del punto decimale ed il numero di decimali devono essere valori BCD nel campo da 0 a 99999999.

Il valore del risultato in d deve essere un valore BCD nel campo da 0° a 90° o 270° o da 270° a 360°.

Il calcolo del risultato viene arrotondato alla quinta cifra.

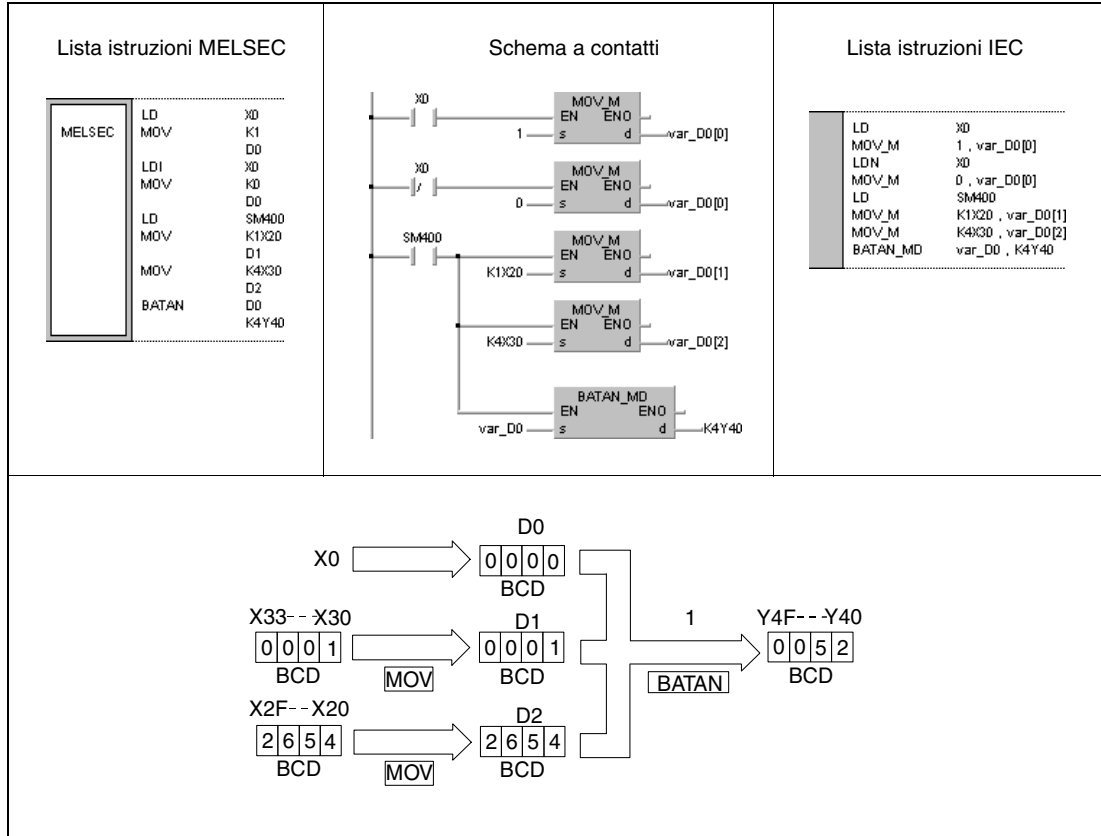
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il dato contenuto da s a s+2 non è un dato BCD (codice di errore 4100).

Esempio di programma BATAN

Se SM400 è attivo, il programma seguente calcola il valore dell'arcotangente in base al bit del segno su X0 (1 = positivo, 0 = negativo), la parte intera BCD a 1 cifra da X20 a X23 e le cifre decimali come valore BCD a 4 cifre, da X30 a X3F. Il valore dell'angolo risultante viene inviato come valore BCD a 4 cifre, sulle uscite da Y40 a Y4F.



¹ Calcolo arcotangente

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.13 Istruzioni per controllo dati

Le istruzioni per controllo dati interessano gli operandi di ingresso e uscita. I dati a 16 o 32 bit degli operandi di ingresso vengono inviati agli operandi di uscita dopo averne controllato i limiti superiore e inferiore, la zona morta e l'offset.

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Controllo limiti superiore e inferiore per dati BIN a 16/32-bit	LIMIT	LIMIT_MD
	LIMITP	LIMIT_P_MD
	DLIMIT	DLIMIT_MD
	DLIMITP	DLIMIT_P_MD
Controllo zona morta per dati BIN a 16/32-bit	BAND	BAND_MD
	BANDP	BAND_P_MD
	DBAND	DBAND_MD
	DBANDP	DBAND_P_MD
Controllo offset per dati BIN a 16/32-bit	ZONE	ZONE_MD
	ZONEP	ZONE_P_MD
	DZONE	DZONE_MD
	DZONEP	DZONE_P_MD

NOTA *Utilizzare istruzioni IEC con gli editor IEC*

7.13.1 LIMIT, LIMITP, DLIMIT, DLIMITP

CPU

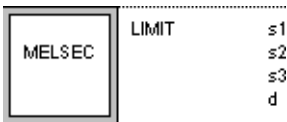
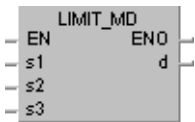
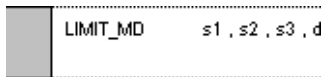
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	●

¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

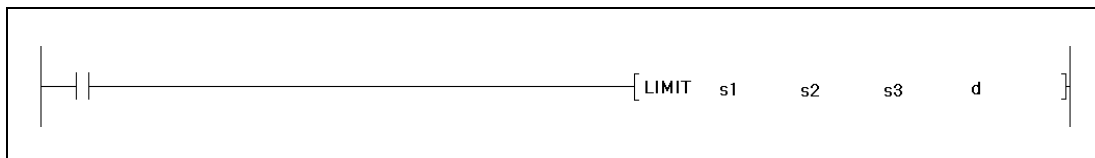
Operandi MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s1	●	●	●	●	●	●	●	●	—	SM0	5
s2	●	●	●	●	●	●	●	●	—		
s3	●	●	●	●	●	●	●	●	—		
d	●	●	●	●	●	●	●	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
---	---	--

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s1	Valore limite inferiore (valore di soglia uscita minima).	BIN 16-bit
s2	Valore limite superiore (valore di soglia uscita massima).	
s3	Valore d'ingresso da limitare.	
d	Indirizzo del primo operando che memorizza il valore di uscita limitato.	

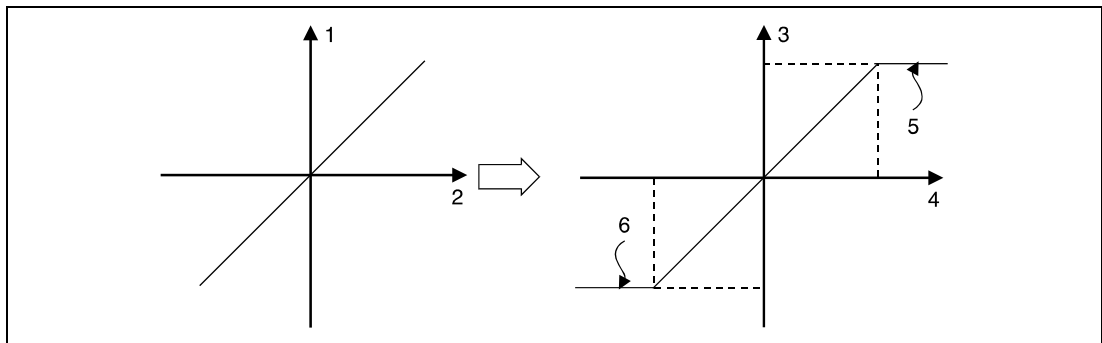
Funzioni Limitazione di valori di uscita per dati BIN a 16 e 32-bit**LIMIT Istruzione di limitazione per dati BIN a 16 bit.**

L'istruzione LIMIT controlla se il dato dell'operando specificato da s3 è compreso fra il limite inferiore specificato da s1 ed il limite superiore indicato da s2. A seconda del risultato del controllo, i valori vengono memorizzati nell'operando specificato da d come segue:

Se il valore del dato in s3 è minore del limite inferiore in s1, il limite inferiore viene memorizzato in d.

Se il valore del dato in s3 è maggiore del limite superiore in s2, il limite superiore viene memorizzato in d.

Se il valore del dato in s3 è compreso fra i valori limite inferiore e superiore, il valore del dato viene memorizzato in d.



- ¹ Valore di uscita
- ² Valore d'ingresso
- ³ Valore di uscita (d)
- ⁴ Valore d'ingresso (s3)
- ⁵ Valore limite superiore (s2)
- ⁶ Valore limite inferiore (s1)

I valori specificati da s1, s2 e s3 sono compresi nel campo da -32768 a 32767.

Se deve essere controllato solo il valore del limite superiore, il valore del limite inferiore in s1 deve essere impostato su -32768.

Se deve essere controllato solo il valore del limite inferiore, il valore del limite superiore in s1 deve essere impostato su 32767.

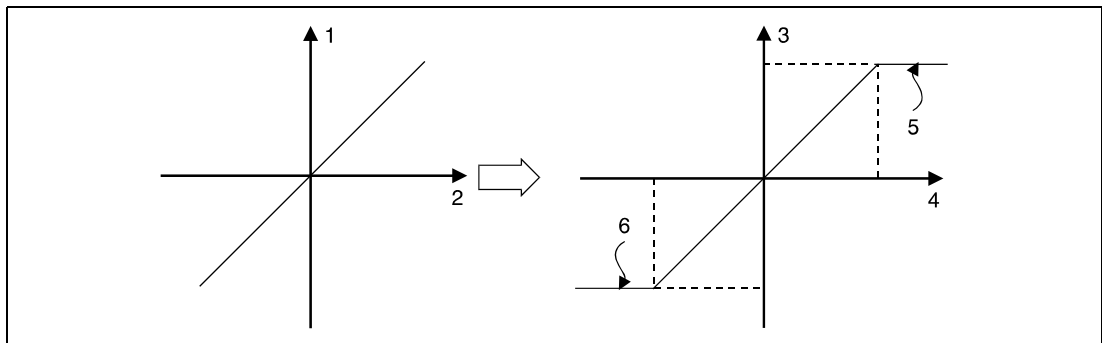
DLIMIT Istruzione di limitazione per dati BIN a 32 bit.

L'istruzione DLIMIT controlla se il dato contenuto negli operandi $s3$ e $(s3)+1$ è compreso fra il limite inferiore specificato da $s1$ e $(s1)+1$ ed il limite superiore specificato da $s2$ e $(s2)+1$. A seconda del risultato del controllo, i valori vengono memorizzati nel dispositivo indicato da d come segue:

Se il valore del dato in $s3$ e $(s3)+1$ è minore del limite inferiore in $s1$ e $(s1)+1$, il limite inferiore viene memorizzato in d e $d+1$.

Se il valore del dato in $s3$ e $(s3)+1$ è maggiore del limite superiore in $s1$ e $(s1)+1$, il limite superiore viene memorizzato in d e $d+1$.

Se il valore del dato in $s3$ e $(s3)+1$ è compreso fra i valori limite inferiore e superiore, il valore del dato viene memorizzato in d e $d+1$.



- ¹ Valore di uscita
- ² Valore d'ingresso
- ³ Valore di uscita ($d+1$, d)
- ⁴ Valore d'ingresso ($(s3)+1$, $s3$)
- ⁵ Valore limite superiore ($(s2)+1$, $s2$)
- ⁶ Valore limite inferiore ($(s2)+1$, $s2$)

I valori specificati da $s1$ e $(s1)+1$, $s2$ e $(s2)+1$, $s3$ e $(s3)+1$ devono essere nel campo da -2147483648 a 2147483647.

Se deve essere controllato solo il valore del limite superiore, il valore del limite inferiore in $s1$ e $(s1)+1$ deve essere impostato su -2147483648.

Se deve essere controllato solo il valore del limite inferiore, il valore del limite superiore in $s1$ e $(s1)+1$ deve essere impostato su 2147483647.

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il valore in $s1$ ($(s1)+1$) è superiore a quello in $s2$ ($(s2)+1$) (codice di errore 4100).

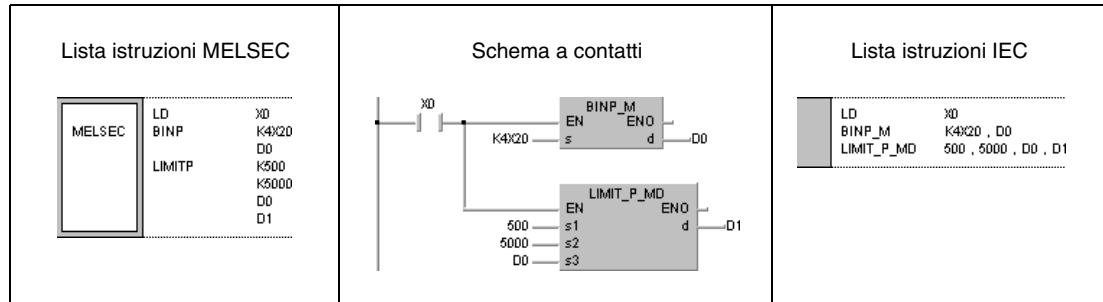
Programma di esempio 1 LIMITP

Con il fronte positivo di X0, il programma seguente controlla se il dato BCD contenuto da X20 a X2F è compreso fra il limite inferiore di 500 e il limite superiore di 5000. Il risultato dell'operazione di controllo viene memorizzato in D1.

Se il valore in D0 è superiore a 5000, il valore 5000 viene inserito in D1.

Se il valore in D0 è inferiore a 500, il valore 500 viene inserito in D1.

Se il valore è compreso fra 500 e 5000, il valore del dato viene inserito in D1.

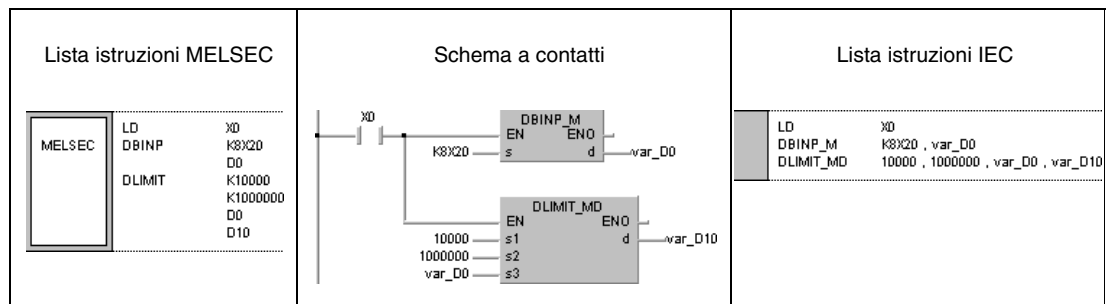
**Programma di esempio 2** DLIMIT

Con il fronte positivo di X0, il programma seguente controlla se il dato BCD contenuto da X20 a X3F è compreso fra il limite inferiore di 10000 e il limite superiore di 1000000. Il risultato dell'operazione di controllo viene memorizzato in D10 e D11.

Se il valore in D0 e D1 è superiore a 1000000, il valore 1000000 viene inserito in D10 e D11.

Se il valore in D0 e D1 è inferiore a 10000, il valore 10000 viene inserito in D10 e D11.

Se il valore è compreso fra 10000 e 1000000, il valore del dato viene inserito in D10 e D11.



7.13.2 BAND, BANDP, DBAND, DBANDP

CPU

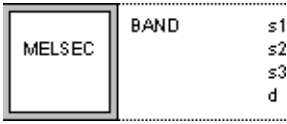
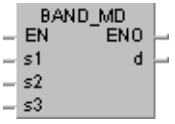
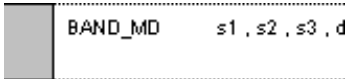
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	●

¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

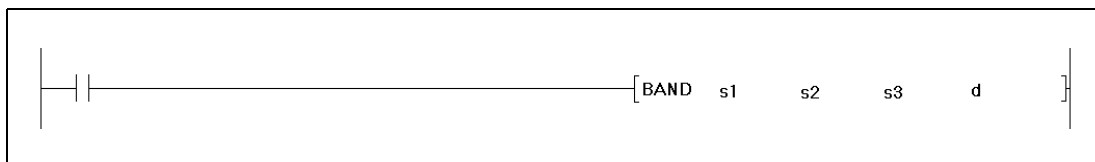
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	●	●	●	●	●	●	●	—	SM0	5	
s2	●	●	●	●	●	●	●	—			
s3	●	●	●	●	●	●	●	—			
d	●	●	●	●	●	●	—	—			

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
--	--	---

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s1	Valore limite inferiore della zona morta (valore di uscita = 0).	BIN 16-bit
s2	Valore limite superiore della zona morta (valore di uscita = 0).	
s3	Valore d'ingresso di cui viene controllata la zona morta.	
d	Indirizzo del primo operando che memorizza il risultato della sottrazione fra il valore d'ingresso e il valore limite.	

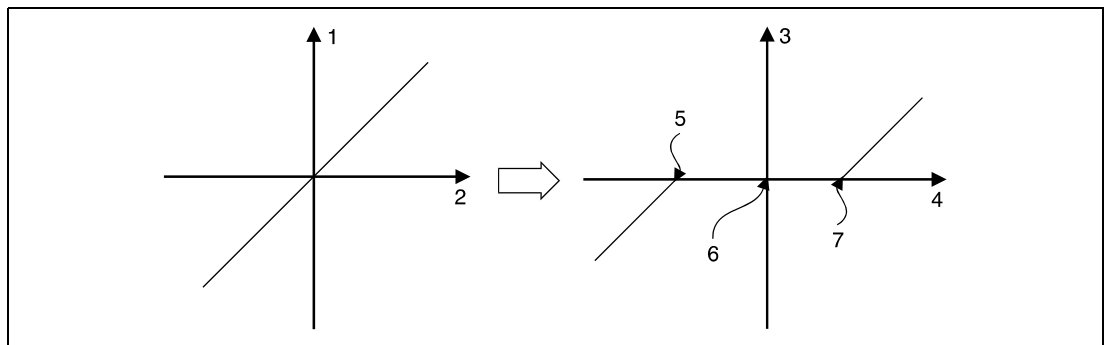
Funzioni **Controllo zona morta dati BIN a 16 e 32-bit****BAND** **Controllo zona morta di dati BIN a 16-bit**

L'istruzione BAND sottrae un valore limite inferiore (negativo) e superiore (positivo) da un valore BIN a 16 bit contenuto nell'operando indicato da s3. Il valore limite inferiore è specificato da s1; il valore limite superiore è specificato da s2. Il risultato viene memorizzato a seconda del valore d'ingresso nell'operando indicato da d, come segue:

Se il valore del dato in s3 è minore del valore del limite inferiore in s1, il risultato della sottrazione s3-s1 viene memorizzato nell'operando specificato da d.

Se il valore del dato in s3 è maggiore del valore del limite superiore in s2, il risultato della sottrazione s3-s2 viene memorizzato nell'operando specificato da d.

Se il valore del dato in s3 è compreso fra i valori limite, il valore 0 viene memorizzato nell'operando indicato da d.



- 1 Valore di uscita
- 2 Valore d'ingresso
- 3 Valore di uscita (d)
- 4 Valore d'ingresso (s3)
- 5 Valore limite inferiore (negativo) (s1)
- 6 Valore di uscita = 0
- 7 Valore limite superiore (positivo) (s2)

I valori specificati da s1, s2 e s3 devono essere nel campo da -32768 a 32767.

Se il risultato della sottrazione risulta fuori del campo da -32768 a 32767, il valore di uscita viene controllato come segue:

Se viene superato il valore -32768, la sottrazione viene eseguita partendo da 32767. Ad esempio, se s3 contiene il valore -32768 e viene sottratto il valore 10 in s1, il risultato diventa

$$-32768 - 10 = 8000H - AH = 7FF6H = 32758.$$

Se viene superato il valore 32760, la sottrazione viene eseguita partendo da -32768.

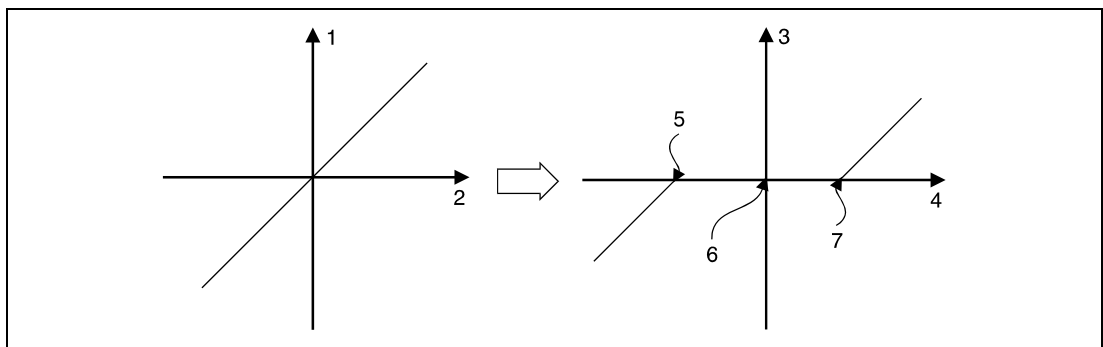
DBAND Controllo zona morta di dati BIN a 32-bit

L'istruzione DBAND sottrae un valore limite inferiore (negativo) e superiore (positivo) da un valore BIN a 32 bit contenuto nell'operando indicato da s3 e (s3)+1. I valore limite inferiore è specificato da s1 e (s1)+1; il valore limite superiore è specificato da s2 e (s2)+1. Il risultato viene memorizzato a seconda del valore d'ingresso nell'operando indicato da d e d+1, come segue:

Se il valore del dato in s3 e (s3)+1 è minore del valore del limite inferiore in s1 e (s1)+1, il risultato della sottrazione s3, (s3)+1 - s1, (s1)+1 viene memorizzato nell'operando specificato da d e d+1.

Se il valore del dato in s3 e (s3)+1 è maggiore del valore del limite superiore in s1 e (s1)+1, il risultato della sottrazione s3, (s3)+1 - s2, (s2)+1 viene memorizzato nell'operando specificato da d e d+1.

Se il valore del dato in s3 e (s3)+1 è compreso fra i valori limite, il valore 0 viene memorizzato nell'operando indicato da d e d+1.



- 1 Valore di uscita
- 2 Valore d'ingresso
- 3 Valore di uscita (d+1, d)
- 4 Valore d'ingresso ((s3)+1, s3)
- 5 Valore limite inferiore (negativo) ((s1)+1, s1)
- 6 Valore di uscita = 0
- 7 Valore limite superiore (positivo) ((s2)+1, s2)

I valori specificati da s1 e (s1)+1, s2 e (s2)+1, s3 e (s3)+1 devono essere nel campo da -2147483648 a 2147483647.

Se il risultato della sottrazione risulta fuori del campo da -2147483648 a 2147483647, il valore di uscita viene controllato come segue:

Se viene superato il valore -2147483648, la sottrazione viene eseguita partendo da 2147483647. Ad esempio, se s3 e (s3)+1 contiene il valore -2147483648 e viene sottratto il valore 1000 contenuto in s1, il risultato diventa

$$-2147483648 - 1000 = 80000000H - 3E8H = 7FFFC18H = 2147482648.$$

Se viene superato il valore 2147483647, la sottrazione viene eseguita partendo da -2147483648.

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il valore in s1 (s1)+1 è superiore a quello in s2 ((s2)+1) (codice di errore 4100).

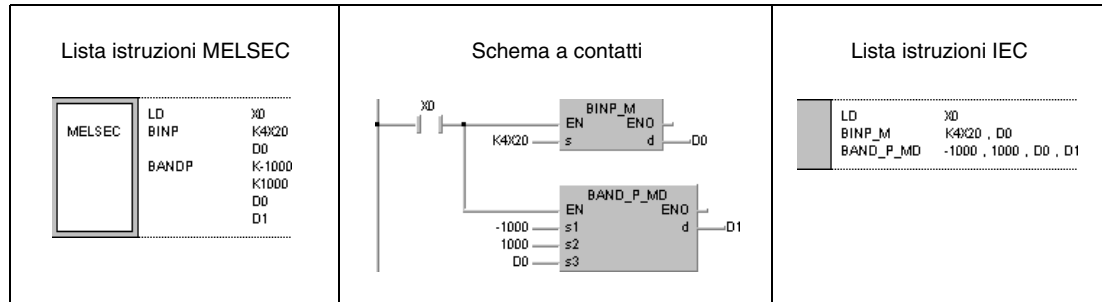
Programma di esempio 1**BANDP**

Con il fronte positivo di X0, il programma seguente sottrae il valore limite inferiore negativo -1000 ed il valore limite superiore (positivo) 1000 dal dato BCD contenuto da X20 a X2F. Il risultato è memorizzato in D1.

Se il valore in D0 è superiore a 1000, il valore $D0 - 1000$ viene inserito in D1.

Se il valore in D0 è inferiore a -1000, il valore $D0 - (-1000)$ viene inserito in D1.

Se il valore in D0 è compreso fra -1000 e 1000, il valore 0 viene inserito in D1.

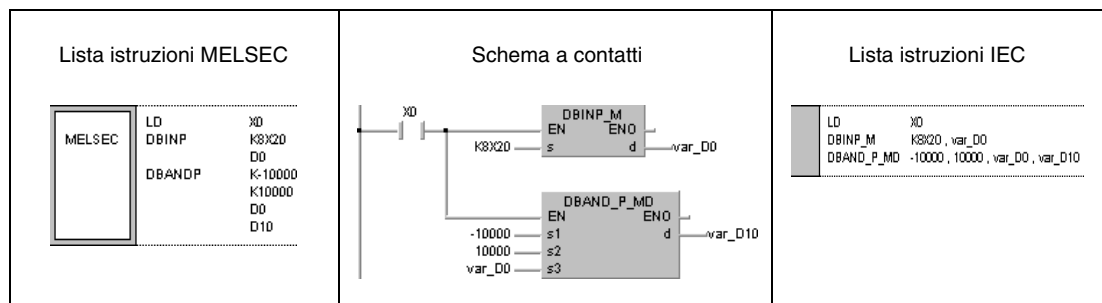
**Programma di esempio 2****DBANDP**

Con il fronte positivo di X0, il programma seguente sottrae il valore limite inferiore negativo -10000 ed il valore limite superiore (positivo) 10000 dal dato BCD contenuto da X20 a X3F. Il risultato viene memorizzato in D10 e D11.

Se il valore in D0 e D1 è superiore a 10000, il valore $D0, D1 - 10000$ viene inserito in D10 e D11.

Se il valore in D0 e D1 è inferiore a -10000, il valore $D0, D1 - (-10000)$ viene inserito in D10 e D11.

Se il valore in D0 e D1 è compreso fra -10000 e 10000, il valore 0 viene inserito in D10 e D11.



7.13.3 ZONE, ZONEP, DZONE, DZONEP

CPU

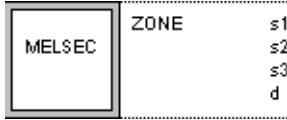
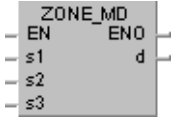
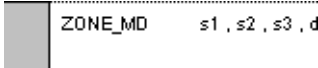
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	●

¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

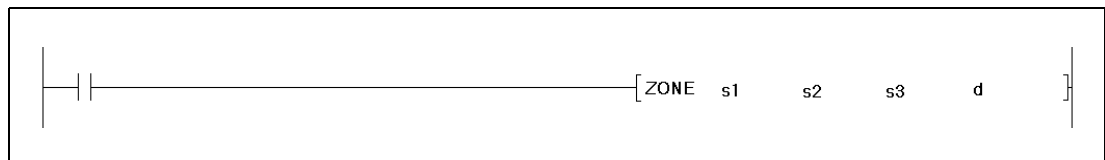
Operandi MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s1	●	●	●	●	●	●	●	●	—	—	5
s2	●	●	●	●	●	●	●	●	—		
s3	●	●	●	●	●	●	●	●	—		
d	●	●	●	●	●	●	●	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
---	---	--

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s1	Valore offset negativo da sommare al valore d'ingresso.	BIN 16-bit
s2	Valore offset positivo da sommare al valore d'ingresso.	
s3	Valore d'ingresso a cui viene sommato il valore di offset.	
d	Indirizzo del primo operando contenente la somma fra valore di ingresso e offset.	

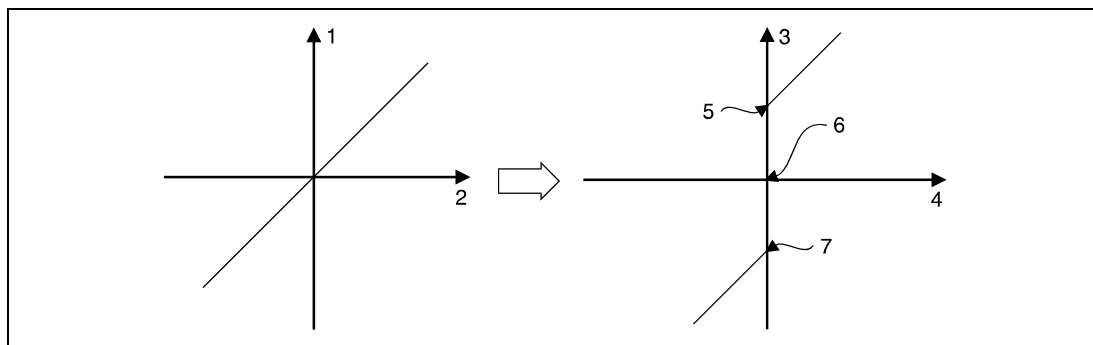
Funzioni Offset su dati BIN a 16/32-bit**ZONE Somma offset su dato BIN a 16-bit**

L'istruzione ZONE somma un valore di offset negativo e positivo al valore BIN a 16 bit del dato contenuto in s3. Il valore di offset negativo è contenuto in s1; il valore di offset positivo è contenuto in s2. Il risultato viene memorizzato a seconda del valore d'ingresso nell'operando indicato da d, come segue:

Se il valore del dato in s3 è minore di 0, il risultato della somma $s3 + s1$ viene memorizzato nell'operando specificato da d.

Se il valore del dato in s3 è maggiore di 0, il risultato della somma $s3 + s2$ viene memorizzato nell'operando specificato da d.

Se il valore del dato in s3 è uguale a 0, il valore 0 viene memorizzato nell'operando indicato da d.



- ¹ Valore di uscita
- ² Valore d'ingresso
- ³ Valore di uscita (d)
- ⁴ Valore d'ingresso (s3)
- ⁵ Valore offset superiore (positivo) (s2)
- ⁶ Valore di ingresso = 0
- ⁷ Valore offset inferiore (negativo) (s1)

I valori specificati da s1, s2 e s3 devono essere nel campo da -32768 a 32767.

Se il risultato della somma risulta fuori del campo da -32768 a 32767, il valore di uscita viene controllato come segue:

Se viene superato il valore -32768, la sottrazione viene eseguita partendo da 32767. Ad esempio, se s3 contiene il valore -32768 e viene sommato il valore -100 contenuto in s1, il risultato diventa

$$-32768 + (-100) = 8000H + FF9CH = 7F9CH = 32668.$$

Se viene superato il valore 32767, la somma viene eseguita partendo da -32768.

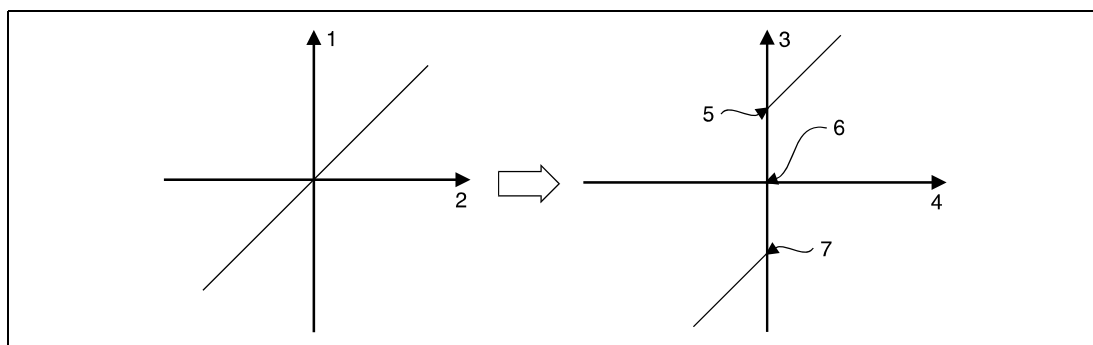
DZONE Somma offset su dato BIN a 32-bit

L'istruzione DZONE somma un valore di offset negativo e positivo al valore BIN a 32 bit del dato contenuto in s3 e (s3)+1. Il valore di offset negativo è contenuto in s1 e (s1)+1; il valore di offset positivo è contenuto in s2 e (s2)+1. Il risultato viene memorizzato a seconda del valore d'ingresso nell'operando indicato da d e d+1, come segue:

Se il valore del dato in s3 e (s3)+1 è minore di 0, il risultato della somma s3, (s3)+1 + s1, (s1)+1 viene memorizzato nell'operando specificato da d e d+1.

Se il valore del dato in s3 e (s3)+1 è maggiore di 0, il risultato della somma s3, (s3)+1 + s2, (s2)+1 viene memorizzato nell'operando specificato da d e d+1.

Se il valore del dato in s3 e (s3)+1 è uguale a 0, il valore 0 viene memorizzato nell'operando indicato da d e d+1.



- 1 Valore di uscita
- 2 Valore d'ingresso
- 3 Valore di uscita (d+1, d)
- 4 Valore d'ingresso ((s3)+1, s3)
- 5 Valore offset superiore (positivo) ((s2)+1, s2)
- 6 Valore di ingresso = 0
- 7 Valore offset inferiore (negativo) ((s1)+1, s1)

I valori specificati da s1 e (s1)+1, s2 e (s2)+1, s3 e (s3)+1 devono essere nel campo da -2147483648 a 2147483647.

Se il risultato della somma risulta fuori del campo da -2147483648 a 2147483647, il valore di uscita viene controllato come segue:

Se viene superato il valore -2147483648, la somma viene eseguita partendo da 2147483647. Ad esempio, se s3 e (s3)+1 contiene il valore -2147483648 e viene sommato il valore -1000 contenuto in s1, il risultato diventa

$$-2147483648 + (-1000) = 80000000H + FFFFFFFC18H = 7FFFFFFC18H = 2147482648.$$

Se viene superato il valore 2147483647, la somma viene eseguita partendo da -2147483648.

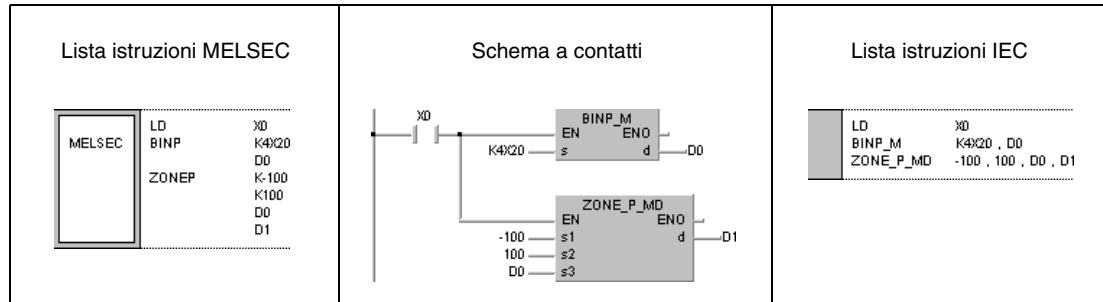
Programma di esempio 1 ZONEP

Con il fronte positivo di X0, il programma seguente somma il valore offset inferiore negativo -100 ed il valore offset superiore (positivo) 100 al dato BCD contenuto da X20 a X2F. Il risultato è memorizzato in D1.

Se il valore in D0 è superiore a 0, il valore $D0 + 100$ viene inserito in D1.

Se il valore in D0 è inferiore a 0, il valore $D0 + (-100)$ viene inserito in D1.

Se il valore in D0 è uguale a 0, il valore 0 viene inserito in D1.

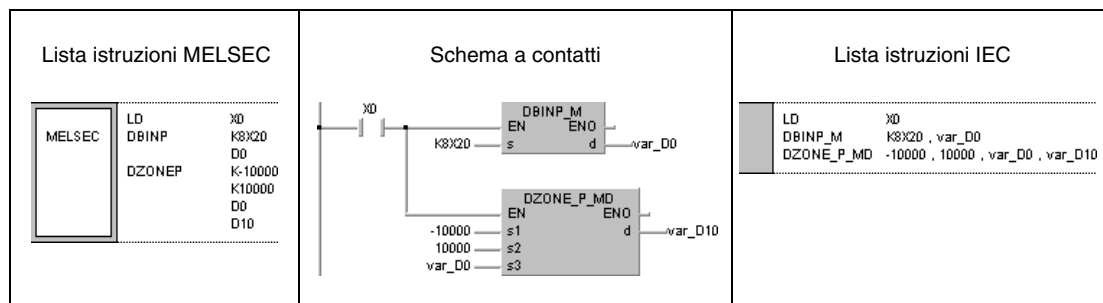
**Programma di esempio 2** DZONEP

Con il fronte positivo di X0, il programma seguente somma il valore offset inferiore negativo -10000 ed il valore offset superiore (positivo) 10000 al dato BCD contenuto da X20 a X3F. Il risultato viene memorizzato in D10 e D11.

Se il valore in D0 e D1 è superiore a 0, il valore $D0, D1 + 10000$ viene inserito in D10 e D11.

Se il valore in D0 e D1 è inferiore a 0, il valore $D0, D1 + (-10000)$ viene inserito in D10 e D11.

Se il valore in D0 e D1 è uguale a 0, il valore 0 viene inserito in D10 e D11.



7.14 Istruzioni di commutazione file registri

Le istruzioni di commutazione consentono la commutazione fra blocchi del file registri e fra nomi di file registri. La tabella che segue offre una panoramica di queste istruzioni.

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Impostazione blocchi registro file	RSET	RSET_MD
		RSET_K_MD
	RSETP	RSET_P_MD
		RSET_K_P_MD
Impostazione nome registro file	QDRSET	QDRSET_M
	QDRSETP	QDRSET_P_MD
Impostazione file commento	QCDSET	QCDSET_M
	QCDSET	QCDSET_P_MD

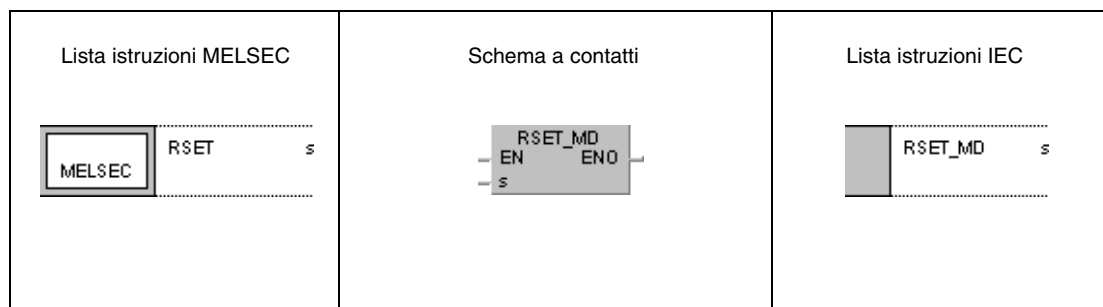
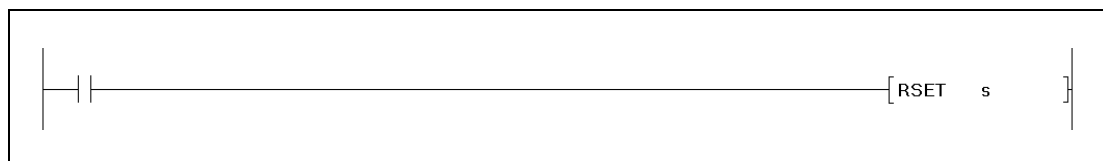
7.14.1 RSET, RSETP**CPU**

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	●

¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

Operandi MELSEC Q

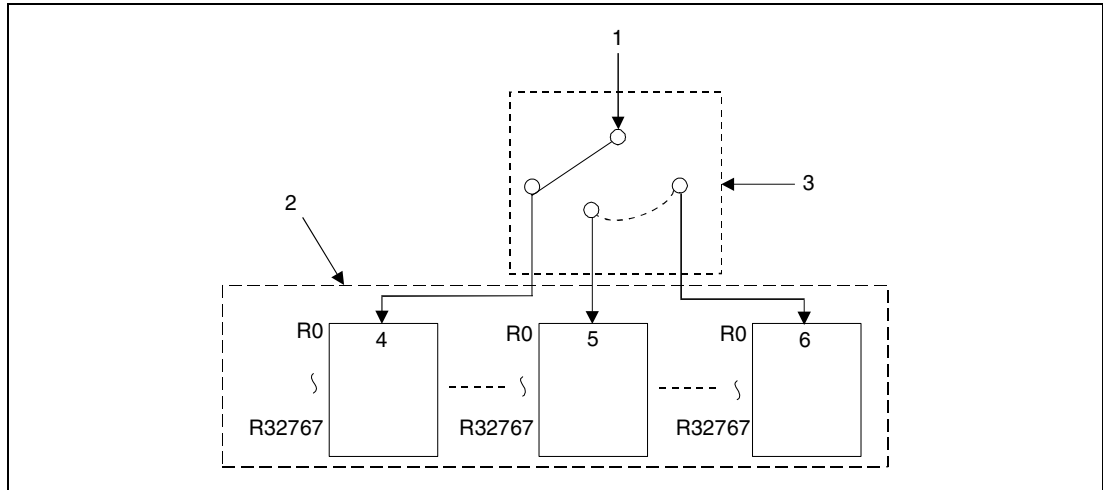
	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	●	●	●	●	●	●	●	—	SM0	2	

GX IEC Developer**GX Developer****Variabili**

Dati impostati	Significato	Tipo dati
s	Numero del blocco nel file registri o indirizzo dell'operando che contiene questo numero.	BIN 16-bit

Funzioni **Impostazione blocchi file registri****RSET** **Istruzione di commutazione blocchi del file registri**

L'istruzione RSET commuta dal blocco del file registri usato da un programma, al blocco del file registri il cui numero è specificato da s. Dopo la commutazione, il programma accede esclusivamente ai registri (R0 - R32767) del blocco specificato.



¹ Elaborazione con accesso al file registri

² File usato dal programma

³ Numero del blocco registri (s)

⁴ Blocco 0

⁵ Blocco 1

⁶ Blocco n

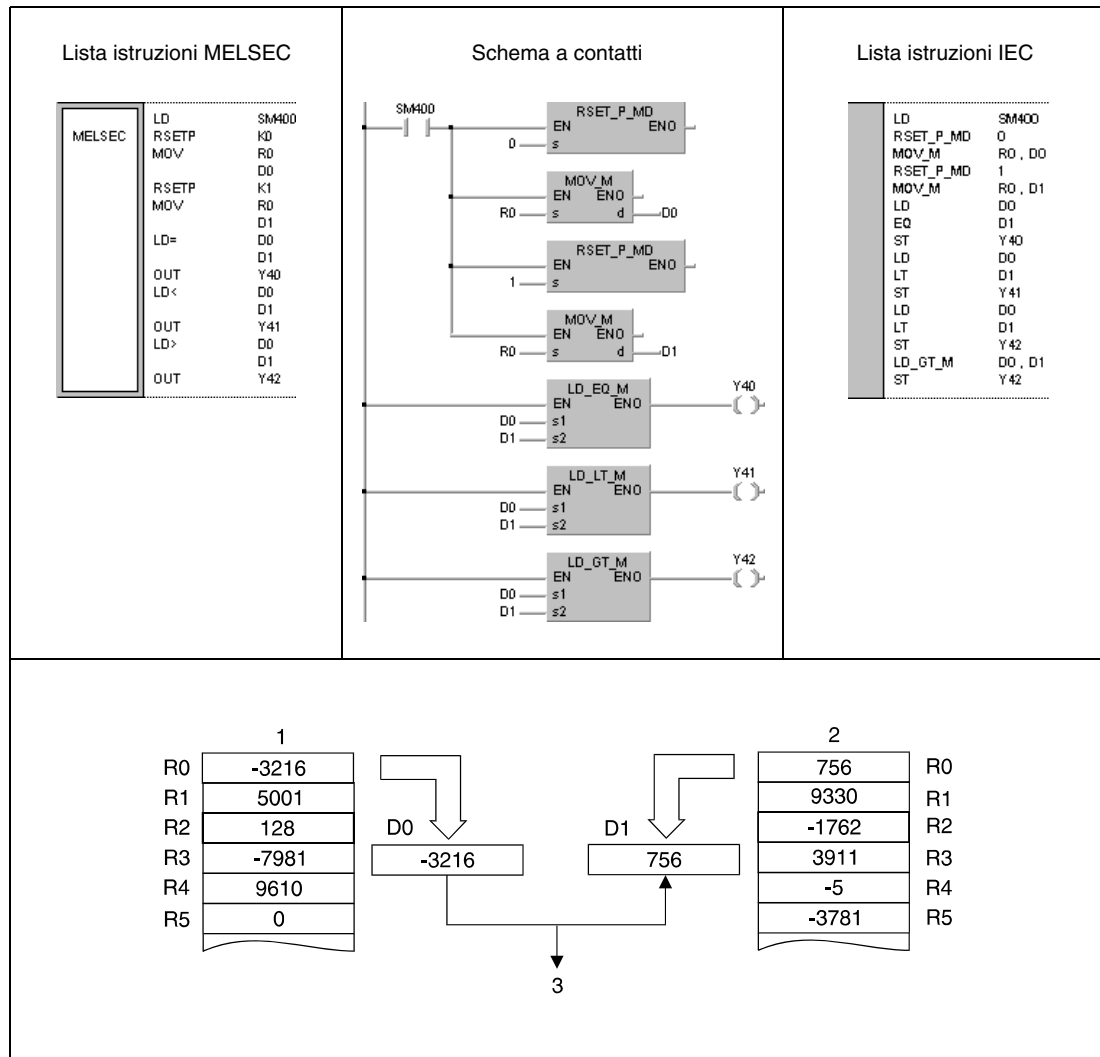
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il di blocco specificato da s non esiste (codice di errore 4100).
- Il blocco specificato da s non contiene registri (codice di errore 4101).

Esempio di programma RSETP

Con il fronte positivo di SM400, il programma seguente confronta il registro su file R0 nel blocco registri 0 con il registro su file R0 nel blocco registri 1. I blocchi di registri 0 e 1 vengono indirizzati tramite l'istruzione RSET. Entrambi i registri R0 vengono letti con l'istruzione MOV. Se il valore in R0 (blocco 0) è uguale al valore in R0 (block 1), viene attivata l'uscita Y40. Se il valore in R0 (blocco 0) è inferiore al valore in R0 (block 1), viene attivata l'uscita Y41. Se il valore in R0 (blocco 0) è superiore al valore in R0 (block 1), viene attivata l'uscita Y42.



¹ Blocco 0

² Blocco 1

³ Y41 viene attivata perché D0 è inferiore a D1.

7.14.2 QDRSET, QDRSETP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

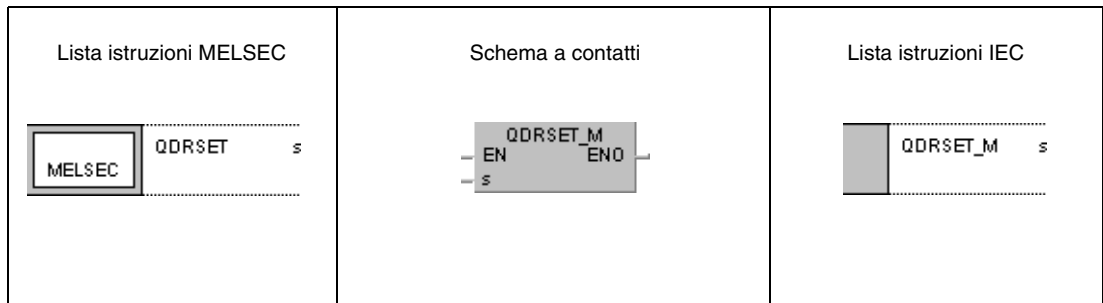
¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

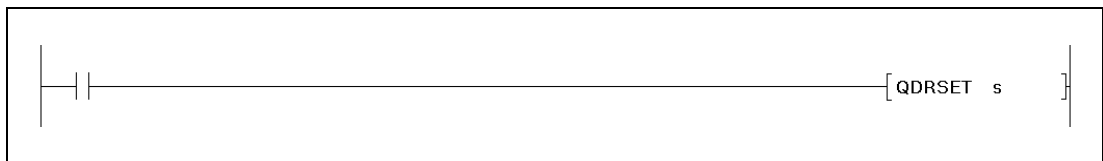
	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti \$			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	●	—	SM0	2 + n ¹⁾

¹ n = (numero di caratteri del nome programma)/2 = numero di passi aggiuntivi (frazioni decimali arrotondate per eccesso)

GX IEC Developer



GX Developer

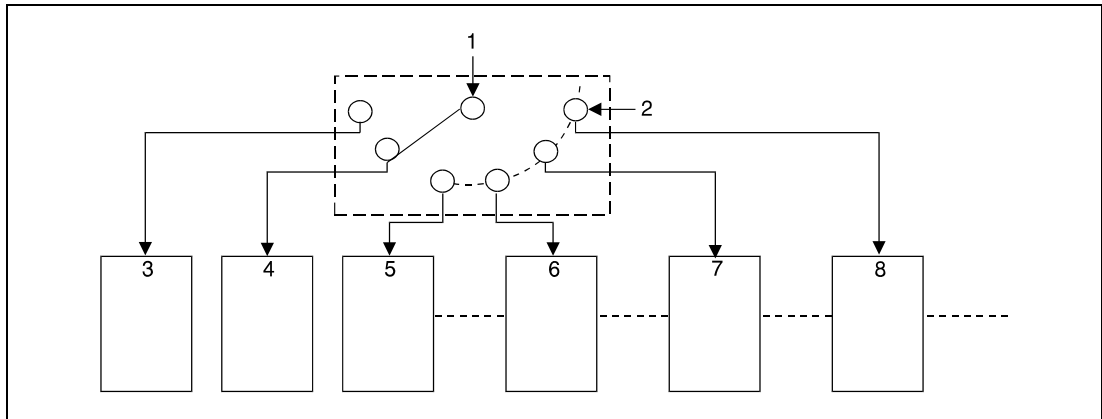


Variabili

Dati impostati	Significato	Tipo dati
s	Numero drive e nome file del file registri su cui commutare, o indirizzo dell'operando che contiene questo dato.	Stringa caratteri

Funzioni **Impostazione nome file registri****QDRSET** **Istruzione di commutazione per file registri**

L'istruzione QDRSET commuta dal file registri usato da un programma, al file registri specificato da s. Dopo la commutazione, il programma accede esclusivamente ai registri (R0 - R32767) del blocco 0 del file registri specificato. I blocchi del file registri vengono selezionati con l'istruzione RSET.



¹ Elaborazione con accesso al file registri

² Impostazione drive e file

³ Drive 1, file A

⁴ Drive 1, file B

⁵ Drive 1, file C

⁶ Drive 2, file A

⁷ Drive 3, file A

⁸ Drive 4, file A

Possono essere assegnati fino a 4 drive (da 1 a 4). Il drive 0 non può essere assegnato; il codice è riservato per la memoria interna.

Non è necessario specificare l'estensione .QDR del nome file.

Un nome file impostato può essere cancellato specificando il carattere NULL (00H) come nome file.

I file registri selezionati con l'istruzione QDRSET sono prioritari rispetto a numero drive e nome file specificati dai parametri di taratura.

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il file registri non esiste sul drive specificato da s (codice di errore 2410).

Esempio di programma

QDRSET/QDRSETP

Con il fronte positivo di X0, il programma seguente commuta sul file registri ABC.QDR sul drive 1. Fino a quando X1 è attivo, viene utilizzato il file registri DEF.QDR sul drive 3.

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC																								
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10%; text-align: center;">MELSEC</td> <td style="width: 10%; border-right: 1px dashed black;">LD</td> <td style="width: 10%;">X0</td> <td style="width: 10%; border-right: 1px dashed black;">QDRSETP</td> <td style="width: 10%;">"1:ABC"</td> <td style="width: 10%;"></td> </tr> <tr> <td></td> <td style="border-right: 1px dashed black;">LD</td> <td>X1</td> <td style="border-right: 1px dashed black;">QDRSET</td> <td>"3:DEF"</td> <td></td> </tr> </table>	MELSEC	LD	X0	QDRSETP	"1:ABC"			LD	X1	QDRSET	"3:DEF"			<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10%;"></td> <td style="width: 10%; border-right: 1px dashed black;">LD</td> <td style="width: 10%;">X0</td> <td style="width: 10%; border-right: 1px dashed black;">QDRSETP_M</td> <td style="width: 10%;">"1:ABC"</td> <td style="width: 10%;"></td> </tr> <tr> <td></td> <td style="border-right: 1px dashed black;">LD</td> <td>X1</td> <td style="border-right: 1px dashed black;">QDRSET_M</td> <td>"3:DEF"</td> <td></td> </tr> </table>		LD	X0	QDRSETP_M	"1:ABC"			LD	X1	QDRSET_M	"3:DEF"	
MELSEC	LD	X0	QDRSETP	"1:ABC"																						
	LD	X1	QDRSET	"3:DEF"																						
	LD	X0	QDRSETP_M	"1:ABC"																						
	LD	X1	QDRSET_M	"3:DEF"																						

7.14.3 QCDSET, QCDSETP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

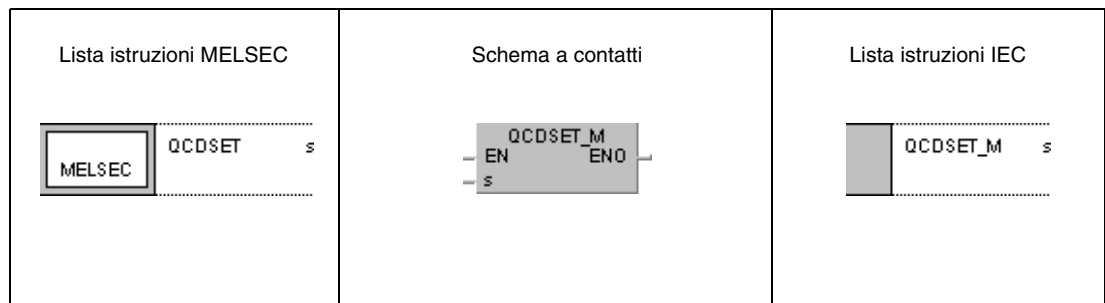
¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

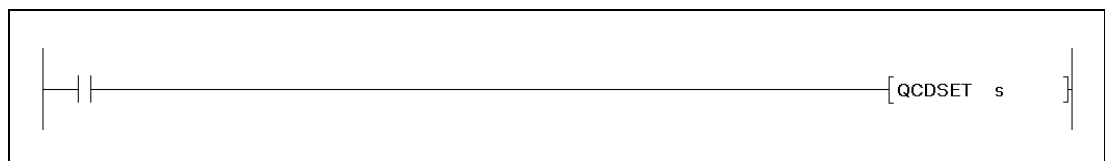
	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti \$			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	●	—	SM0	2 + n ¹⁾

¹ n = (numero di caratteri del nome programma)/2 = numero di passi aggiuntivi (frazioni decimali arrotondate per eccesso)

GX IEC Developer



GX Developer



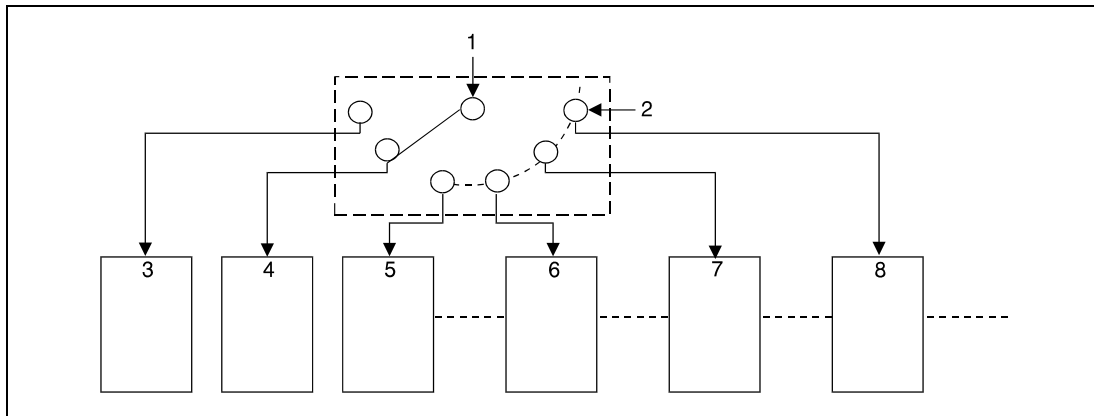
Variabili

Dati impostati	Significato	Tipo dati
s	Numero drive e nome file del file commenti su cui commutare, o indirizzo dell'operando che contiene questo dato.	Stringa caratteri

Funzioni **Impostazione file commenti**

QCDSET **Istruzione di commutazione per file di commento**

L'istruzione QCDSET commuta dal file registri usato da un programma, al file commenti specificato da s. Dopo la commutazione, il programma accede esclusivamente al nuovo file commenti specificato.



- 1 Elaborazione con accesso a file commenti
- 2 Impostazione drive e file commenti
- 3 Drive 1, file A
- 4 Drive 1, file B
- 5 Drive 1, file C
- 6 Drive 2, file A
- 7 Drive 3, file A
- 8 Drive 4, file A

Possono essere assegnati fino a 4 drive (da 1 a 4). Il drive 0 non può essere assegnato; il codice è riservato per la memoria interna.

Non è necessario specificare l'estensione .QCD del nome file.

Un nome file impostato può essere cancellato specificando il carattere NULL (00H) come nome file.

I file registri selezionati con l'istruzione QCDSET sono prioritari rispetto a numero drive e nome file specificati dai parametri di taratura.

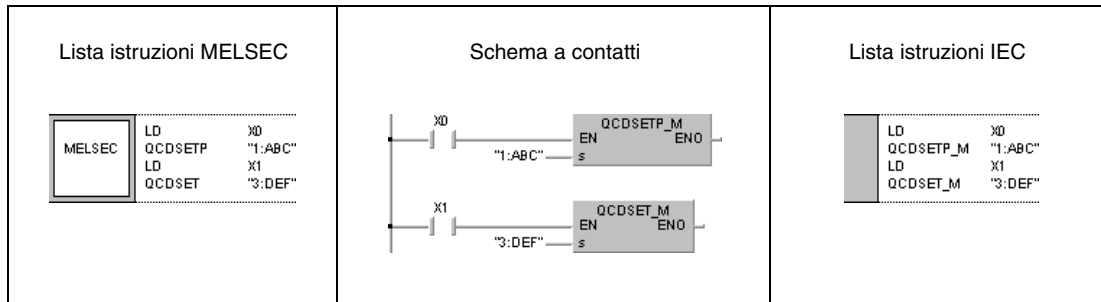
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il file commenti non esiste sul drive specificato da s (codice di errore 2410).

Esempio di programma**QCDSET/QCDSETP**

Con il fronte positivo di X0, il programma seguente commuta sul file commenti ABC.QCD sul drive 1. Fino a quando X1 è attivo, viene utilizzato il file registri DEF.QCD sul drive 3.



7.15 Istruzioni per orologio

Queste istruzioni leggono e scrivono, sommano e sottraggono, oppure modificano il formato dei dati orologio. La tabella che segue offre una panoramica di queste istruzioni.

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Lettura dati orologio	DATERD	DATERD_MD
	DATERDP	DATERD_P_MD
Scrittura dati orologio	DATEWR	DATEWR_MD
	DATEWRP	DATEWR_P_MD
Somma dati orologio	DATE+	DATEPLUS_M
	DATE+P	DATEPLUSP_M
Sottrazione dati orologio	DATE-	DATEMINUS_M
	DATE-P	DATEMINUSP_M
Modifica formato orologio da hh:mm:ss a secondi	SECOND	SECOND_M
	SECONDP	SECONDP_M
Modifica formato orologio da secondi a hh:mm:ss	HOUR	HOUR_M
	HOURP	HOURP_M

7.15.1 DATERD, DATERDP

CPU

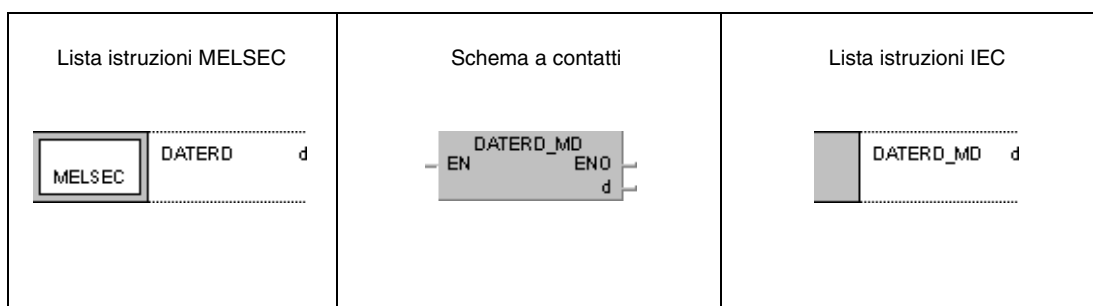
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	●

¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

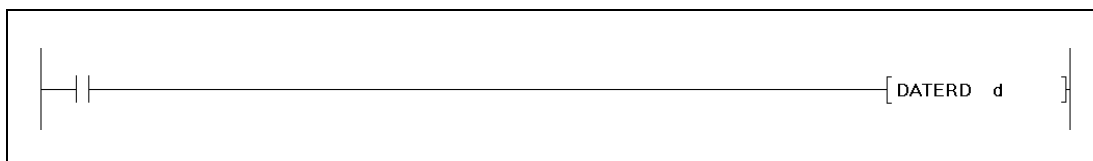
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti			Altro
	Bit	Word		Bit	Word						
d	—	●	●	—	—	—	—	—	—	2	

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
d	Indirizzo del primo operando che memorizza i dati orologio letti.	BIN 16-bit	Array [0.0.6] di ANY16

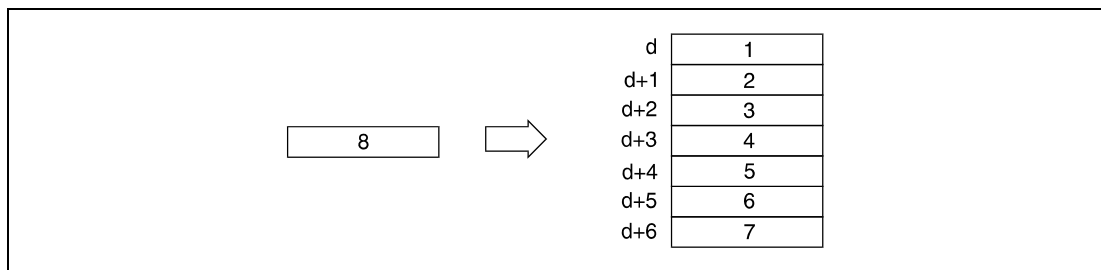
Funzioni Lettura dati orologio

DATERD Istruzione di lettura

L'istruzione DATERD legge anno, mese, giorno, ora, minuto, secondo e giorno della settimana dall'orologio interno delle CPU QnA e memorizza in binario i dati orologio negli operandi da d+0 (Array_d[0]) a d+6 (Array_d[6]). L'assegnazione dei registri ai dati orologio è mostrata di seguito:

- d+0, array_d[0] = anno (1)
- d+1, array_d[1] = mese (Gennaio = 1, Dicembre = 12) (2)
- d+2, array_d[2] = giorno (3)
- d+3, array_d[3] = ora (formato 24 ore) (4)
- d+4, array_d[4] = minuto (5)
- d+5, array_d[5] = secondo (6)
- d+6, array_d[6] = giorno della settimana (7)

L'orologio della CPU QnA è indicato con 8 nella figura.



La tabella seguente contiene l'insieme dei valori orologio nei dati da d+0 a d+6:

Dati orologio	Anno	Mese	Giorno	Ora	Minuto	Secondo	Giorno della settimana
Campo d'ingresso	1)	1 - 12	1 - 31	0 - 23	0 - 59	0 - 59	0 - 6
Operandi	d+0 (array_d[0])	d+1 (array_d[1])	d+2 (array_d[2])	d+3 (array_d[3])	d+4 (array_d[4])	d+5 (array_d[5])	d+6 (array_d[6])

¹ da 0 a 99 per CPU QnA, da 1980 a 2079 per CPU System Q

L'anno è memorizzato nella CPU QnA come numero a due cifre. Vengono memorizzate solo le unità e decine (ad es. 1998 = 98).

Se viene usata una CPU System Q, l'anno viene memorizzato come numero a quattro cifre.

Il giorno della settimana contenuto in d+6 (Array_d[6]) viene espresso da un numero da 0 a 6. La tabella seguente mostra l'assegnazione dei giorni della settimana:

Giorno della settimana	Domenica	Lunedì	Martedì	Mercoledì	Giovedì	Venerdì	Sabato
Valore	0	1	2	3	4	5	6

Gli anni bisestili vengono calcolati automaticamente dall'orologio della CPU.

Esempio di programma DATERD (CPU QnA)

Se SM400 è attivo, il programma seguente legge i dati dell'orologio interno della CPU e li invia in formato BCD sulle uscite da Y47 a Y67, come segue:

Y60 - Y67 = mese

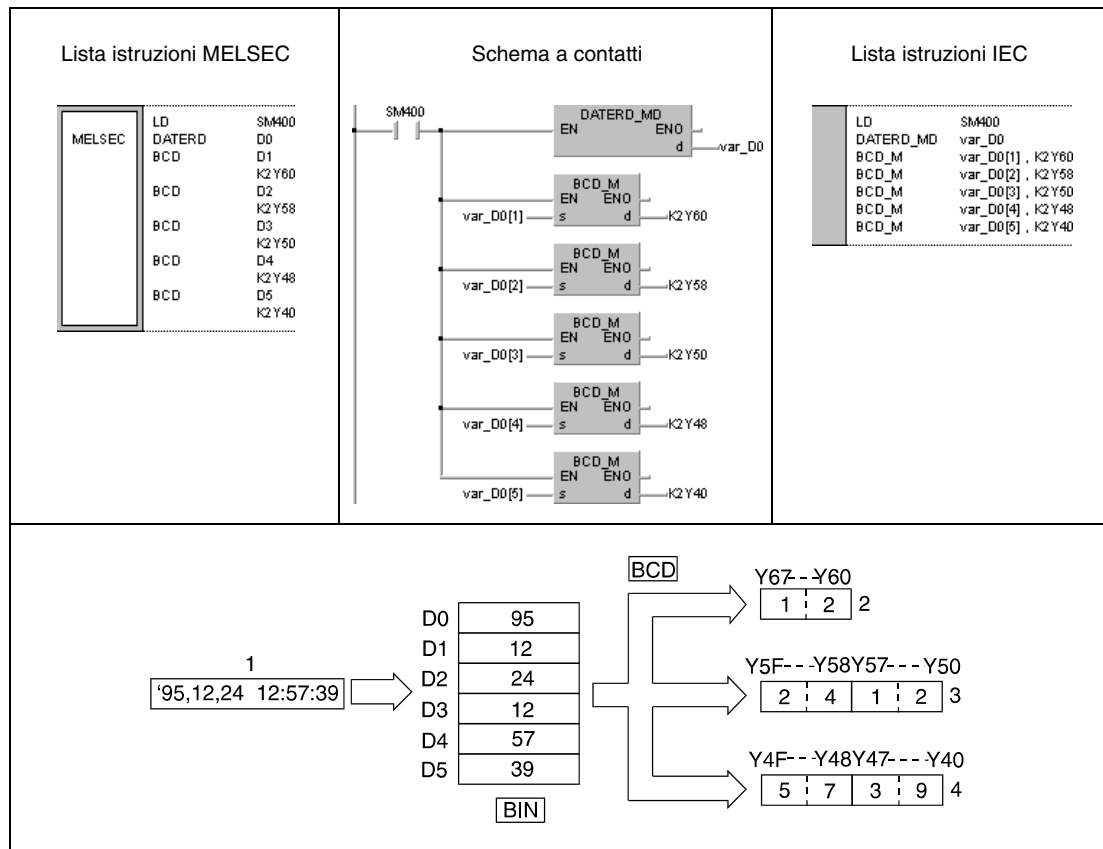
Y58 - Y5F = giorno

Y50 - Y57 = ora

Y48 - Y4F = minuto

Y40 - Y47 = secondo

Dati orologio	Anno	Mese	Giorno	Ora	Minuto	Secondo	Giorno della settimana
Operandi	D0 (var_D0[0])	D1 (var_D0[1])	D2 (var_D0[2])	D3 (var_D0[3])	D4 (var_D0[4])	D5 (var_D0[5])	D6 (var_D0[6])



¹ Dati orologio

² Mese

³ Giorno, ora

⁴ Minuto, secondo

NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

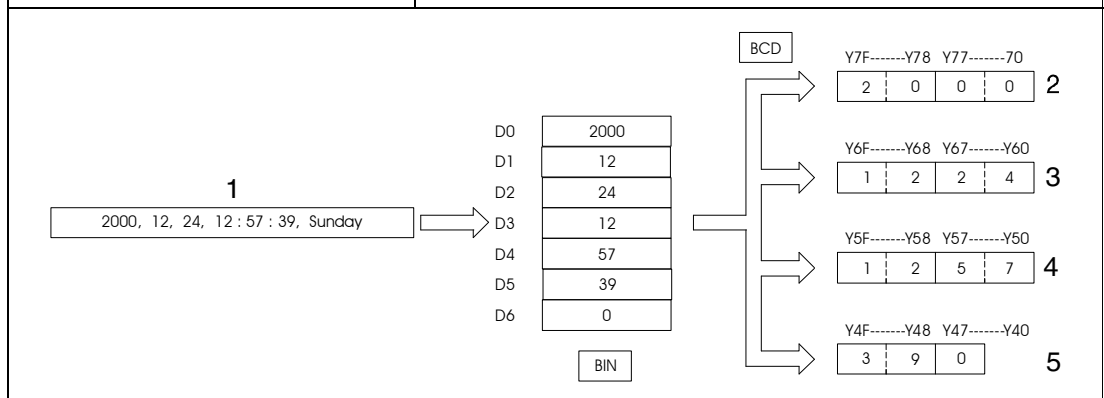
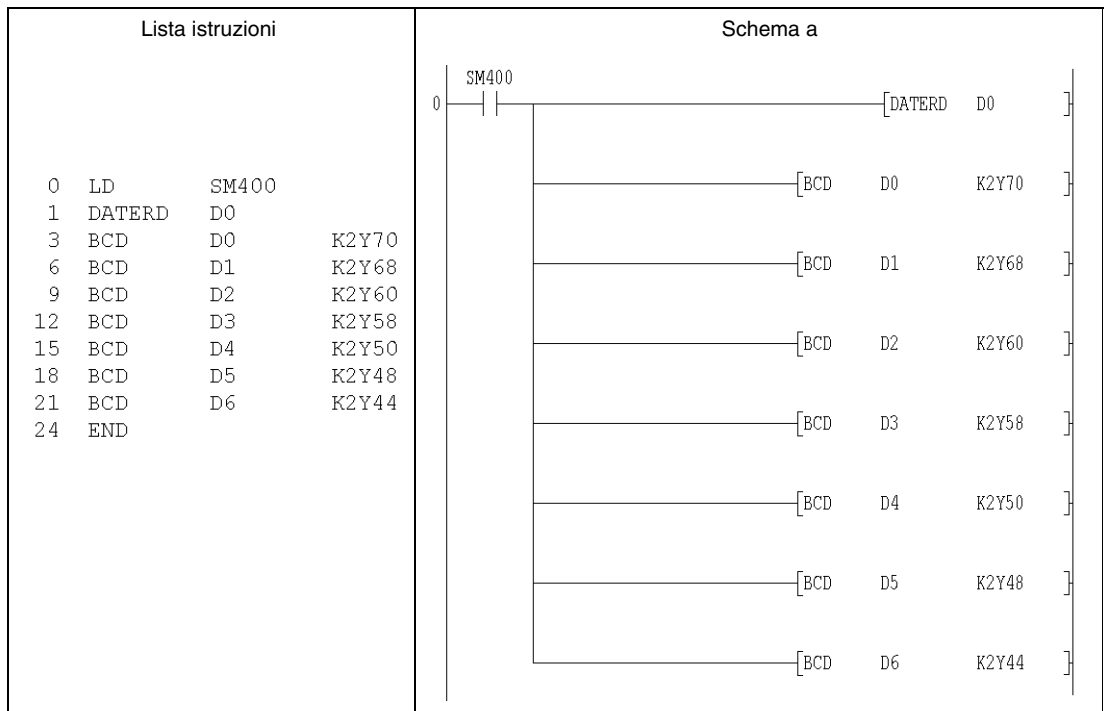
Programma DATERD (CPU System Q)

Esempio

Se SM400 è attivo, il programma seguente legge i dati dell'orologio interno della CPU e li invia in formato BCD sulle uscite da Y47 a Y67, come segue:

- Y70 - Y7F = anno Y68 - Y6F = mese
- Y60 - Y67 = giorno Y58 - Y5F = ora
- Y50 - Y57 = minuto Y48 - Y4F = secondi
- Y44 - Y47 = giorno della settimana

Dati orologio	Anno	Mese	Giorno	Ora	Minuto	Secondo	Giorno della settimana
Operandi	D0	D1	D2	D3	D4	D5	D6



- ¹ Dati orologio
- ² Anno
- ³ Mese, giorno
- ⁴ Ora, minuto
- ⁵ Secondo, giorno della settimana

7.15.2 DATEWR, DATEWRP

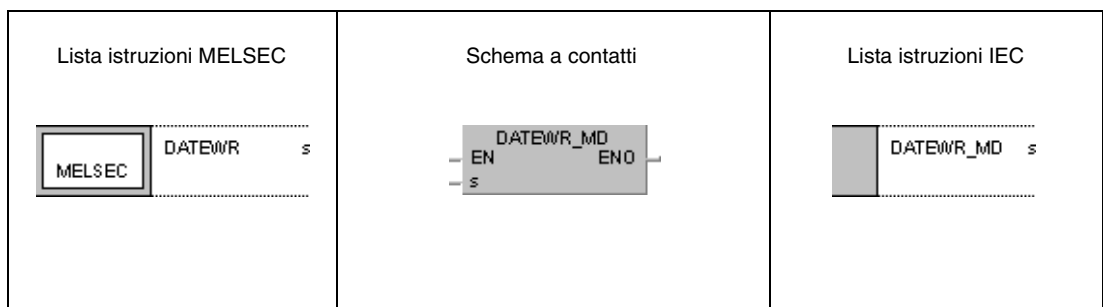
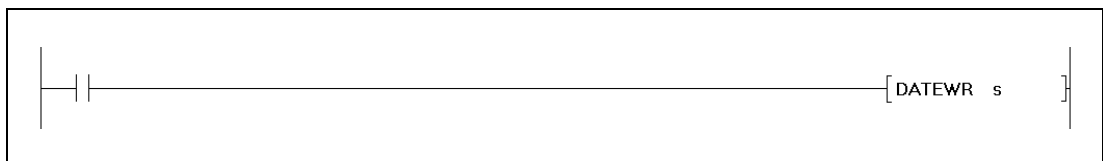
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	●

¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

Operandi
MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro DY
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	—	—	SM0	2

GX IEC
DeveloperGX
Developer

Variabili

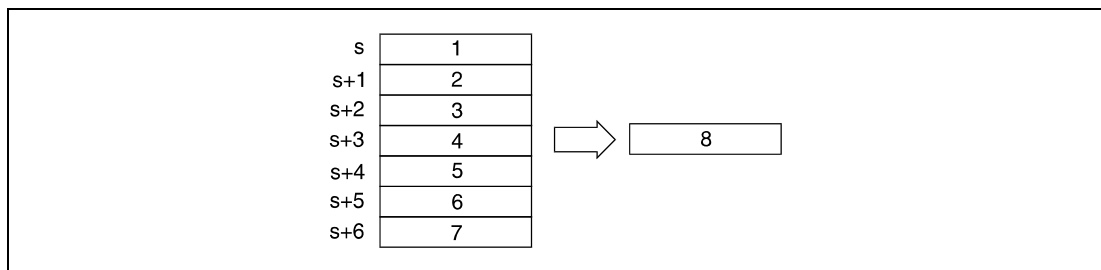
Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s	Indirizzo del primo operando che contiene i dati da scrivere nell'orologio interno della CPU.	BIN 16-bit	Array [0.0.6] di ANY16

Funzioni Scrittura dati orologio**DATEWR Istruzione di scrittura**

L'istruzione DATEWR scrive i dati orologio anno, mese, giorno, ora, minuto, secondo e giorno della settimana contenuti negli operandi da d+0 (Array_d[0]) a d+6 (Array_d[6]) nell'orologio interno della CPU. I dati dell'orologio vengono memorizzati in formato binario. L'assegnazione dei registri ai dati orologio è mostrata di seguito:

- s+0, array_s[0] = anno (1)
- s+1, array_s[1] = mese (Gennaio = 1, Dicembre = 12) (2)
- s+2, array_s[2] = giorno (3)
- s+3, array_s[3] = ora (formato 24 ore) (4)
- s+4, array_s[4] = minuto (5)
- s+5, array_s[5] = secondo (6)
- s+6, array_s[6] = giorno della settimana (7)

L'orologio della CPU QnA è indicato con 8 nella figura.



La tabella seguente contiene l'insieme dei valori orologio nei dati da d+0 a d+6:

Dati orologio	Anno	Mese	Giorno	Ora	Minuto	Secondo	Giorno della settimana
Campo d'ingresso	1)	1 - 12	1 - 31	0 - 23	0 - 59	0 - 59	0 - 6
Operandi	s+0 (array_s[0])	s+1 (array_s[1])	s+2 (array_s[2])	s+3 (array_s[3])	s+4 (array_s[4])	s+5 (array_s[5])	s+6 (array_s[6])

¹ da 0 a 99 per CPU QnA, da 1980 a 2079 per CPU System Q

L'anno è memorizzato nella CPU QnA come numero a due cifre. Vengono memorizzate solo le unità e decine (ad es. 1998 = 98). Se viene usata una CPU System Q, l'anno viene memorizzato come numero a quattro cifre.

Il giorno della settimana contenuto in s+6 (Array_s[6]) viene espresso da un numero da 0 a 6. La tabella seguente mostra l'assegnazione dei giorni della settimana:

Giorno della settimana	Domenica	Lunedì	Martedì	Mercoledì	Giovedì	Venerdì	Sabato
Valore	0	1	2	3	4	5	6

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- I dati orologio specificati da s+0 (Array_s[0]) a s+6 (Array_s[6]) supera il campo valori ammesso (codice di errore 4100).

Esempio di programma DATEWRP (CPU QnA)

Con il fronte positivo di X40, il programma seguente scrive i dati orologio in formato binario contenuti sugli ingressi da X0 a X2F nell'orologio interno della CPU. Gli ingressi vengono assegnati ai dati orologio come segue:

- X28 - X2F = anno
- X20 - X27 = mese
- X18 - X1F = giorno
- X10 - X17 = ora
- X8 - XF = minuto
- X0 - X7 = secondo

Dati orologio	Anno	Mese	Giorno	Ora	Minuto	Secondo	Giorno della settimana
Operandi	D0 (var_D0[0])	D1 (var_D0[1])	D2 (var_D0[2])	D3 (var_D0[3])	D4 (var_D0[4])	D5 (var_D0[5])	D6 (var_D0[6])

Lista istruzioni MELSEC

```

MELSEC
LD      X40
BIN     K2X28
        D0
BIN     K2X20
        D1
BIN     K2X18
        D2
BIN     K2X10
        D3
BIN     K2X8
        D4
BIN     K2X0
        D5
DATEWRP D0
        
```

Schema a contatti

Lista istruzioni IEC

```

LD      X40
BIN_M   K2X28 , var_D0[0]
BIN_M   K2X20 , var_D0[1]
BIN_M   K2X18 , var_D0[2]
BIN_M   K2X10 , var_D0[3]
BIN_M   K2X8  , var_D0[4]
BIN_M   K2X0  , var_D0[5]
DATEWRP_P_MD var_D0
        
```

X2F---X28X27--X20 **BIN**

1 9 5 0 1

X1F---X18X17--X10

2 0 1 1 2

X0F---X08X07--X00

3 0 0 0 0

D0 95

D1 D1

D2 D1

D3 12

D4 D0

D5 D0

BIN

4

'95,01,01 12:00:00

- ¹ Anno, mese
- ² Giorno, ora
- ³ Minuto, secondo
- ⁴ Dati orologio

NOTA

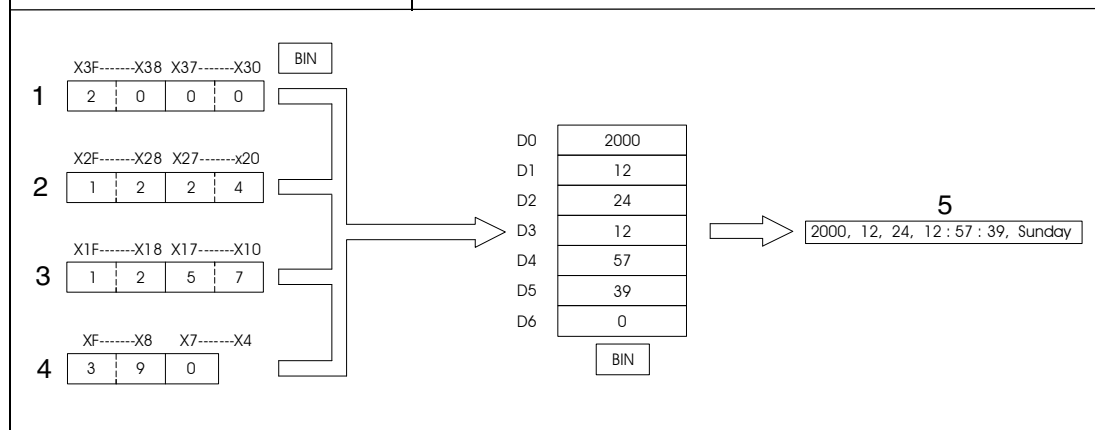
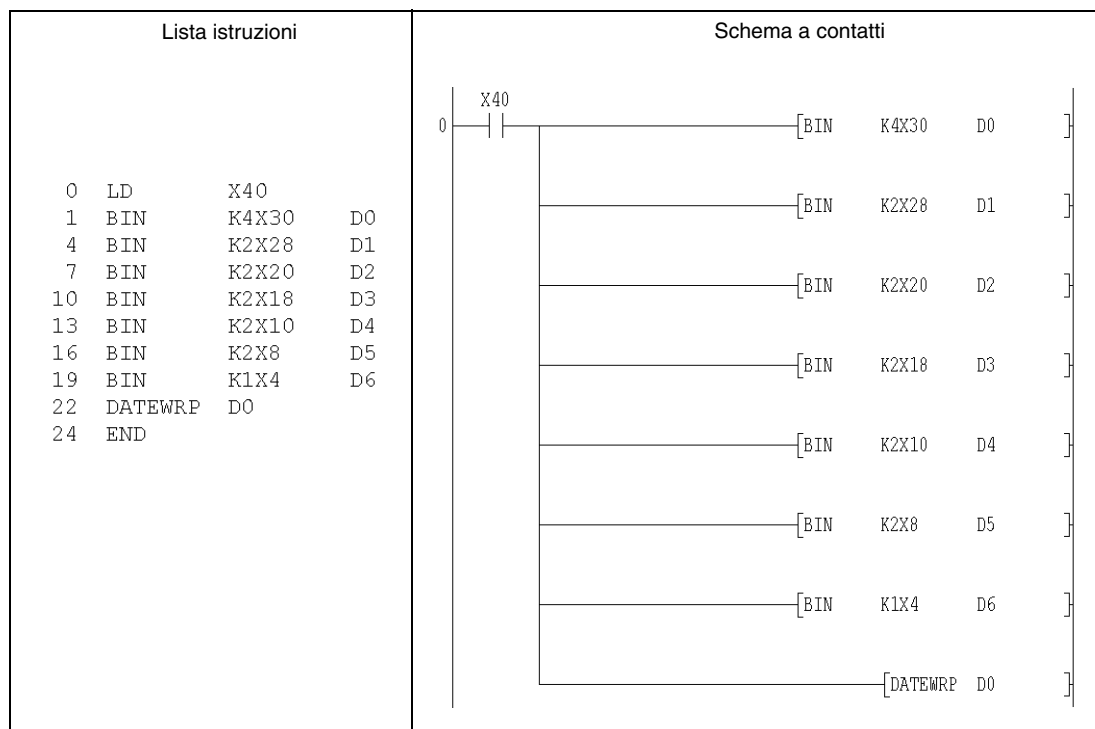
Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provocano messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

Programma DATEWRP (CPU System Q)
Esempio

Con il fronte positivo di X40, il programma seguente scrive i dati orologio in formato binario contenuti sugli ingressi da X0 a X2F nell'orologio interno della CPU. Gli ingressi vengono assegnati ai dati orologio come segue:

- X30 - X3F = anno
- X18 - X1F = ora
- X28 - X2F = mese
- X10 - X17 = minuto
- X20 - X27 = giorno
- X8 - XF = secondo

Dati orologio	Anno	Mese	Giorno	Ora	Minuto	Secondo	Giorno della settimana
Operandi	D0	D1	D2	D3	D4	D5	D6



- ¹ Anno
- ² Mese, giorno
- ³ Ora, minuto
- ⁴ Secondo, giorno della settimana
- ⁵ Dati orologio

7.15.3 DATE+, DATE+P

CPU

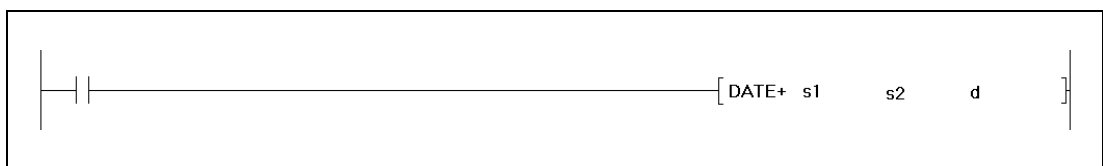
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi
MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	SM0	4	
s2	—	●	●	—	—	—	—	—			
d	—	●	●	—	—	—	—	—			

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
<div style="border: 1px solid black; padding: 5px; display: inline-block;">MELSEC</div> <div style="margin-left: 20px;">DATE+ s1 s2 d</div>		<div style="border: 1px solid black; padding: 5px; display: inline-block;">DATEPLUS_M</div> s1, s2, d

GX
Developer

Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s1	Dati orologio a cui si deve sommare.	BIN 16-bit	Array [0.0.2] di ANY16
s2	Dati orologio da sommare.		
d	Indirizzo del primo operando che memorizza i dati orologio che risultano dall'operazione.		

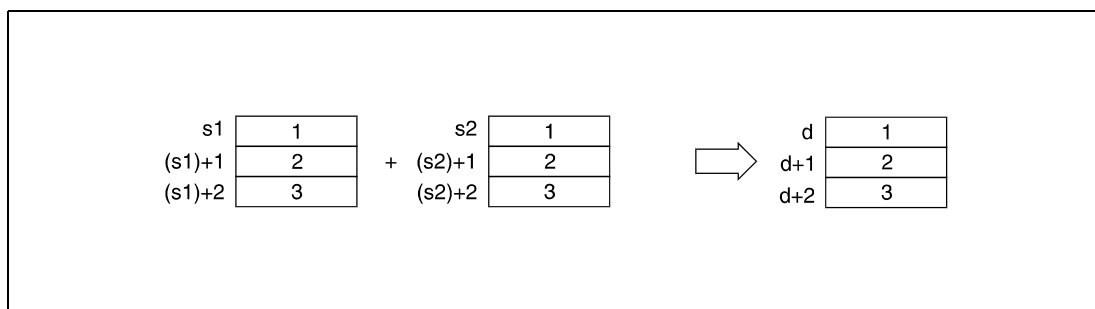
Funzioni **Somma dati orologio**

DATE+ Istruzione di somma

L'istruzione DATE+ somma i dati orologio contenuti negli operandi indicati da s2 ai dati orologio contenuti negli operandi indicati da s1. I dati orologio che risultano dall'operazione vengono memorizzati negli operandi specificati da d.

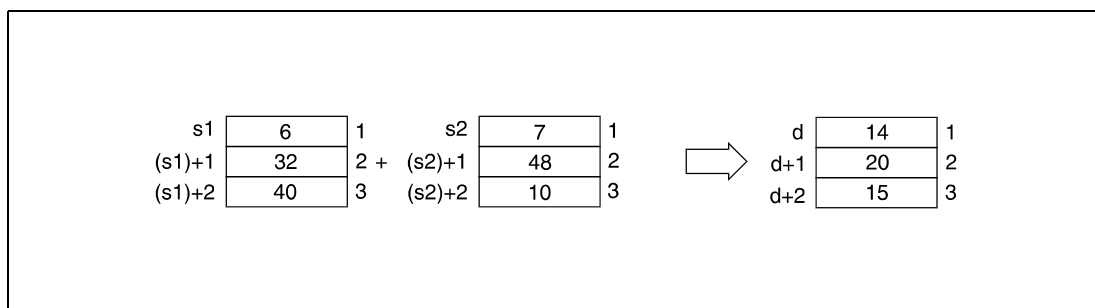
La tabella seguente contiene i campi di valori dei dati orologio da (s1)+0 a (s1)+2, da (s2)+0 a (s2)+2, e da d+0 a d+2):

Dati orologio	Anno	Mese	Giorno	Ora	Minuto	Secondo	Giorno della settimana
Campo d'ingresso	—	—	—	0 - 23	0 - 59	0 - 59	—
Operandi	—	—	—	s1+0 (array_s1[0])	s1+1 (array_s1[1])	s1+2 (array_s1[2])	—
Operandi	—	—	—	s2+0 (array_s2[0])	s2+1 (array_s2[1])	s2+2 (array_s2[2])	—
Operandi	—	—	—	d+0 (array_d[0])	d+1 (array_d[1])	d+2 (array_d[2])	—



- ¹ Ora
- ² Minuto
- ³ Secondo

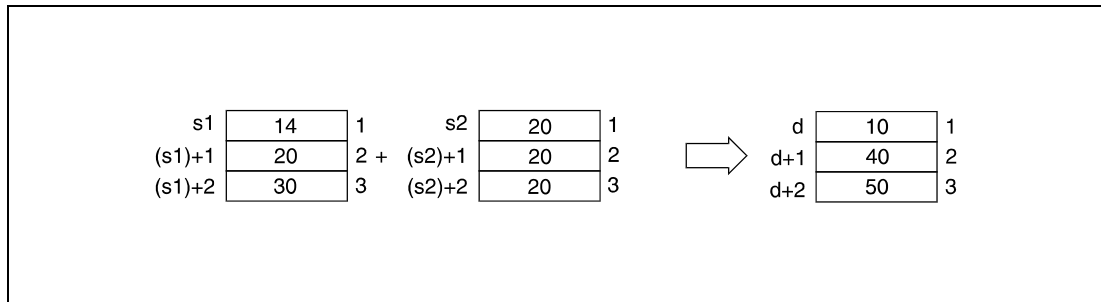
Nella figura seguente i dati orologio 6 ore, 32 minuti, 40 secondi da ((s1)+0 a (s1)+2) vengono aggiunti ai dati orologio 7 ore, 48 minuti, 10 secondi da ((s2)+0 a (s2)+2). Il risultato 14 ore, 20 minuti, 50 secondi viene memorizzato da d+0 a d+2).



- ¹ Ora
- ² Minuto
- ³ Secondo

Se il risultato della somma dei dati orologio supera 24 ore, vengono sottratte automaticamente 24 ore per ottenere un valore di tempo corretto.

La figura seguente mostra la somma di 14 ore, 20 minuti e 30 secondi con 20 ore, 20 minuti e 20 secondi. Il risultato sarebbe 34 ore, 40 minuti e 50 secondi. Dato che questo risultato non è un formato orologio corretto, dopo la sottrazione automatica di 24 ore, il risultato viene corretto in 10 ore, 40 minuti e 50 secondi (10:40:50 del giorno successivo).



¹ Ora

² Minuto

³ Secondo

NOTA

Fare riferimento alla sezione "Scrittura dati orologio" per ulteriori informazioni su questo argomento.

Errori di esecuzione

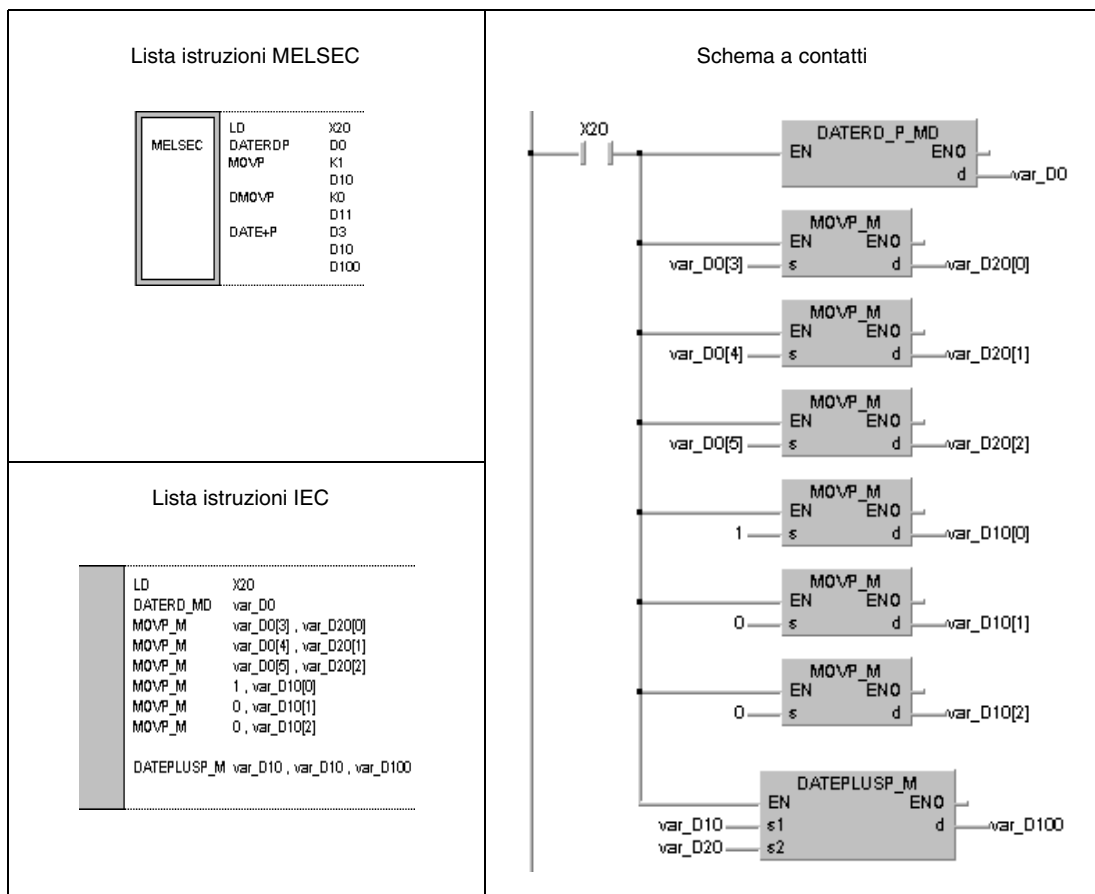
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il dato orologio da (s1)+0 a (s1)+2 e da (s2)+0 a (s2)+2 supera il campo d'ingresso ammesso.

Esempio di programma DATE+P

Con il fronte di salita di X20, il programma seguente legge i dati dall'orologio interno della CPU con l'istruzione DATERDP e li memorizza nei registri da D0 a D6 (vedi figura seguente). L'istruzione DATE+P somma un'ora (D10, D11, D12= ai dati letti. Il risultato viene memorizzato da D100 a D102 (vedi figura seguente).

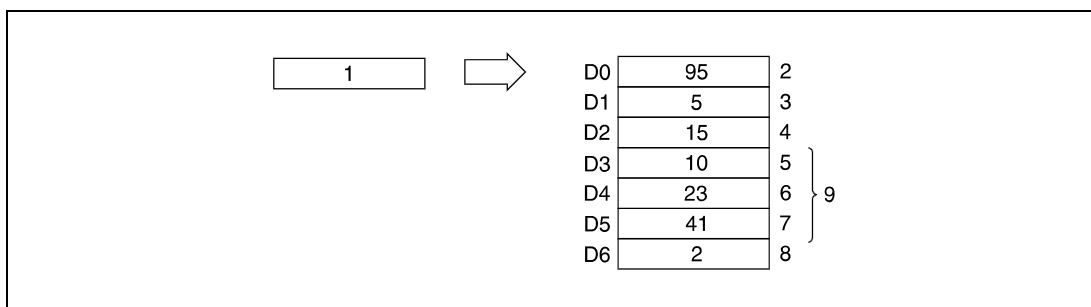
Dati orologio	Anno	Mese	Giorno	Ora	Minuto	Secondo	Giorno della settimana
Operandi	D0 (var_D0[0])	D1 (var_D0[1])	D2 (var_D0[2])	D3 (var_D0[3])	D4 (var_D0[4])	D5 (var_D0[5])	D6 (var_D0[6])
Operandi	—	—	—	D20 (var_D20[0])	D21 (var_D20[1])	D22 (var_D20[2])	—
Operandi	—	—	—	D10 (var_D10[0])	D11 (var_D10[1])	D12 (var_D10[2])	—
Operandi	—	—	—	D100 (var_D100[0])	D101 (var_D100[1])	D102 (var_D100[2])	—



NOTA

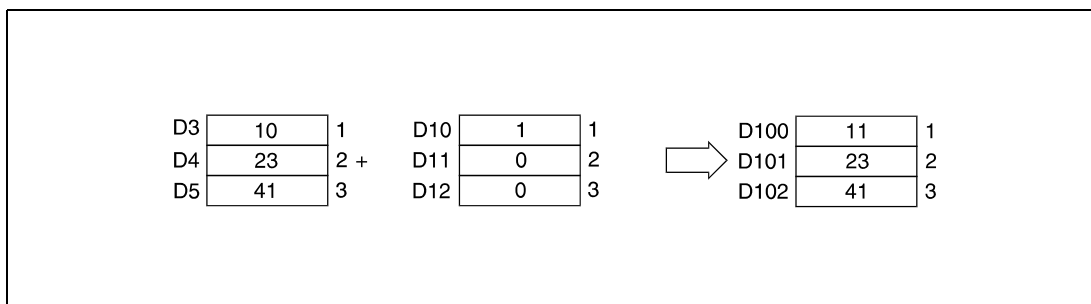
Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

La figura seguente mostra la lettura dei dati orologio con l'istruzione DATERDP.



- ¹ Orologio CPU QnA
- ² Anno
- ³ Maggio (Gennaio = 1, Dicembre = 12)
- ⁴ Giorno
- ⁵ Ora (formato 24 ore)
- ⁶ Minuto
- ⁷ Secondo
- ⁸ Giorno della settimana
- ⁹ Dati orologio

La figura seguente mostra la somma tramite l'istruzione DATE+P.



- ¹ Ora
- ² Minuto
- ³ Secondo

7.15.4 DATE-, DATE-P

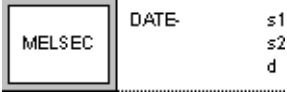
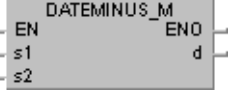
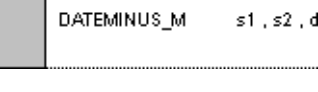
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

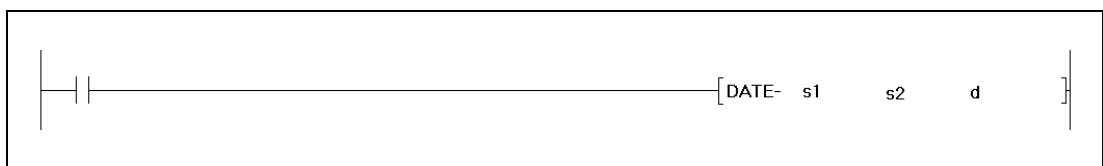
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro DY
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	SM0	4	
s2	—	●	●	—	—	—	—	—			
d	—	●	●	—	—	—	—	—			

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
--	---	---

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s1	Indirizzo del primo operando che contiene il dato orologio da cui si deve sottrarre.	BIN 16-bit
s2	Indirizzo del primo operando che contiene il dato orologio da sottrarre.	
d	Indirizzo del primo operando che memorizza i dati orologio che risultano dalla sottrazione.	

Funzioni **Sottrazione dati orologio**

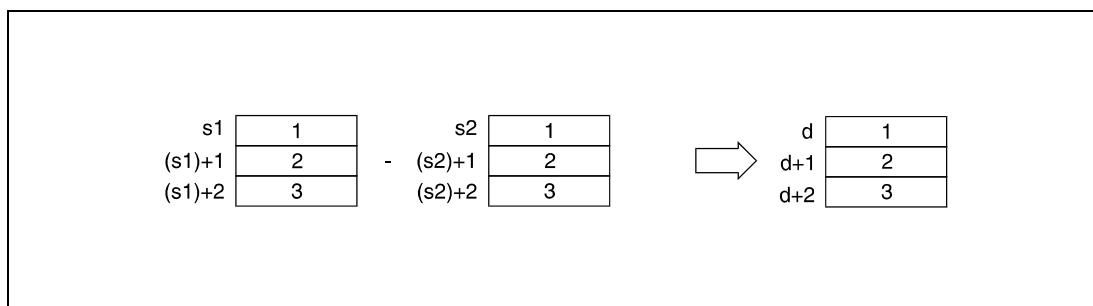
DATE- Istruzione di sottrazione

L'istruzione DATE- sottrae i dati orologio contenuti negli operandi indicati da s2 dai dati orologio contenuti negli operandi indicati da s1. I dati orologio che risultano dall'operazione vengono memorizzati negli operandi specificati da d.

La tabella seguente mostra i campi di valori dei dati orologio da (s1)+0 a (s1)+2, da (s2)+0 a (s2)+2, e da d+0 a d+2):

Dati orologio	Anno	Mese	Giorno	Ora	Minuto	Secondo	Giorno della settimana
Campo d'ingresso	—	—	—	0 - 23	0 - 59	0 - 59	—

Dati orologio	Anno	Mese	Giorno	Ora	Minuto	Secondo	Giorno della settimana
Operandi	—	—	—	s1+0 (array_s1[0])	s1+1 (array_s1[1])	s1+2 (array_s1[2])	—
Operandi	—	—	—	s2+0 (array_s2[0])	s2+1 (array_s2[1])	s2+2 (array_s2[2])	—
Operandi	—	—	—	d+0 (array_d[0])	d+1 (array_d[1])	d+2 (array_d[2])	—

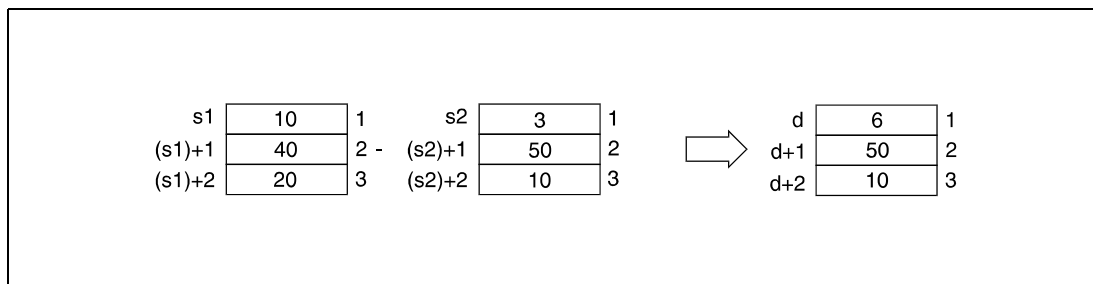


¹ Ora

² Minuto

³ Secondo

La figura seguente mostra la sottrazione di 3 ore, 50 minuti e 10 secondi ((s2)+0 - (s2)+2) da 10 ore, 40 minuti e 20 secondi ((s1)+0 - (s1)+2). Il risultato 6 ore, 50 minuti, 10 secondi viene memorizzato da d+0 a d+2.



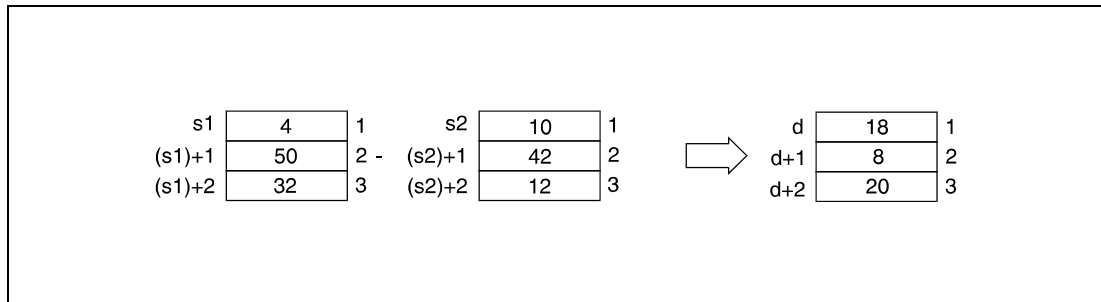
¹ Ora

² Minuto

³ Secondo

Se il risultato della sottrazione dei dati orologio diventa negativo, vengono sommate automaticamente 24 ore per ottenere un valore di tempo corretto.

La figura seguente mostra la sottrazione di 10 ore, 42 minuti e 12 secondi da 4 ore, 50 minuti e 32 secondi. Il risultato sarebbe -6 ore, 8 minuti e 20 secondi. Dato che questo risultato non è un formato orologio corretto, dopo la somma automatica di 24 ore, il risultato viene corretto in 18 ore, 8 minuti e 20 secondi (18:08:20 del giorno precedente).



¹ Ora

² Minuto

³ Secondo

NOTA

Fare riferimento alla sezione "Scrittura dati orologio" per ulteriori informazioni su questo argomento.

Errori di esecuzione

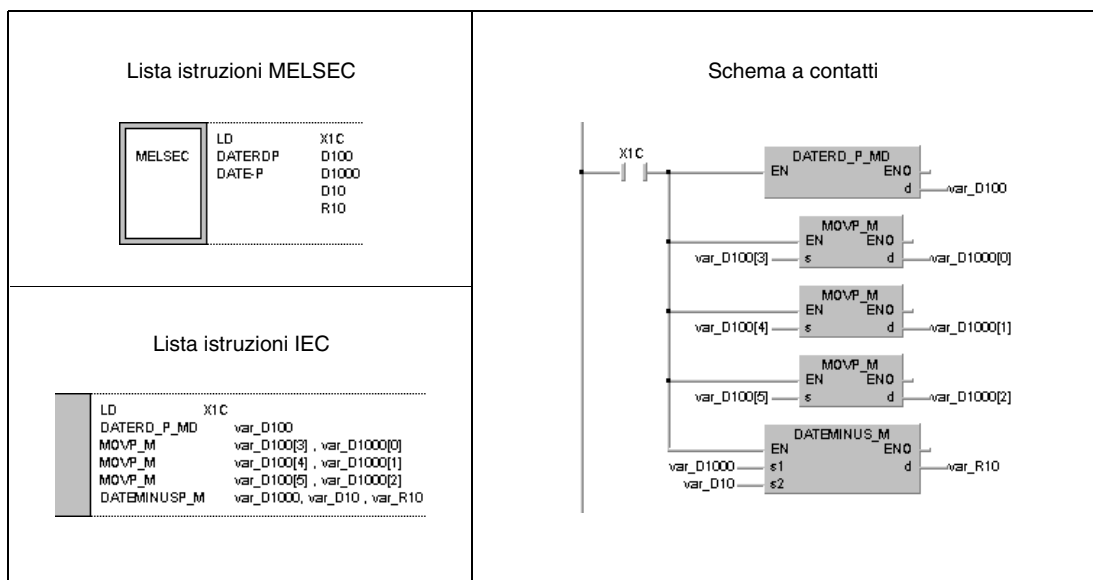
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il dato orologio da (s1)+0 a (s1)+2 e da (s2)+0 a (s2)+2 supera il campo d'ingresso ammesso.

Esempio di programma DATE-P

Con il fronte di salita di X1C, il programma seguente legge i dati dall'orologio interno della CPU con l'istruzione DATERDP e li memorizza nei registri da D100 a D106 (vedi figura seguente). L'istruzione DATE-P sottrae 10 ore (D10), 40 minuti (D11) e 10 secondi (D12) dai dati letti. Al risultato negativo della sottrazione, -8 ore, 41 minuti e 10 secondi vengono sommate 24 ore. Il risultato corretto, 16 ore, 41 minuti e 10 secondi (16:41:10 del giorno precedente) viene memorizzato da R10 a R12 (vedi figure seguenti).

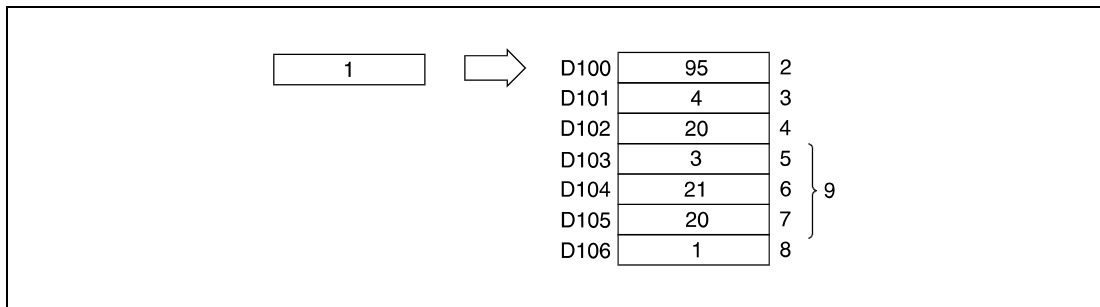
Dati orologio	Anno	Mese	Giorno	Ora	Minuto	Secondo	Giorno della settimana
Operandi	D100 (var_D100[0])	D101 (var_D100[1])	D102 (var_D100[2])	D103 (var_D100[3])	D104 (var_D100[4])	D105 (var_D100[5])	D106 (var_D100[6])
Operandi	—	—	—	D1000 (var_D1000[0])	D1001 (var_D1000[1])	D1002 (var_D1000[2])	—
Operandi	—	—	—	D10 (var_D10[0])	D11 (var_D10[1])	D12 (var_D10[2])	—
Operandi	—	—	—	R10 (var_R10[0])	R11 (var_R10[1])	R12 (var_R10[2])	—



NOTA

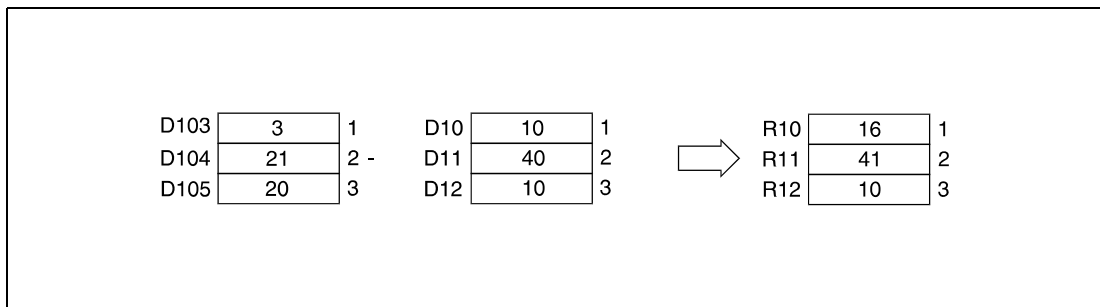
Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

La figura seguente mostra la lettura dei dati orologio con l'istruzione DATERDP.



- ¹ Orologio CPU QnA
- ² Anno
- ³ Maggio (Gennaio = 1, Dicembre = 12)
- ⁴ Giorno
- ⁵ Ora (formato 24 ore)
- ⁶ Minuto
- ⁷ Secondo
- ⁸ Giorno della settimana
- ⁹ Dati orologio

La figura seguente mostra la sottrazione tramite l'istruzione DATE-P.



- ¹ Ora
- ² Minuto
- ³ Secondo

7.15.5 SECOND, SECONDP, HOUR, HOURP

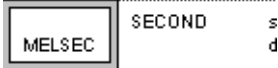
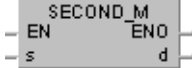
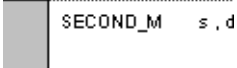
CPU

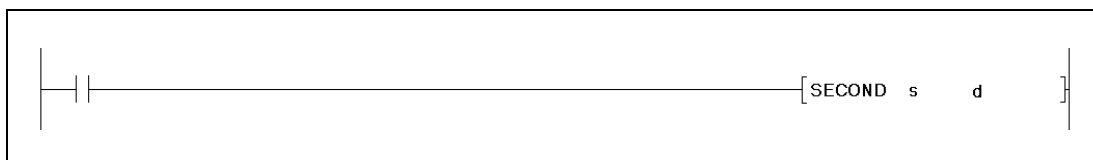
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi
MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
SECOND											
s	—	●	●	—	—	—	—	—	—	SM0	3
d	●	●	●	●	●	●	●	—	—		
HOUR											
s	●	●	●	●	●	●	●	●	—	SM0	3
d	—	●	●	—	—	—	—	—	—		

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
		

GX
Developer

Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
SECOND			
s	Ore, minuti, secondi	BIN 16-/32-bit	Array [0.0.2] di ANY16
d	Secondi		ANY32
HOUR			
s	Secondi	BIN 16-/32-bit	ANY32
d	Ore, minuti, secondi		Array [0.0.2] di ANY16

Funzioni Modifica formato orologio

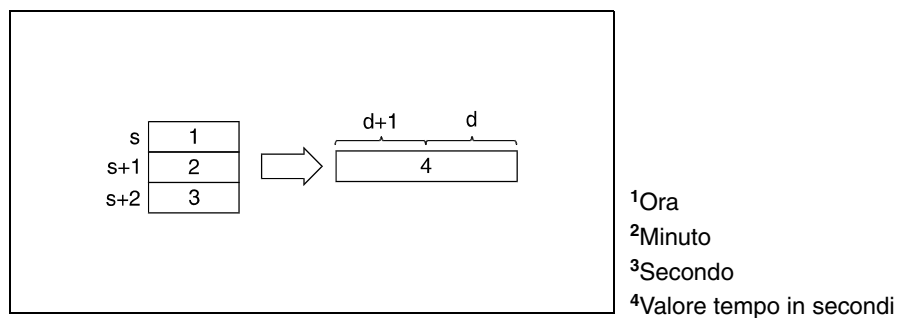
SECOND Modifica formato ora da hh:mm:ss a secondi

L'istruzione SECOND modifica i dati orologio contenuti negli operandi da s+0 a s+2 dal formato hh:mm:ss al formato in soli secondi. Il risultato viene memorizzato negli operandi specificati da d e d+1.

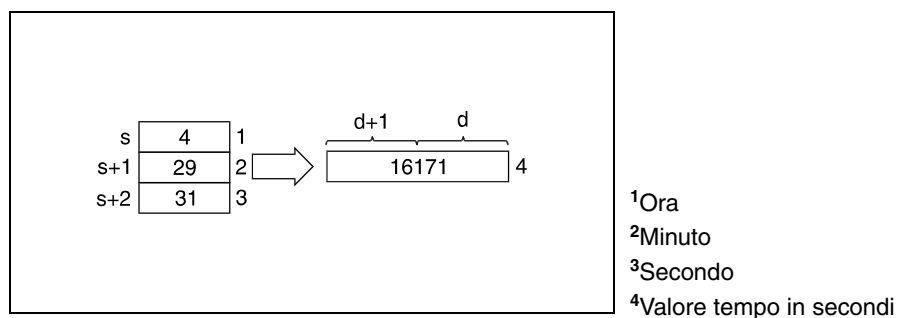
La tabella seguente mostra i campi di ingresso dei dati orologio contenuti da s+0 a s+2:

Dati orologio	Anno	Mese	Giorno	Ora	Minuto	Secondo	Giorno della settimana
Campo d'ingresso	—	—	—	0 - 23	0 - 59	0 - 59	—

Dati orologio	Anno	Mese	Giorno	Ora	Minuto	Secondo	Giorno della settimana
Operandi	—	—	—	s+0 (array_s[0])	s+1 (array_s[1])	s+2 (array_s[2])	—
Operandi	—	—	—	—	—	d+0 (array_d[0]) a d+1 (array_d[1])	—



La figura seguente mostra la conversione di 4 ore, 29 minuti e 31 secondi in 16171 secondi.



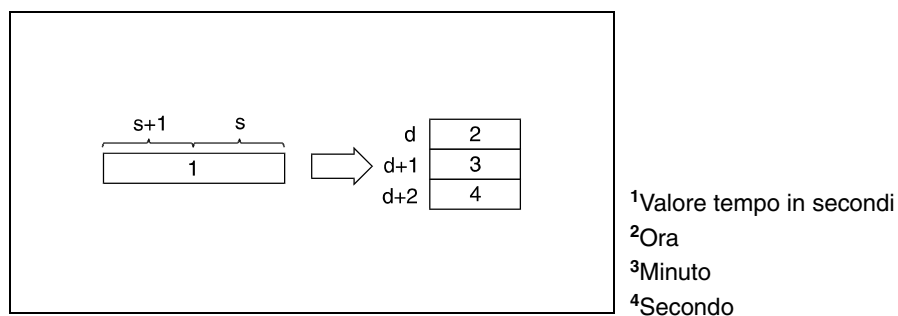
HOUR Modifica formato ora da secondi a hh:mm:ss

L'istruzione HOUR modifica i dati orologio contenuti negli operandi da s+0 a s+2 dal formato in soli secondi al formato hh:mm:ss.

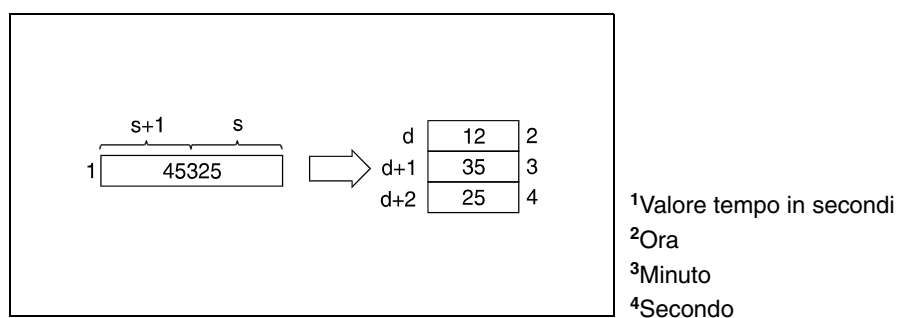
La tabella seguente mostra i campi di ingresso dei dati orologio contenuti da d+0 a d+2:

Dati orologio	Anno	Mese	Giorno	Ora	Minuto	Secondo	Giorno della settimana
Campo d'ingresso	—	—	—	0 - 23	0 - 59	0 - 59	—

Dati orologio	Anno	Mese	Giorno	Ora	Minuto	Secondo	Giorno della settimana
Operandi	—	—	—	d+0 (array_d[0])	d+1 (array_d[1])	d+2 (array_d[2])	—
Operandi	—	—	—	—	—	s+0 (array_s[0]) a s+1 (array_s[1])	—



La figura seguente mostra la conversione di 45325 secondi, 12 ore, 35 minuti e 25 secondi.



Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

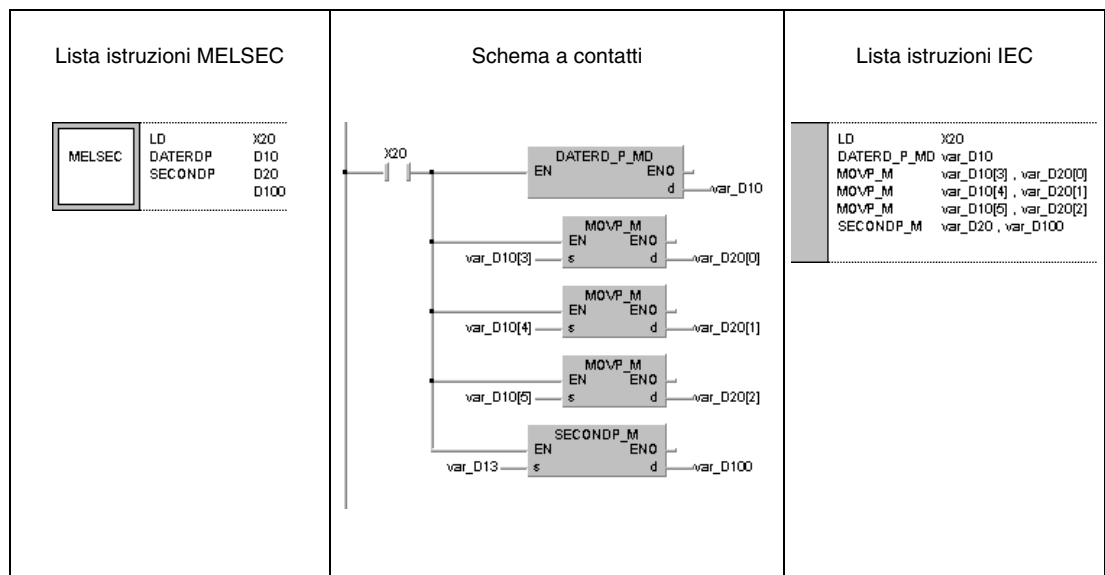
- I dati orologio da s+0 (array_s[0]) a s+2 (array_s[2]) per l'istruzione SECOND, o in s+0 e s+1 per l'istruzione HOUR supera il campo ammesso (codice di errore 4100).

Programma di esempio 1

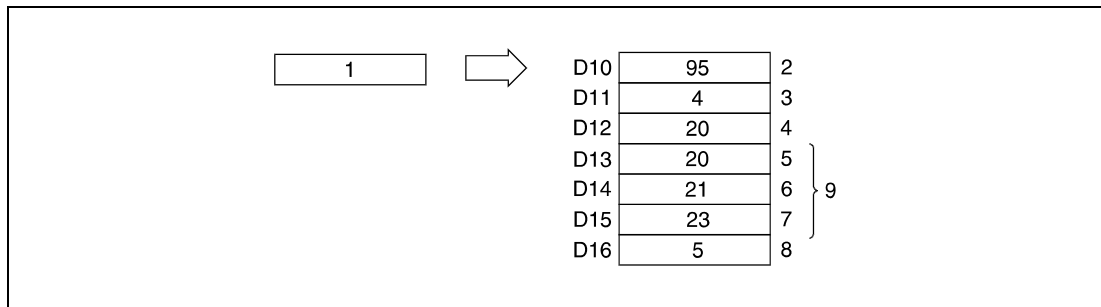
SECONDP

Con il fronte di salita di X20, il programma seguente legge i dati dall'orologio interno della CPU con l'istruzione DATERDP e li memorizza nei registri da D10 a D16 (vedi figura seguente). Le ore D20, minuti D21, e secondi D22 dei dati orologio vengono convertiti in secondi tramite l'istruzione SECONDP. Il risultato viene memorizzato da D100 a D101 (vedi figure seguenti).

Dati orologio	Anno	Mese	Giorno	Ora	Minuto	Secondo	Giorno della settimana
Operandi	D10 (var_D10[0])	D11 (var_D10[1])	D12 (var_D10[2])	D13 (var_D10[3])	D14 (var_D10[4])	D15 (var_D10[5])	D16 (var_D10[6])
Operandi	—	—	—	D20 (var_D20[0])	D21 (var_D20[1])	D22 (var_D20[2])	—
Operandi	—	—	—	—	—	D100 (var_D10[0]) bis D101 (var_D10[1])	—

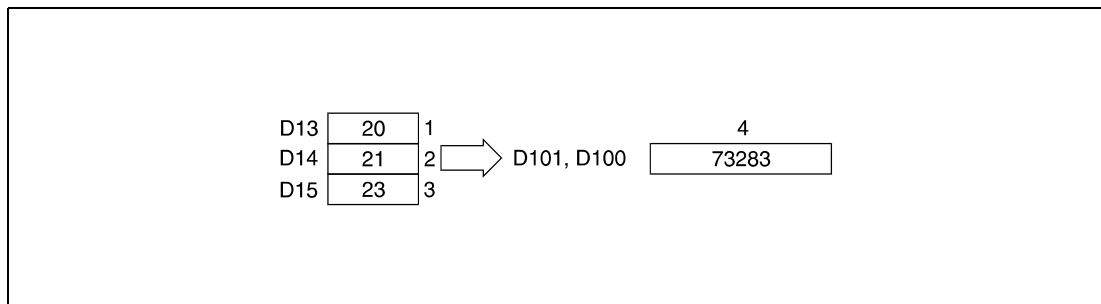


La figura seguente mostra la lettura dei dati orologio con l'istruzione DATERDP.



- ¹ Orologio CPU QnA
- ² Anno
- ³ Maggio (Gennaio = 1, Dicembre = 12)
- ⁴ Giorno
- ⁵ Ora (formato 24 ore)
- ⁶ Minuto
- ⁷ Secondo
- ⁸ Giorno della settimana
- ⁹ Dati orologio

La figura seguente mostra la conversione in secondi tramite l'istruzione SECONDP.

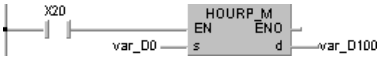


- ¹ Ora
- ² Minuto
- ³ Secondo
- ⁴ Secondi convertiti

Programma di esempio 2 HOURP

Con il fronte positivo di X20, il programma seguente converte i secondi contenuti in D0 e D1 in ore, minuti, secondi. Il risultato viene memorizzato negli operandi racchiusi in parentesi.

Dati orologio	Anno	Mese	Giorno	Ora	Minuto	Secondo	Giorno della settimana
Operandi	—	—	—	D0 (var_D0[1])	D1 (var_D0[2])	D2 (var_D0[3])	—
Operandi	—	—	—	—	—	D100 (var_D100[0]) a D101 (var_D100[1])	—

<p style="text-align: center;">Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="border: 1px solid black; padding: 2px;">MELSEC</td> <td style="padding: 2px;">LD</td> <td style="padding: 2px;">X20</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;"></td> <td style="padding: 2px;">HOURP</td> <td style="padding: 2px;">D0 D100</td> </tr> </table> </div>	MELSEC	LD	X20		HOURP	D0 D100	<p>Schema a contatti</p> 	<p style="text-align: center;">Lista istruzioni IEC</p> <div style="border: 1px solid black; padding: 5px; margin: 5px 0;"> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">LD</td> <td style="padding: 2px;">X20</td> </tr> <tr> <td style="padding: 2px;">HOURP_M</td> <td style="padding: 2px;">var_D0 , var_D100</td> </tr> </table> </div>	LD	X20	HOURP_M	var_D0 , var_D100			
MELSEC	LD	X20													
	HOURP	D0 D100													
LD	X20														
HOURP_M	var_D0 , var_D100														
<table style="margin: auto;"> <tr> <td style="text-align: right;">D1, D0</td> <td style="border: 1px solid black; padding: 5px; text-align: center;">1 40000</td> <td style="font-size: 2em; vertical-align: middle;">➔</td> <td style="border: 1px solid black; padding: 5px;"> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">D100</td> <td style="padding: 2px; text-align: center;">11</td> <td style="padding: 2px;">2</td> </tr> <tr> <td style="padding: 2px;">D101</td> <td style="padding: 2px; text-align: center;">6</td> <td style="padding: 2px;">3</td> </tr> <tr> <td style="padding: 2px;">D102</td> <td style="padding: 2px; text-align: center;">40</td> <td style="padding: 2px;">4</td> </tr> </table> </td> </tr> </table>			D1, D0	1 40000	➔	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">D100</td> <td style="padding: 2px; text-align: center;">11</td> <td style="padding: 2px;">2</td> </tr> <tr> <td style="padding: 2px;">D101</td> <td style="padding: 2px; text-align: center;">6</td> <td style="padding: 2px;">3</td> </tr> <tr> <td style="padding: 2px;">D102</td> <td style="padding: 2px; text-align: center;">40</td> <td style="padding: 2px;">4</td> </tr> </table>	D100	11	2	D101	6	3	D102	40	4
D1, D0	1 40000	➔	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">D100</td> <td style="padding: 2px; text-align: center;">11</td> <td style="padding: 2px;">2</td> </tr> <tr> <td style="padding: 2px;">D101</td> <td style="padding: 2px; text-align: center;">6</td> <td style="padding: 2px;">3</td> </tr> <tr> <td style="padding: 2px;">D102</td> <td style="padding: 2px; text-align: center;">40</td> <td style="padding: 2px;">4</td> </tr> </table>	D100	11	2	D101	6	3	D102	40	4			
D100	11	2													
D101	6	3													
D102	40	4													

- ¹ Valore da convertire in secondi
- ² Ora
- ³ Minuto
- ⁴ Secondo

NOTA

Questo programma non funziona senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

7.16 Istruzioni per dispositivi periferici

Le istruzioni per dispositivi periferici consentono l'emissione di messaggi verso dispositivi periferici e il ricevimento di dati da tastiere connesse tramite dispositivi periferici.

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Invio messaggi a dispositivi periferici	MSG	MSG_M
Ingresso dati tastiera da dispositivo periferico	PKEY	PKEY_M

7.16.1 MSG



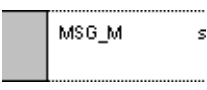
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti \$			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	●	—	—	2

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
--	--	---

GX Developer

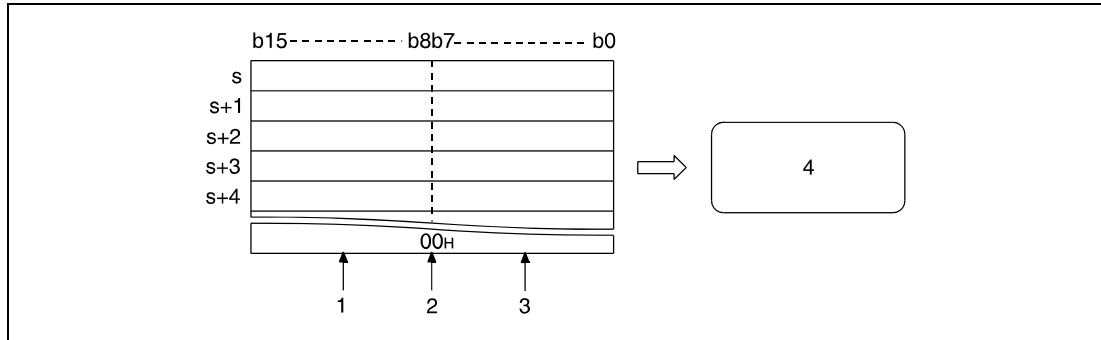


Variabili

Dati impostati	Significato	Tipo dati
s	Stringa di caratteri da visualizzare sul dispositivo periferico o indirizzo del primo operando che la contiene.	Stringa caratteri

Funzioni **Invio messaggi a dispositivi periferici****MSG** **Istruzione di uscita**

L'istruzione MSG invia una stringa di caratteri memorizzata nell'operando indicato da s, verso il dispositivo periferico specificato in modo terminale. La fine della stringa di caratteri è indicata dal codice carattere "00H".



¹ carattere 2., 4.,..., (n+1)

² Il codice "00H" indica la fine della stringa di caratteri.

³ carattere 1., 3., ..., n

⁴ Visualizzazione di stringhe di caratteri (messaggi) su dispositivo periferico

Si possono visualizzare fino a 64 caratteri sul dispositivo periferico.

La stringa di caratteri contenuta a partire da s viene trasferita nei registri speciali da SD738 a SD773 (area memorizzazione messaggi).

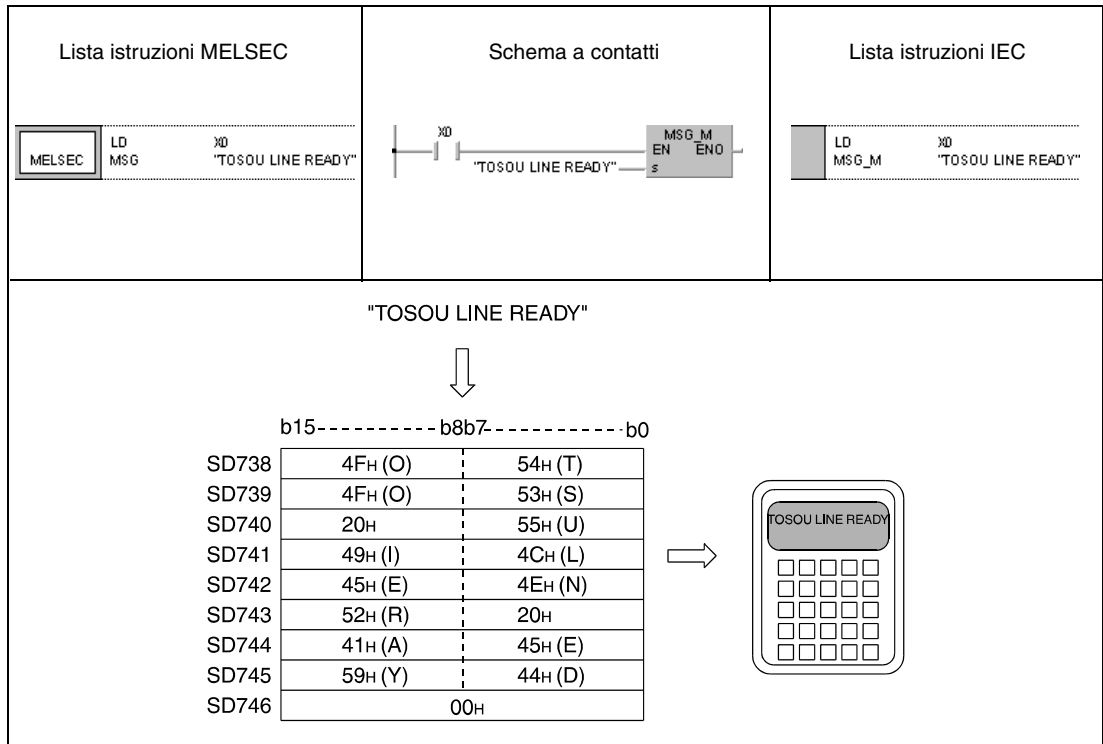
Durante l'esecuzione dell'istruzione MSG il relé speciale SM738 (segnale di esecuzione dell'istruzione MSG) è attivo. Se il relé speciale SM738 è attivo, nessun'altra istruzione MSG viene eseguita.

Al termine dell'istruzione MSG, cioè dopo la visualizzazione di tutti i caratteri, il registro speciale SM738 viene azzerato ed il contenuto (stringa di caratteri) nei registri speciali da SD738 a SD773 viene cancellato (sovrascritto dal codice carattere "00H").

Esempio di programma

MSG

Se X0 è attivo, il programma seguente invia e visualizza la stringa di caratteri "TOSOU LINE READY" come messaggio per il display del dispositivo periferico.



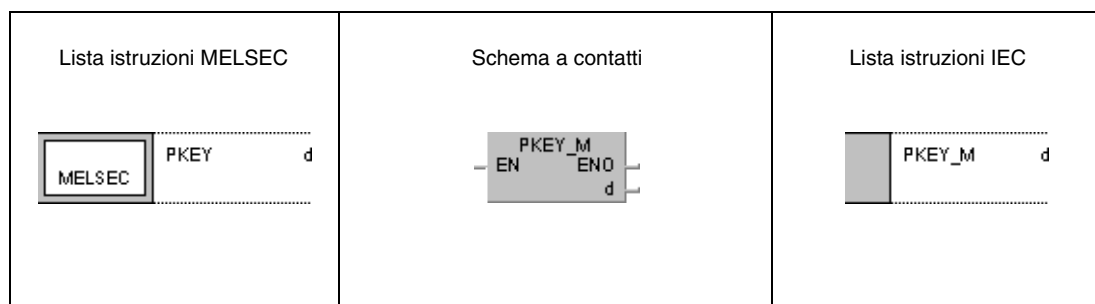
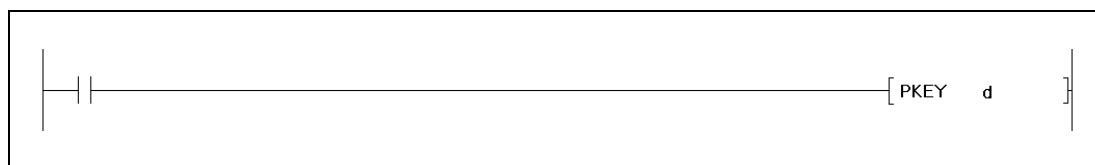
7.16.2 PKEY

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi
MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro U
	Bit	Word		Bit	Word						
d	—	●	●	—	—	—	—	●	—	SM0	2

GX IEC
DeveloperGX
Developer

Variabili

Dati impostati	Significato	Tipo dati
d	Indirizzo del primo operando che memorizza la stringa di caratteri d'ingresso.	BIN 16-bit

Funzioni Ingresso tastiera da dispositivo periferico

PKEY Istruzione di ingresso

L'istruzione PKEY cancella le word dati degli operandi da d+0 a d+17 e attiva il relé speciali SM376 (segnale di esecuzione per l'istruzione PKEY). Viene inoltre attivato il relé speciale SM737 (flag ricezione tasto). Al termine dell'istruzione PKEY i dati di ingresso della tastiera (caratteri) vengono letti dal dispositivo periferico specificato in modo terminale e scritti in formato ASCII sugli operandi indicati da d+0 a d+17.

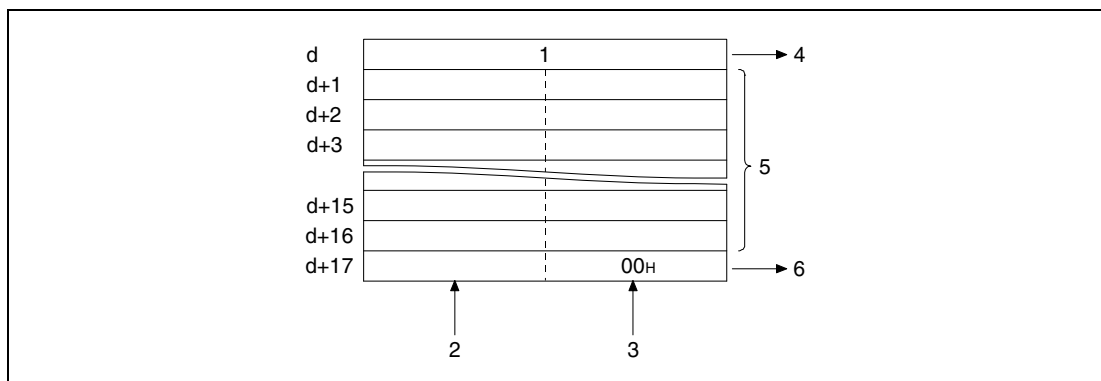
La disattivazione della condizione di esecuzione dell'istruzione PKEY disattiva anche i relé speciali SM736 e SM737.

Il relé speciale SM737 si attiva se il carattere inviato dalla tastiera viene ricevuto dal dispositivo periferico, mentre viene disattivato se la CPU memorizza il tasto ricevuto. Il dispositivo periferico non può ricevere i dati in ingresso dalla tastiera per tutto il tempo che il relé speciale SM737 è attivo.

L'ingresso da tastiera sul dispositivo periferico viene considerato terminato, quando si riceve il carattere "CR".

Si possono inserire al massimo 32 caratteri. Dopo aver ricevuto 32 caratteri, la ricezione da tastiera viene terminata dal dispositivo periferico, anche senza aver ricevuto il carattere "CR".

La figura seguente mostra la memorizzazione dei dati da tastiera (caratteri) negli operandi da d+1 a d+17:



¹ Contatore

² caratteri da 2 a 32

³ caratteri da 1 a 31

⁴ Numero caratteri inseriti (valore binario)

⁵ Massimo 16 caratteri

⁶ Il codice "00H" indica la fine della stringa di caratteri.

(numero caratteri inseriti: dispari = byte pesante, pari = byte leggero)

L'istruzione PKEY non può essere eseguita contemporaneamente su più locazioni. Se si desidera ricevere dati da tastiera da più locazioni, si deve predisporre un interblocco tramite il relé SM376 (condizione di esecuzione dell'istruzione PKEY) per evitare la ricezione simultanea

Errori di esecuzione

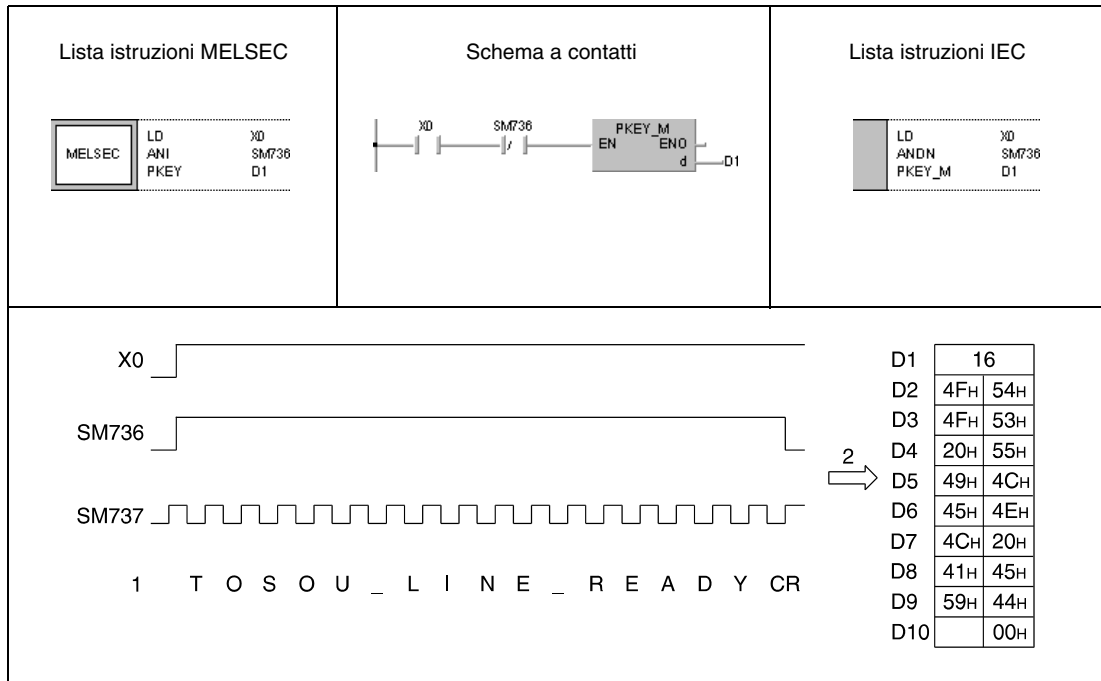
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- I dati da tastiera inseriti superano il campo ammesso per gli operandi da d+0 a d+17 (codice di errore 4101).

Esempio di programma

PKEY

Se X0 è attivo, il programma seguente memorizza la stringa di caratteri "TOSOU LINE READY" introdotta da tastiera nel dispositivo periferico, nei registri da D1 a D10.



¹ Dati tastiera in ingresso

² Memorizzazione dati inseriti

7.17 Istruzioni controllo di programma

Le istruzioni per il controllo del programma commutano fra i diversi modi operativi. La tabella che segue offre una panoramica di queste istruzioni:

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Commutazione programmi in modo standby	PSTOP	PSTOP_M
	PSTOPP	PSTOPP_M
Commutazione programmi in modo standby e azzeramento uscite	POFF	POFF_M
	POFFP	POFFP_M
Commutazione programmi in modo esecuzione scansione	PSCAN	PSCAN_M
	PSCANP	PSCANP_M
Commutazione programmi in modo esecuzione a bassa velocità	PLOW	PLOW_M
	PLOWP	PLOWP_M

NOTA

Controllare se queste funzioni sono disponibili e supportate dalla versione di GX IEC Developer utilizzata.

7.17.1 PSTOP, PSTOPP**CPU**

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

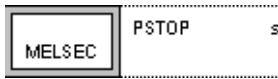

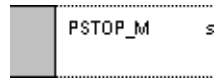
¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

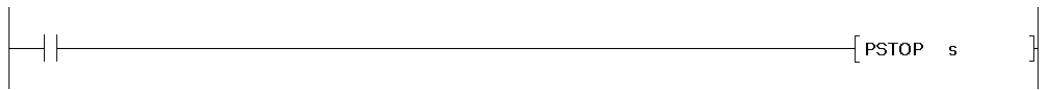
	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti \$			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	●	—	SM0	2 + n ¹⁾

¹ n = (numero di caratteri del nome programma)/2 = numero di passi aggiuntivi (frazioni decimali arrotondate per eccesso)

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
		

GX Developer


--

Variabili

Dati impostati	Significato	Tipo dati
s	Nome del file di programma da commutare in modo standby o indirizzo del primo operando che lo contiene.	Stringa caratteri

Funzioni **Impostazione programma in modo standby**

PSTOP Istruzione di commutazione nel modo standby

L'istruzione PSTOP imposta il programma specificato dall'operando s nel modo standby. In questa modalità il programma viene eseguito solo a richiesta.

Possono essere impostati in modo standby solo i programmi contenuti nella memoria interna (drive 0).

Il modo standby viene attivato solo dopo l'esecuzione dell'istruzione END.

L'istruzione PSTOP è prioritaria rispetto alla modalità di esecuzione specificata dai parametri.

Non è necessario specificare l'estensione .QPG, dato che il tipo di file viene riconosciuto automaticamente.

Errori di esecuzione

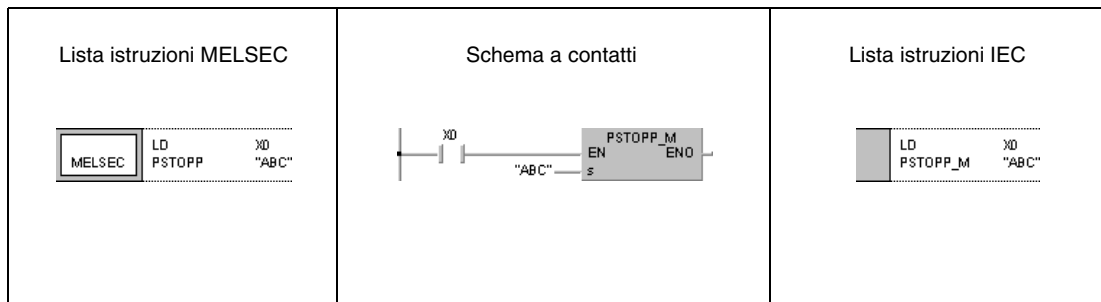
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il file di programma specificato non esiste (codice di errore 2410).

Esempio di programma

PSTOPP

Con il fronte positivo di X0, il programma seguente commuta il programma di nome "ABC" nel modo standby.



7.17.2 POFF, POFFP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

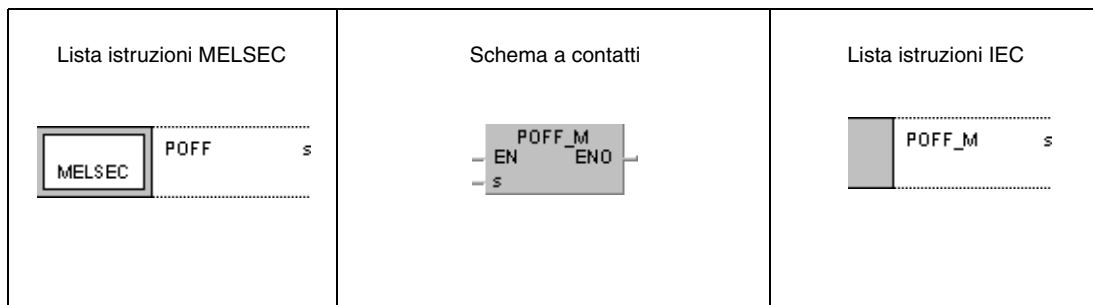
¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

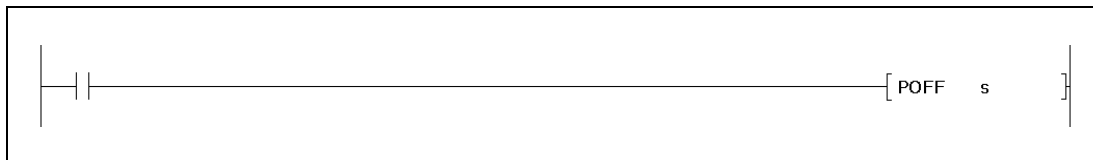
	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti \$			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	●	—	SM0	2 + n ¹⁾

¹ n = (numero di caratteri del nome programma)/2 = numero di passi aggiuntivi (frazioni decimali arrotondate per eccesso)

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando contenente il nome file da commutare in modo standby con azzeramento delle uscite.	Stringa caratteri

Funzioni **Commutazione di un programma in modo standby con azzeramento delle uscite**

POFF **Istruzione di commutazione nel modo standby con azzeramento uscite**

L'istruzione POFF dispone il programma specificato dall'operando s in modo stand-by e azzerava le uscite comandate dal programma. In questa modalità tutte le uscite utilizzate dal programma vengono impostate nello stesso stato assunto quando le rispettive condizioni di esecuzione non sono attive. Il programma entra poi nel modo standby.

Possono essere impostati in modo standby solo i programmi contenuti nella memoria interna (drive 0).

Il modo standby viene attivato solo dopo l'esecuzione dell'istruzione END.

L'istruzione POFF è prioritaria rispetto alla modalità di esecuzione specificata dai parametri.

Non è necessario specificare l'estensione .QPG, dato che il tipo di file viene riconosciuto automaticamente.

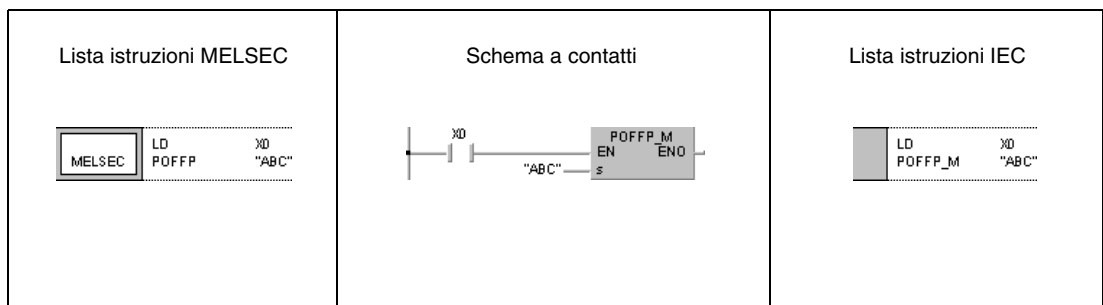
NOTA *A seguito dell'esecuzione dell'istruzione POFF, le bobine oggetto di una istruzione OUT vengono azzerate (vedi funzioni).*

Errori di esecuzione Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il file di programma specificato non esiste (codice di errore 2410).

Esempio di programma **POFFP**

Con il fronte positivo di X0, il programma seguente commuta il programma di nome "ABC" nel modo standby. In questa modalità tutte le uscite utilizzate dal programma "ABC" vengono impostate nello stesso stato assunto quando le rispettive condizioni di esecuzione non sono attive. Il programma "ABC" entra poi nel modo standby.



7.17.3 PSCAN, PSCANP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

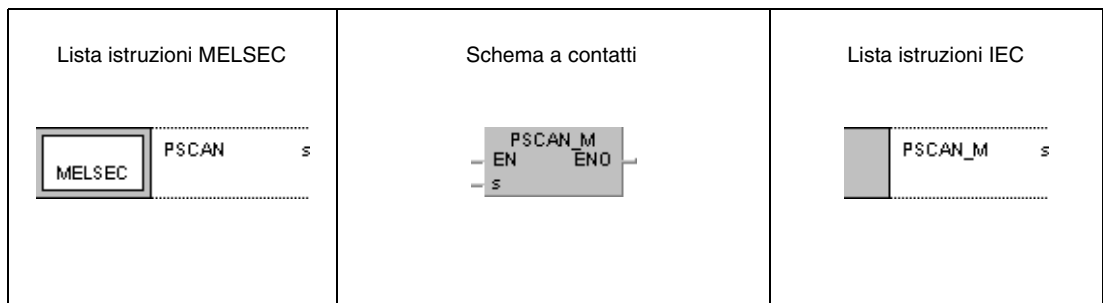
¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

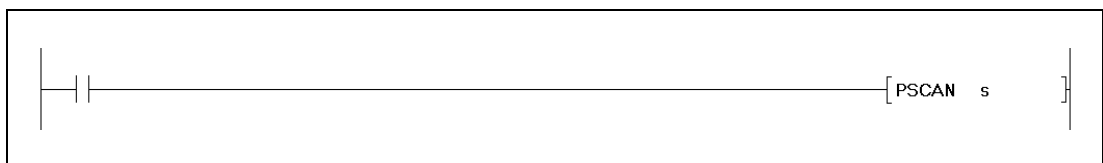
	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti \$			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	●	—	S00	2 + n ¹⁾

¹ n = (numero di caratteri del nome programma)/2 = numero di passi aggiuntivi (frazioni decimali arrotondate per eccesso)

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Nome del file di programma da commutare in modo scansione o indirizzo del primo operando che lo contiene.	Stringa caratteri

Funzioni **Impostazione programma in modo scansione**

PSCAN Istruzione di commutazione nel modo scansione

L'istruzione PSCAN imposta il programma specificato dell'operando indicato da s nel modo esecuzione scansione. In questo modo il programma viene eseguito solo una volta per ogni scansione.

Possono essere impostati in modo scansione solo i programmi contenuti nella memoria interna (drive 0).

Il modo scansione viene attivato solo dopo l'esecuzione dell'istruzione END.

L'istruzione PSCAN è prioritaria rispetto alla modalità di esecuzione specificata dai parametri.

Non è necessario specificare l'estensione .QPG, dato che il tipo di file viene riconosciuto automaticamente.

Errori di esecuzione

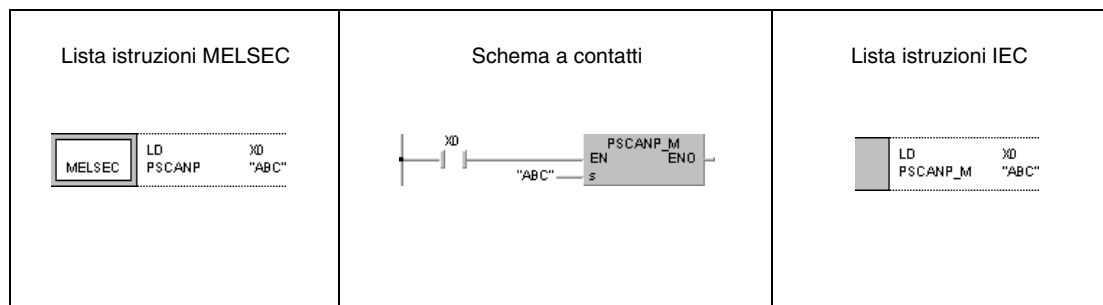
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il file di programma specificato non esiste (codice di errore 2410).

Esempio di programma

PSCANP

Con il fronte positivo di X0, il programma seguente commuta il programma di nome "ABC" nel modo scansione.



7.17.4 PLOW, PLOWP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹


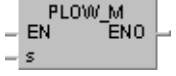
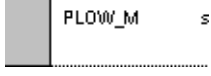
¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

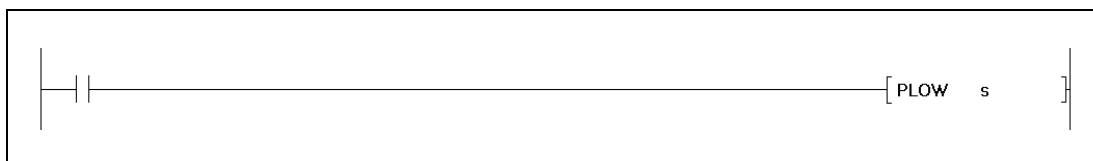
	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti \$			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	●	—	SM0	2 + n ¹⁾

¹ n = (numero di caratteri del nome programma)/2 = numero di passi aggiuntivi (frazioni decimali arrotondate per eccesso)

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
		

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Nome del file di programma da commutare in modo bassa velocità o indirizzo del primo operando che lo contiene.	Stringa caratteri

Funzioni **Impostazione programma in modo bassa velocità**

PLOW Istruzione di commutazione nel modo esecuzione a bassa velocità

L'istruzione PLOW imposta il programma specificato dell'operando indicato da s nel modo esecuzione lenta. In questa modalità il programma viene eseguito solo a bassa velocità.

Possono essere impostati in modo scansione solo i programmi contenuti nella memoria interna (drive 0).

Il modo bassa velocità viene attivato solo dopo l'esecuzione dell'istruzione END.

L'istruzione PLOW è prioritaria rispetto alla modalità di esecuzione specificata dai parametri.

Non è necessario specificare l'estensione .QPG, dato che il tipo di file viene riconosciuto automaticamente.

Errori di esecuzione

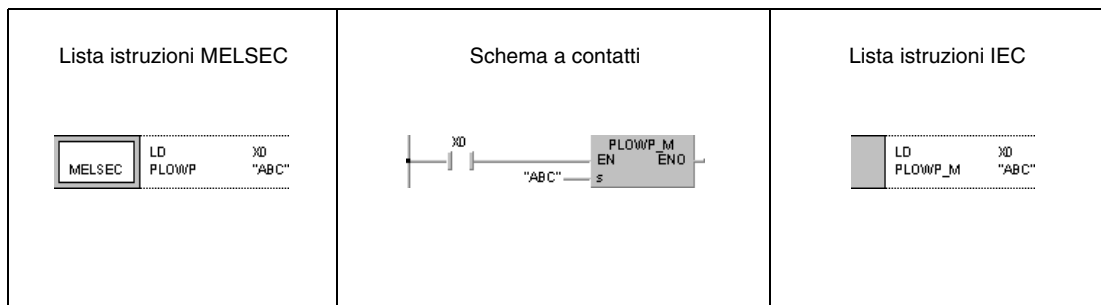
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il file di programma specificato non esiste (codice di errore 2410).
- Il file di programma contiene una istruzione CHK (codice di errore 4235).

Esempio di programma

PLOWP

Con il fronte positivo di X0, il programma seguente commuta il programma di nome "ABC" nel modo esecuzione a bassa velocità.



7.18 Altre istruzioni utili

Questa sezione contiene istruzioni per attivare e disattivare il timer WDT di watchdog, il segnale di riporto, per impostare il numero di scansioni di programma da eseguire, per la lettura, scrittura ed inserimento dati da e per diverse memorie. La tabella che segue offre una panoramica di queste istruzioni:

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Cancella timer di watchdog	WDT	WDT_M
Attivazione e disattivazione segnale di riporto	STC	STC_M
	CLC	CLC_M
Imposta il numero delle scansioni in esecuzione	DUTY	DUTY_M
Lettura diretta di un byte	ZRRDB	ZRRDB_M
	ZRRDBP	ZRRDBP_M
Scrittura diretta di un byte	ZRWRB	ZRWRB_M
	ZRWRBP	ZRWRBP_M
Memorizza operando per indirizzamento indiretto	ADRSET	ADRSET_M
	ADRSETP	ADRSETP_M
Ingresso tasto numerico da tastiera	KEY	KEY_MD
Salvataggio contenuto gruppo registri indice	ZPUSH	ZPUSH_M
	ZPUSHP	ZPUSHP_M
Ripristino contenuto gruppo registri indice	ZPOP	ZPOP_M
	ZPOPP	ZPOPP_M
Scrittura in blocco di dati su EEPROM	EROMWR	EROMWR_M
	EROMWRP	EROMWRP_M

NOTA

Le istruzioni ADRSET e ADRSETP non sono supportate da GX IEC Developer.

7.18.1 WDT, WDTP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

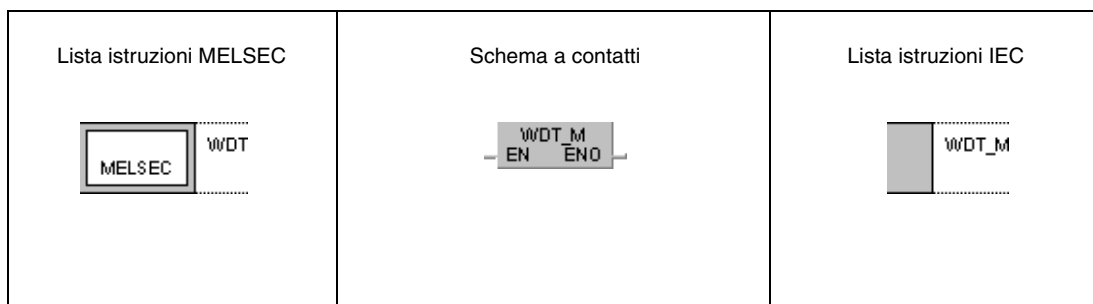
Operandi utilizzabili																	Definizione cifra	Numero di passi	Indice	Riporto	Errore				
Operandi a bit							Operandi a word (16-bit)							Costanti		Puntatore						Livello			
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H (16#)	P	I	N				M9012	M9011
																						1	●		

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale di programmazione per il corrispondente numero di passi.

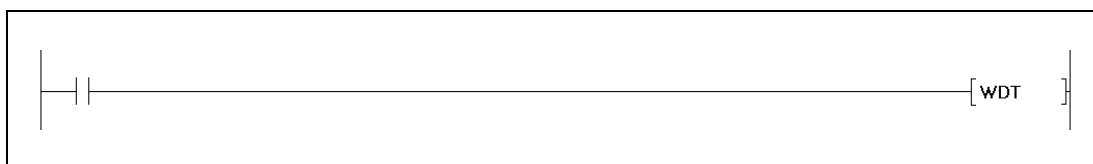
Operandi MELSEC Q

Operandi utilizzabili										Errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti	Altro			
Bit	Word		Bit	Word							
—	—	—	—	—	—	—	—	—	—	—	1

GX IEC Developer



GX Developer



Variabili

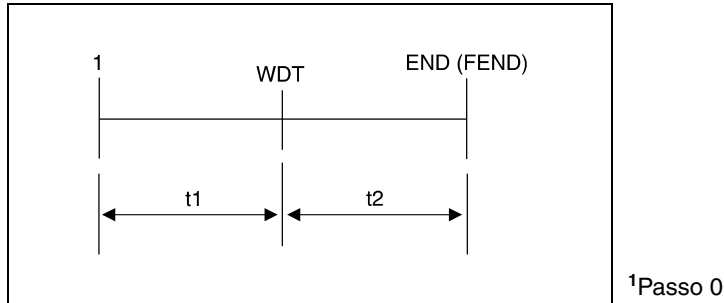
Dati impostati	Significato	Tipo dati
—	—	—

Funzioni **Disattivazione timer di watchdog**
WDT **Cancellazione**

L'istruzione WDT cancella il timer di watchdog (WDT) durante l'esecuzione di un programma di sequenza.

L'istruzione WDT è necessaria solo se il tempo di scansione del programma dal passo 0 all'istruzione END/FEND supera il tempo WDT impostato in determinate condizioni. Se il tempo WDT impostato viene superato ad ogni scansione, è necessario correggere adeguatamente questo tempo.

Il valore impostato deve essere tale per cui i periodi t1 (da passo 0 a istruzione WDT) o t2 (da istruzione WDT a istruzione END/FEND) non superino il tempo WDT impostato.



L'istruzione WDT può essere utilizzata un numero di volte qualsiasi in una scansione di programma. Durante la programmazione occorre comunque considerare che le uscite non vengono azzerate in caso di errore.

I valori del tempo di scansione programma memorizzati nei registri non vengono azzerati con l'istruzione WDT. Quindi i valori memorizzati possono essere superiori ai valori WDT impostati con i parametri.

NOTA *Le CPU serie A seguenti supportano solo valori fissi per i timer di watchdog:
A3H, A3M, AnA, AnAS, e AnU*

7.18.2 STC, CLC

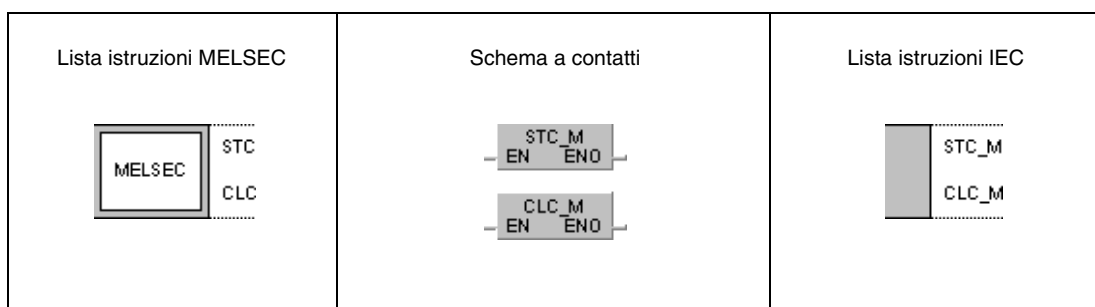
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●		

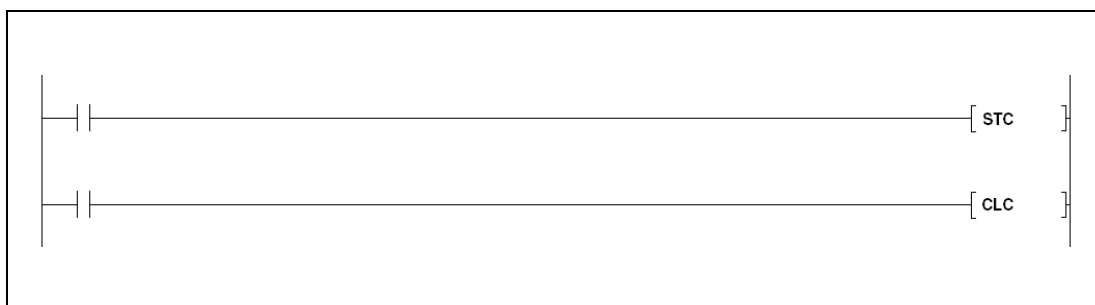
Operandi MELSEC A

Operandi utilizzabili																	Definizione cifra	Numero di passi	Indice	Riporto	Errore				
Operandi a bit							Operandi a word (16-bit)							Costanti		Puntatore						Livello			
X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V	K	H (16#)	P	I	N				M9012	M9010 M9011
																						1			

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
—	—	—

Funzioni Attivazione e disattivazione segnale di riporto

STC Attiva segnale di riporto

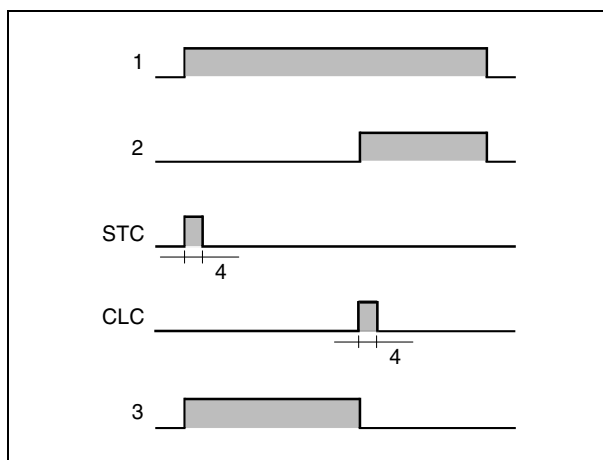
Il flag di riporto memorizza il riporto (0 o 1) delle operazioni di scorrimento e rotazione. Il riporto viene rappresentato nel programma come un contatto del relé speciale M9012. M9012 è attivo se il segnale di riporto è 1; non attivo se il segnale di riporto è 0.

A seguito dell'esecuzione dell'istruzione STC il segnale di riporto (M9012) viene forzato ON.

CLC Disattiva segnale di riporto

Il flag di riporto viene cancellato dopo l'esecuzione di una istruzione CLC. Contemporaneamente viene azzerato il relé speciale M9012.

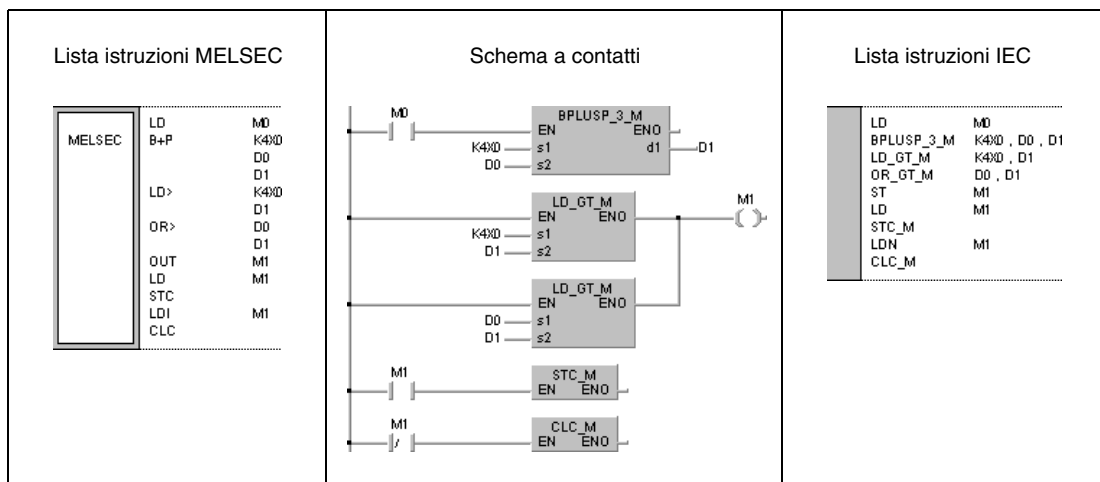
Le istruzioni STC/CLC vengono eseguite una sola volta sul fronte di salita della condizione di ingresso.



- ¹Condizione di esecuzione dell'istruzione STC
- ²Condizione di esecuzione dell'istruzione CLC
- ³Segnale di riporto (M9012)
- ⁴Una esecuzione

Esempio di programma STC, CLC

SUI fronte di salita di M0, il programma che segue somma il dato BCD da X0 a XF con il dato BCD in D0. Il risultato è memorizzato in D1. Se il risultato della somma è superiore a 9999, M1 viene attivato e viene eseguita l'istruzione STC (M9012 attivo). Se il risultato è inferiore o uguale a 9999, il segnale di riporto non è attivo.



7.18.3 DUTY

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	●

Operandi MELSEC A

	Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Riporto	Errore				
	Operandi a bit						Operandi a word (16-bit)						Costanti		Puntatore							Livello			
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V				K	H (16#)	P	I	N	M9012
n1																●	●								
n2																●	●								●
d		●																							● ²

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale di programmazione per il corrispondente numero di passi.


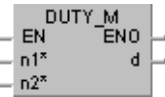
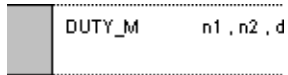
² Indicizzazione supportata solo dalle CPU A3H, A3M, AnA, AnAS e AnU.

Operandi MELSEC Q

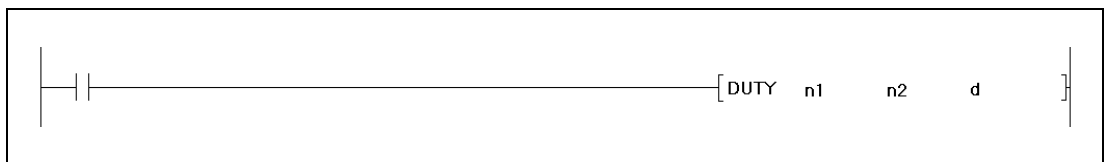
	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
n1	●	●	●	●	●	●	●	●	●	SM0	4
n2	●	●	●	●	●	●	●	●			
d	● ¹	—	—	—	—	—	—	—			

¹ Da SM420 a SM424 e da SM430 a SM434

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
--	--	---

GX Developer

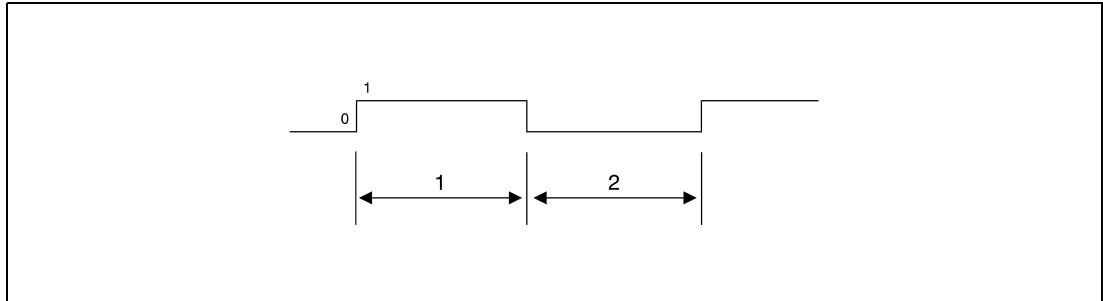


Variabili

Dati impostati	Significato	Tipo dati
n1	Numero di scansioni in cui i relé speciali sono attivi.	BIN 16-bit
n2	Numero di scansioni in cui i relé speciali sono non attivi.	
d	Indirizzo del relé speciale (serie A = M9020 - M9024, serie Q e System Q = SM420 - SM424 e SM430 - SM434).	Bit

Funzioni Preset del numero di scansioni di attivazione di un operando**DUTY Preset scansioni di attivazione**

L'istruzione DUTY attiva gli operandi indicati da d (serie A = da M9020 a M9024, serie Q e System Q = da SM420 a SM424 e da SM430 a SM434) per il numero di scansioni di programma indicato da n1 e li disattiva per il numero di scansioni specificato da n2. I relé speciali relativi servono come condizione di ingresso per le operazioni seguenti.



¹ Numero di scansioni con attivazione

² Numero di scansioni senza attivazione

I programmi che vengono eseguiti una volta per ogni ciclo, utilizzano i relé da SM420 a SM424 (serie Q e System Q).

I programmi che eseguono a bassa velocità, utilizzano i relé da SM430 a SM434 (serie Q e System Q).

All'inizio del ciclo (inizializzazione) i relé (serie A = da M9020 a M9024, serie Q/System Q = da SM420 a SM424 e da SM430 a SM434) vengono azzerati.

Se il valore $n1 = 0$, i relé rimangono disattivati.

Se il valore $n2 = 0$ ed il valore $n1$ maggiore di zero, i relé vengono eccitati e rimangono eccitati.

I valori $n1$, $n2$ e d vengono impostati all'invocazione dell'istruzione DUTY. L'impulso di scansione (relé) viene attivato o disattivato quando si incontra l'istruzione END.

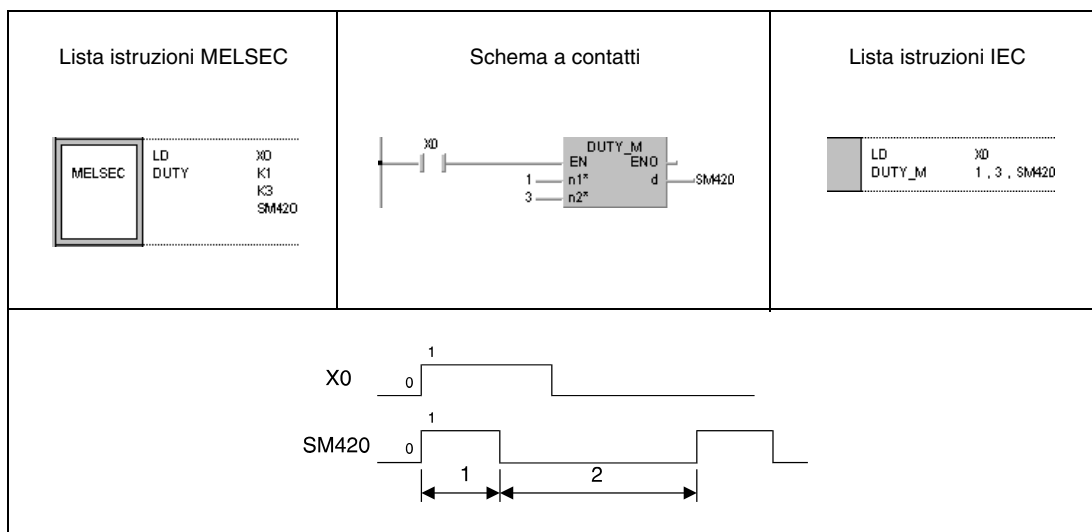
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- L'operando specificato da d non è un relé rilevante per le serie A e Q (codice di errore 4101).
- I valori $n1$ e $n2$ sono minori di zero (codice di errore 4100).

Esempio di programma DUTY (serie Q e System Q)

Con il fronte positivo di X0, il programma seguente attiva il relé speciale per una scansione di programma e lo disattiva per 3 scansioni. Queste operazioni vengono eseguite fino a quando il programma è in esecuzione (vedi NOTA seguente).

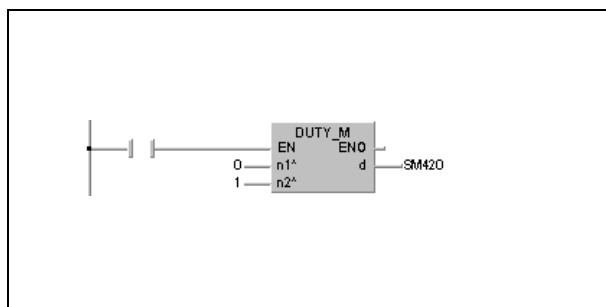


¹ Una scansione di programma con attivazione

² Tre scansioni senza attivazione

NOTA

Se la condizione di attivazione viene azzerata (X0 = OFF) l'uscita dell'impulso dell'istruzione DUTY con la relativa attivazione/disattivazione del relé specificato viene mantenuta. Per arrestare l'uscita degli impulsi, è necessario inserire il seguente segmento di programma.



7.18.4 ZRRDB, ZRRDBP



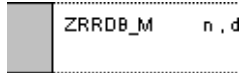
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

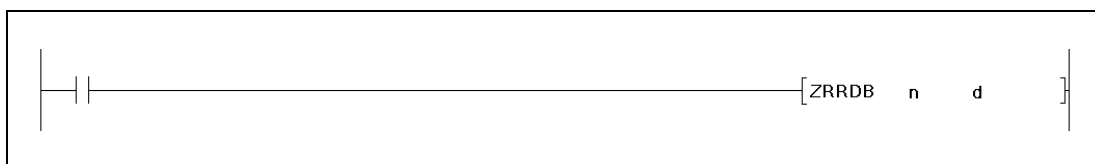
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
n	●	●	●	●	●	●	●	●	—	SM0	3
d	●	●	●	●	●	●	●	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
--	--	---

GX Developer



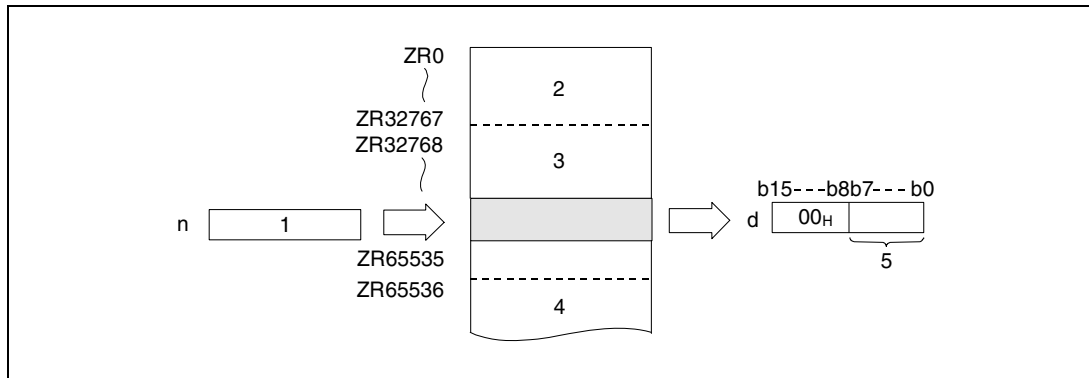
Variabili

Dati impostati	Significato	Tipo dati
n	Lettura diretta di un byte da un file registri.	BIN 32-bit
d	Numero dell'operando che contiene il byte letto.	BIN 16-bit

Funzioni **Letture diretta di un byte da un file registri**

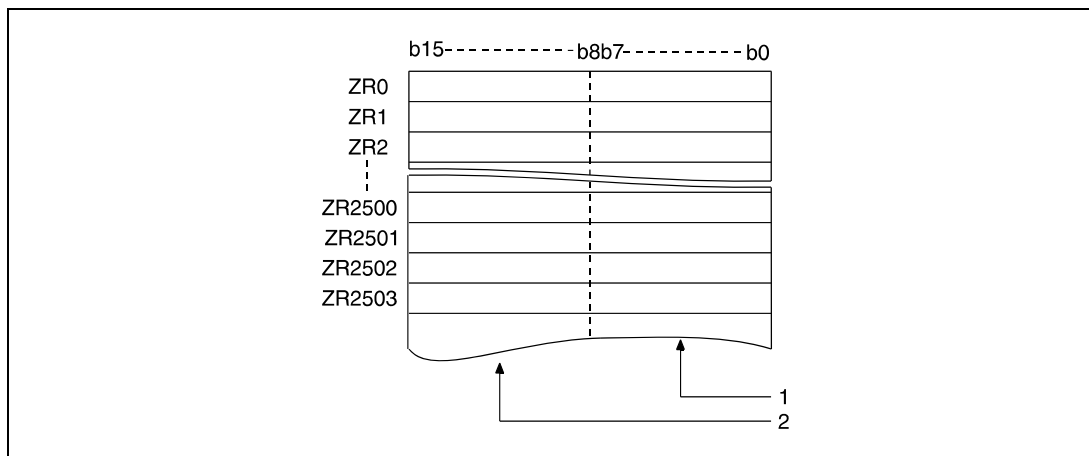
ZRRDB **Letture di un byte**

L'istruzione ZRRDB legge il byte specificato da n da un file registri. Il numero di byte non indica un indirizzo di blocco. Il byte è contenuto nella parte leggera dell'operando indicato da d. Il byte pesante dell'operando specificato da d contiene il valore "00H".



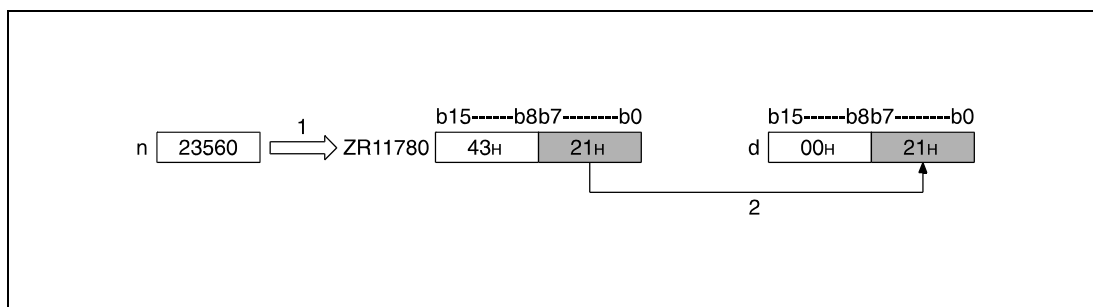
- 1 Indirizzo seriale del byte
- 2 Area file registri per blocco 0
- 3 Area file registri per blocco 1
- 4 Area file registri per blocco 2
- 5 Byte letto

L'assegnazione degli indirizzi dei file registri ai corrispondenti indirizzi seriali del byte è mostrata qui sotto:



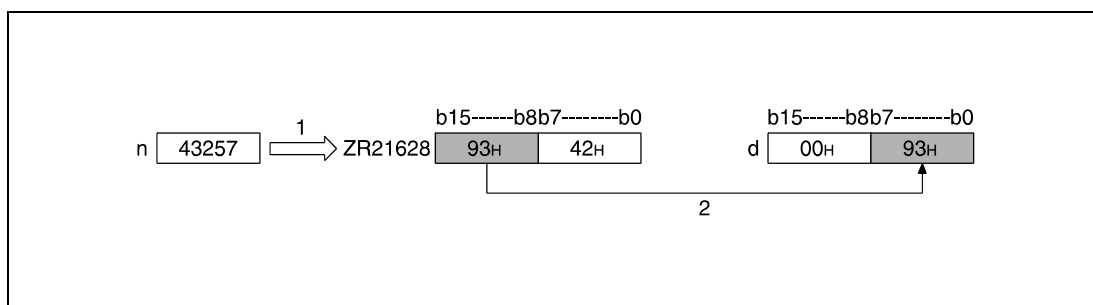
- 1 Area di memoria per indirizzi byte pari (qui: indirizzi da 0 a 5006)
- 2 Area di memoria per indirizzi byte dispari (qui: indirizzi da 1 a 5007)

Se viene specificato il byte 23560, viene letto il byte leggero del file registri ZR11780.



- ¹ Indirizzo
- ² Memoria

Se viene specificato il byte 43257, viene letto il byte leggero del file registri ZR21628.



- ¹ Indirizzo
- ² Memoria

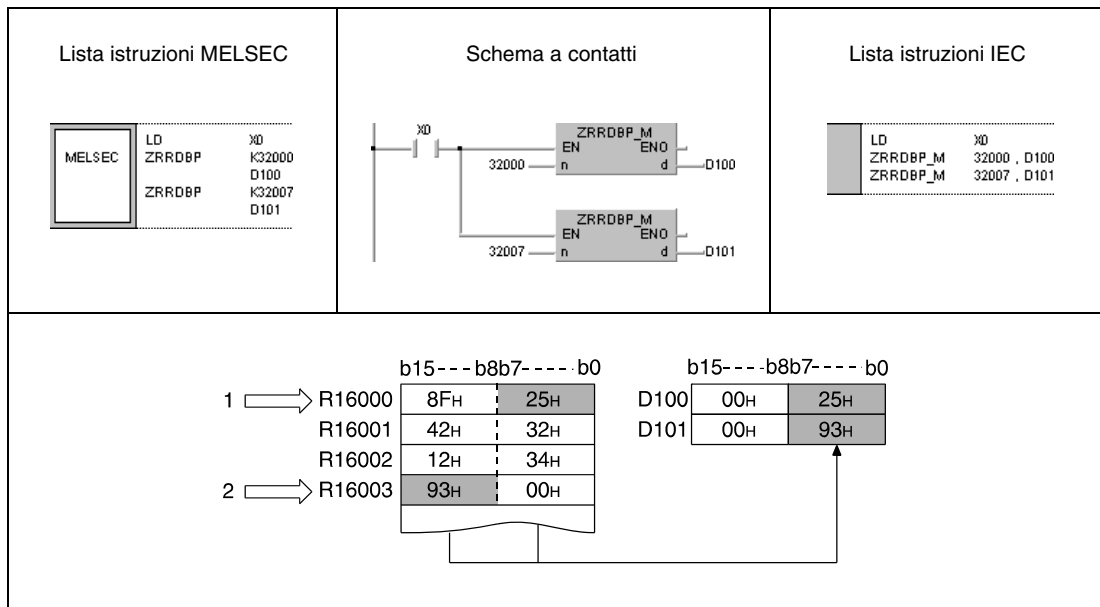
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il numero dell'operando (indirizzo seriale del byte) supera il campo operando ammesso (codice di errore 4101).

Esempio di programma ZRRDBP

Con il fronte positivo di X0, il programma seguente legge il byte leggero del file registri R16000 (byte numero 32000) ed il byte pesante del registro R16003 (byte numero 32007). I byte vengono memorizzati in D100 e D101.



¹ Indirizzo byte seriale 32000 (byte leggero del registro R16000)

² Indirizzo byte seriale 32007 (byte pesante del registro R16003)

7.18.5 ZRWRB, ZRWRBP

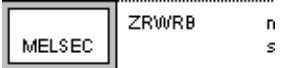
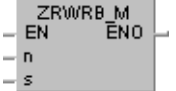
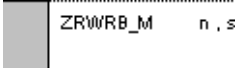
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

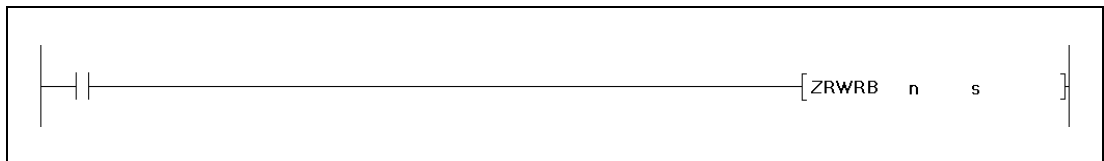
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
n	●	●	●	●	●	●	●	●	—	SM0	3
s	●	●	●	●	●	●	●	●	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
--	--	---

GX Developer



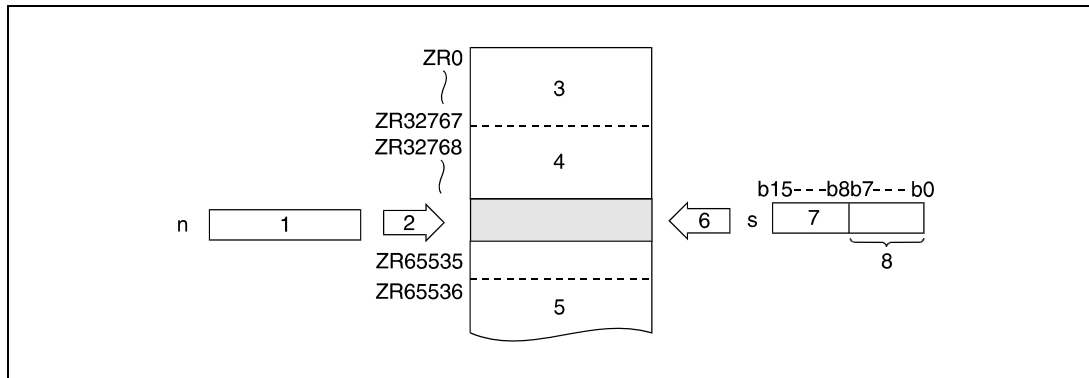
Variabili

Dati impostati	Significato	Tipo dati
n	Scrittura diretta di un byte in un file registri.	BIN 32-bit
s	Operando che contiene i dati da scrivere.	BIN 16-bit

Funzioni Scrittura diretta di un byte in un file registri

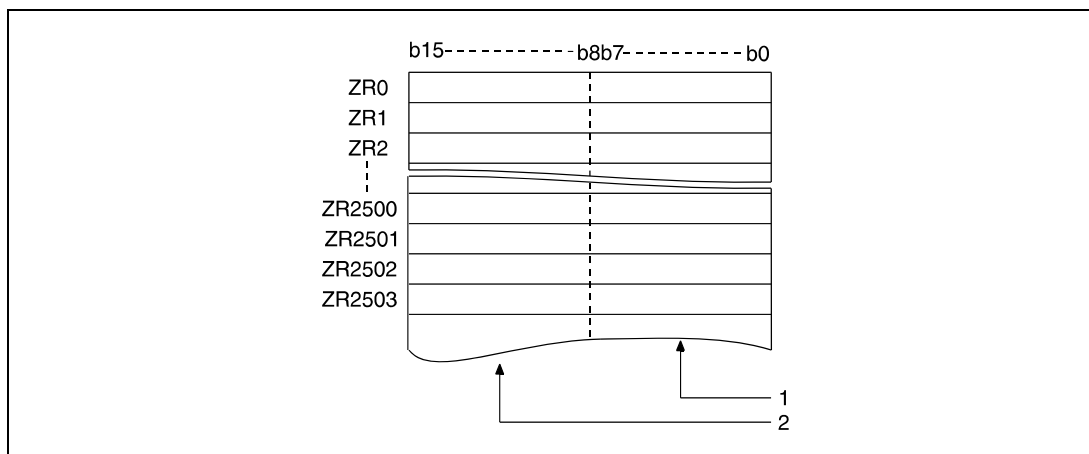
ZRWRB Scrittura di un byte

L'istruzione ZRRDB scrive il contenuto del byte leggero dell'operando specificato da s nel file registri specificato da n come indirizzo seriale. Il numero di byte in s non contiene un indirizzo di blocco. Il byte pesante dell'operando in s viene ignorato.



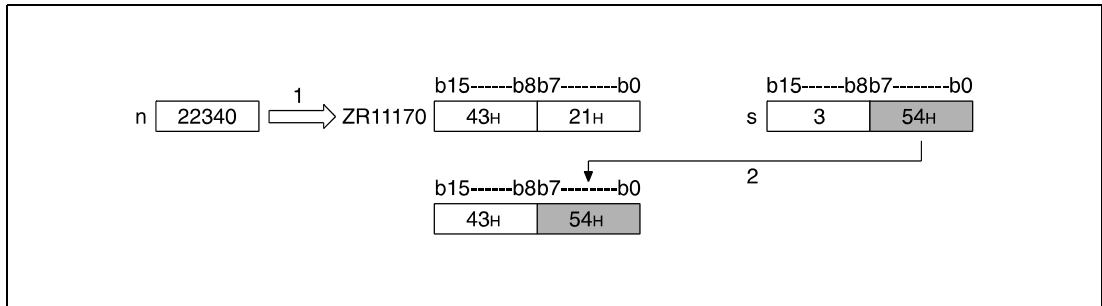
- 1 Indirizzo seriale del byte
- 2 Indirizzo
- 3 Area file registri per blocco 0
- 4 Area file registri per blocco 1
- 5 Area file registri per blocco 2
- 6 Scrive dati
- 7 Questo byte viene ignorato
- 8 Byte da scrivere

L'assegnazione degli indirizzi dei file registri ai corrispondenti indirizzi seriali del byte è mostrata qui sotto:



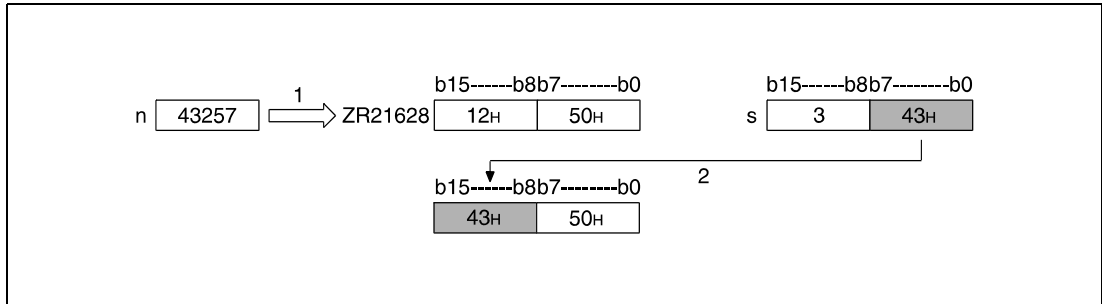
- 1 Area di memoria per indirizzi byte pari (qui: indirizzi da 0 a 5006)
- 2 Area di memoria per indirizzi byte dispari (qui: indirizzi da 1 a 5007)

Se viene specificato il numero 22340, il byte leggero dell'operando indicato da s viene scritto nel byte leggero del file registri ZR11170.



- ¹ Indirizzo
- ² Scrive byte
- ³ Questo byte viene ignorato

Se viene specificato il numero 43257, il byte leggero dell'operando indicato da s viene scritto nel byte leggero del file registri ZR21628.



- ¹ Indirizzo
- ² Scrive byte
- ³ Questo byte viene ignorato

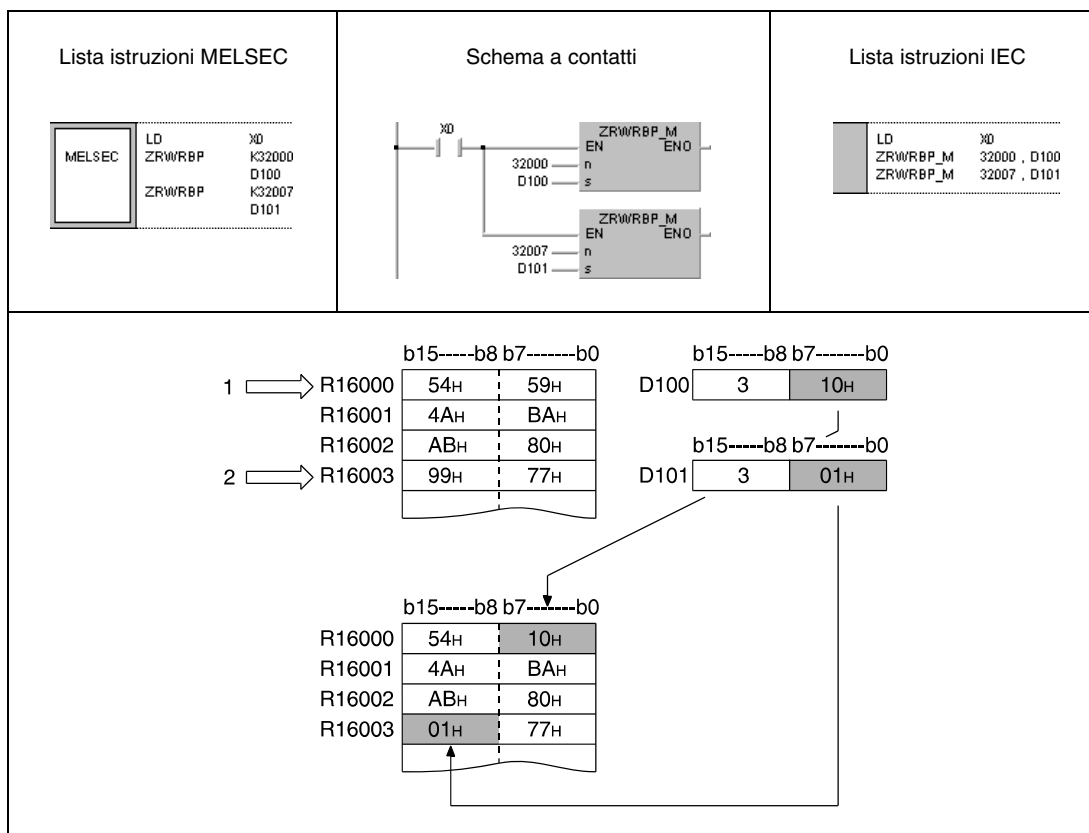
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il numero di operando (indirizzo byte seriale) specificato da n supera il campo operando ammesso (codice di errore 4101).

Esempio di programma ZRWRBP

Con il fronte positivo di X0 il programma seguente scrive il contenuto dei byte leggeri dei registri da D100 e D101 nel byte leggero del file registri R16000 (byte numero 32000) e nel byte pesante del file registri R16003 (numero byte 32007).



- ¹ Indirizzo byte seriale 32000 (byte leggero del registro R16000)
- ² Indirizzo byte seriale 32007 (byte pesante del registro R16003)
- ³ Questi byte sono ignorati.

7.18.6 ADRESET, ADRSETP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

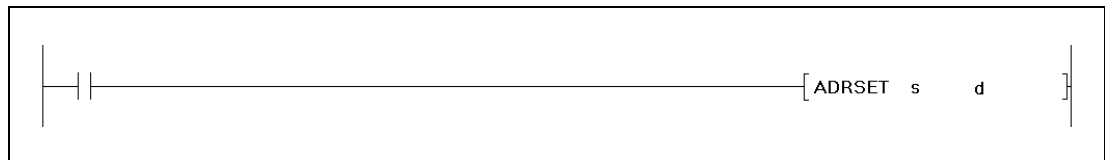
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	●	●	●	—	—	—	—	—	SM0	3	
d	●	●	●	—	—	—	—	—			

NOTA

Le istruzioni ADRSET e ADRSETP non sono supportate da GX IEC Developer.

GX Developer



Variabili

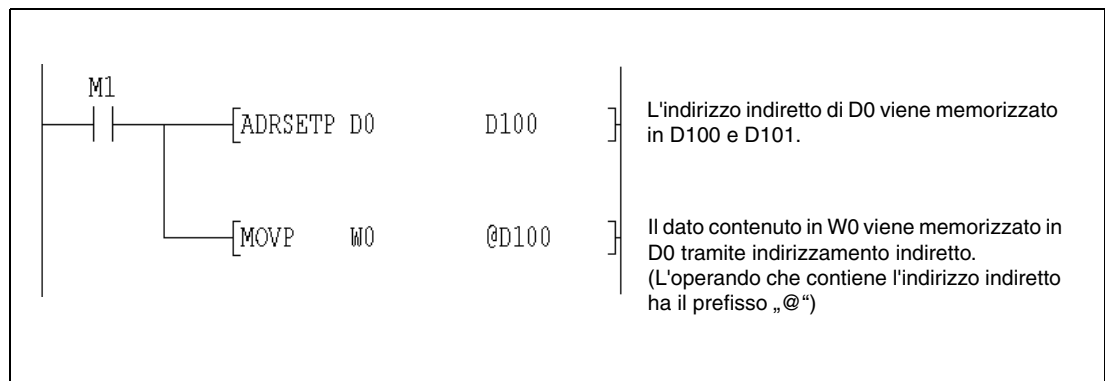
Dati impostati	Significato	Tipo dati
s	Numero operando per lettura indiretta	Nome operando
d	Indirizzo operando che memorizza l'indirizzo indiretto dell'operando definito da s.	BIN 32-bit

Funzioni

Operazioni di lettura con indirizzi indiretti

ADRSET Memorizza indirizzo indiretto

Memorizza l'indirizzo indiretto dell'operando designato da s su d e d+1. L'indirizzo memorizzato nell'operando indicato da d viene usato in lettura quando il programma di sequenza effettua l'indirizzamento indiretto di un operando. L'operando s non può indicare un operando a bit.



Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- È stato indicato un operando la cui designazione non è ammessa (codice di errore 4101).

7.18.7 KEY

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
		● ¹	● ¹	●	● ²

¹ Utilizzando una CPU AnA o AnU, questa istruzione dedicata dell'editor IEC può essere programmata come funzione, mentre nell'editor MELSEC può essere programmata in combinazione con le istruzioni LEDA, LEDC e LEDR.

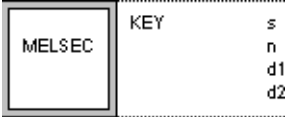
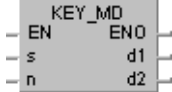
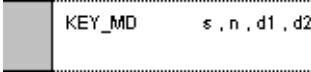
² Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

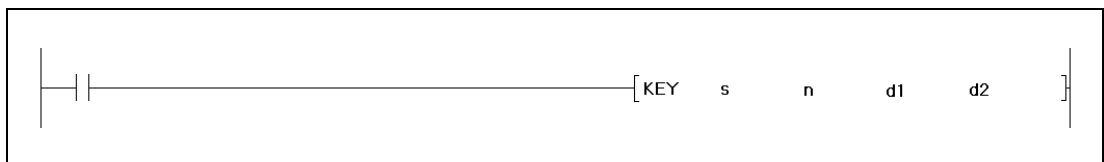
	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s	● ¹	—	—	—	—	—	—	—	—	SM0	5
n	●	●	●	●	●	●	●	●	—		
d1	—	●	●	—	—	—	—	—	—		
d2	●	●	●	●	●	●	—	—	—		

¹ Solo X

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
--	--	---

GX Developer



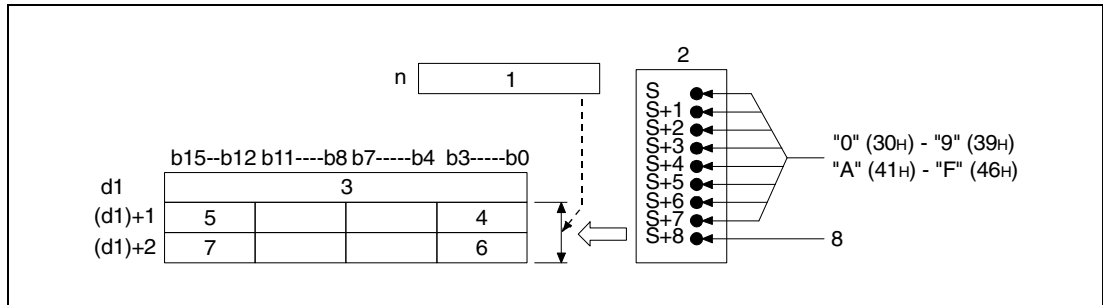
Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
s	Indirizzo del primo operando (X), che riceve l'input numerico da tastiera.	Bit	Array [1..9] di BOOL
n	Numero di cifre da digitare.	BIN 16-bit	ANY16
d1	Indirizzo del primo operando che memorizza l'input numerico da tastiera.	BIN 16-bit	Array [1.0.3] di ANY16
d2	Indirizzo dell'operando a bit da attivare al completamento dell'input da tastiera.	Bit	BOOL

Funzioni Inserimento tasti numerici

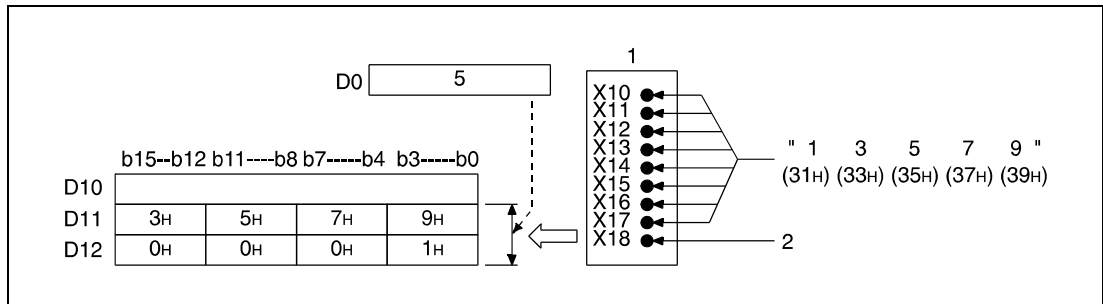
KEY Istruzione di ingresso

L'istruzione KEY supporta l'inserimento da tastiera dei caratteri ASCII da 0 (30H) a 9 (39H) e da A (41H) a F (46H) negli ingressi specificati da s+0 (array_s[0]) a s+7 (array_s[7]). I valori inseriti negli ingressi sono codificati in formato esadecimale e memorizzati negli operandi indicati da (d1)+0 (array_d1[0]) a (d1)+2 (array_d1[2]). Il numero di caratteri da inserire è indicato in n.



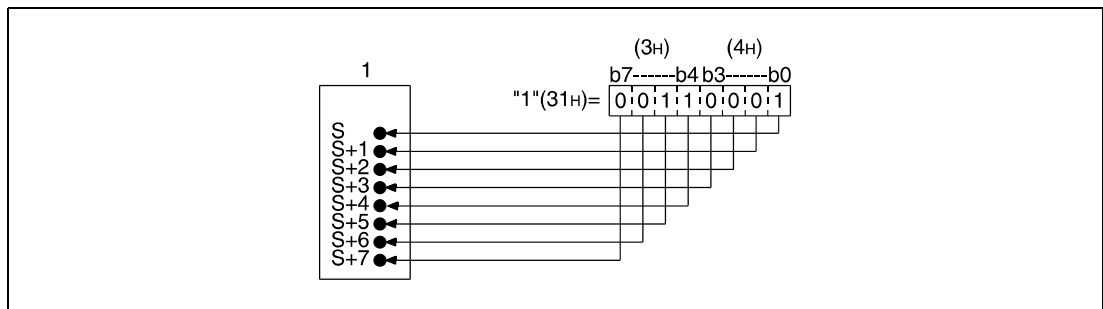
- ¹ Numero di valori da inserire
- ² Modulo ingressi
- ³ Numero valori inseriti
- ⁴ Ottavo carattere inserito
- ⁵ Quinto carattere inserito
- ⁶ Quarto carattere inserito
- ⁷ Primo carattere inserito
- ⁸ Segnale di strobe

Nella figura seguente n specifica il valore 5, mentre i valori da 1 (31H) a 5 (35H) vengono inseriti sugli ingressi da X10 a X18 del modulo di ingresso.



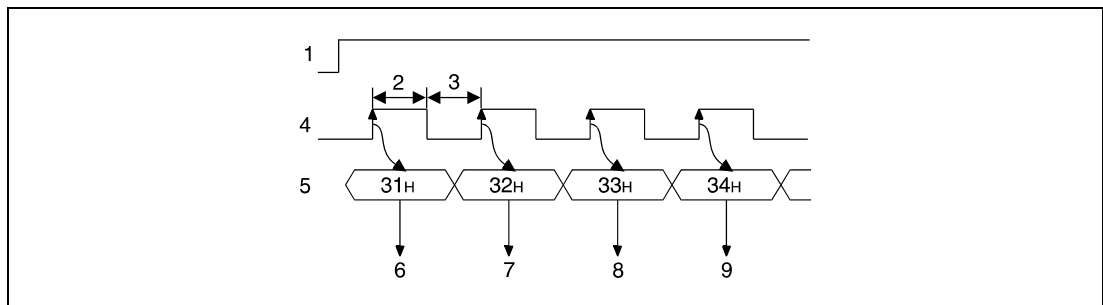
- ¹ Modulo ingressi
- ² Segnale di strobe

I caratteri ASCII inviati agli ingressi (X) specificati da s+0 (array_s[0]) a s+7 (array_s[7]) sono codificati in formato binario a 8 bit, come mostrato di seguito:



¹ Modulo ingressi

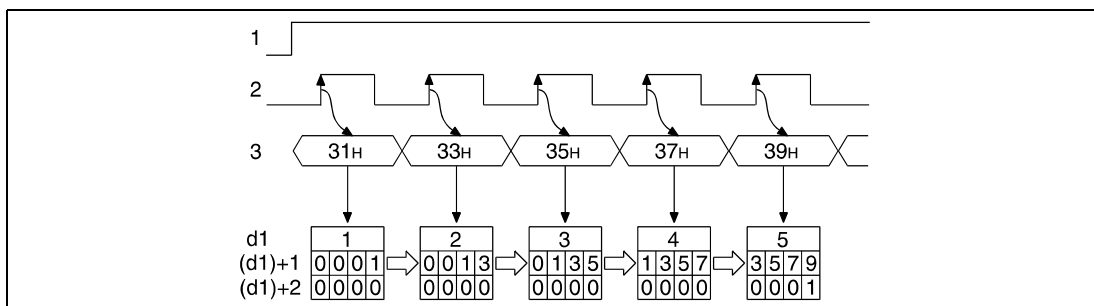
Dopo l'inserimento di un carattere ASCII da s+0 (array_s[0]) a s+7 (array_s[7]), viene attivato il segnale di strobe (s+8, array_s[8]) per collegare internamente l'ingresso dei dati. Il periodo in cui il segnale di strobe rimane attivo o non attivo deve essere superiore al tempo di scansione per assicurare un collegamento corretto dei dati inseriti.



- ¹ Condizione di esecuzione dell'istruzione KEY
- ² Attiva per oltre una scansione di programma
- ³ Non attiva per oltre una scansione di programma
- ⁴ Segnale di strobe (s+8, array_s[8])
- ⁵ Dati ASCII in ingresso (da s+0 a s+7, array_s[0] a array_s[7])
- ⁶ Lettura "1"
- ⁷ Lettura "2"
- ⁸ Lettura "3"
- ⁹ Lettura "4"

L'istruzione KEY può essere eseguita solo con la condizione di esecuzione attiva. La condizione di esecuzione deve rimanere attiva fino al completamento dell'inserimento degli n caratteri specificati.

Il numero di caratteri inseriti è contenuto in (d1)+0 (array_d[0]). I caratteri ASCII inseriti sono in effetti memorizzati negli operandi (d1)+1 (array_d[1]), (d1)+2 (array_d[2]) e (d1)+2 (array_d[2]) come valori binari esadecimali; cioè vengono inseriti 4 bit per ogni carattere inserito. I valori binari esadecimali dei caratteri da 0H a FH è nel campo da "0000" a "1111".

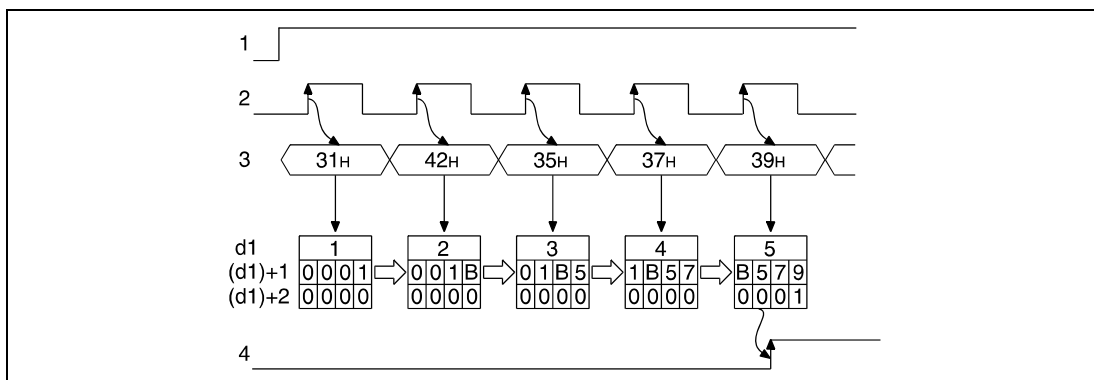


- ¹ Condizione di esecuzione dell'istruzione KEY
- ² Segnale di strobe (s+8, array_s[8])
- ³ Dati ASCII in ingresso (da s+0 a s+7, array_s[0] a array_s[7])

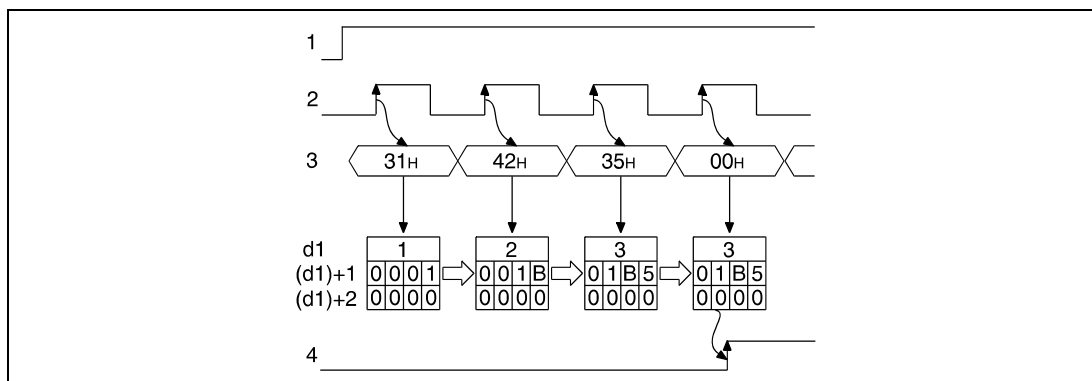
Il numero di caratteri da inserire specificato da n deve essere nel campo da 1 a 8.

Se viene inserito il numero di caratteri specificato, oppure il codice "00H", il collegamento dei dati di ingresso è completato e l'operando indicato da d2 viene attivato. Le figure seguenti mostrano queste operazioni. Per n viene specificato il valore 5.

Nella figura seguente, l'input viene completato dopo 5 caratteri. Nella figura successiva, l'input viene completato dopo il codice "00H".



- ¹ Condizione di esecuzione dell'istruzione KEY
- ² Segnale di strobe (s+8, array_s[8])
- ³ Dati ASCII in ingresso (da s+0 a s+7, array_s[0] a array_s[7])
- ⁴ Input caratteri terminato (operando specificato da d2 attivo)



- ¹ Condizione di esecuzione dell'istruzione KEY
- ² Segnale di strobe (s+8, array_s[8])
- ³ Dati ASCII in ingresso (da s+0 a s+7, array_s[0] a array_s[7])
- ⁴ Input caratteri terminato (operando specificato da d2 attivo)

Prima di poter inserire nuovi caratteri, il contenuto degli operandi da (d1)+0 (array_d1[0]) a (d1)+2 (array_d[2]) deve essere azzerato e l'operando specificato da d2 deve essere disattivato; altrimenti non è possibile acquisire nuovi caratteri.

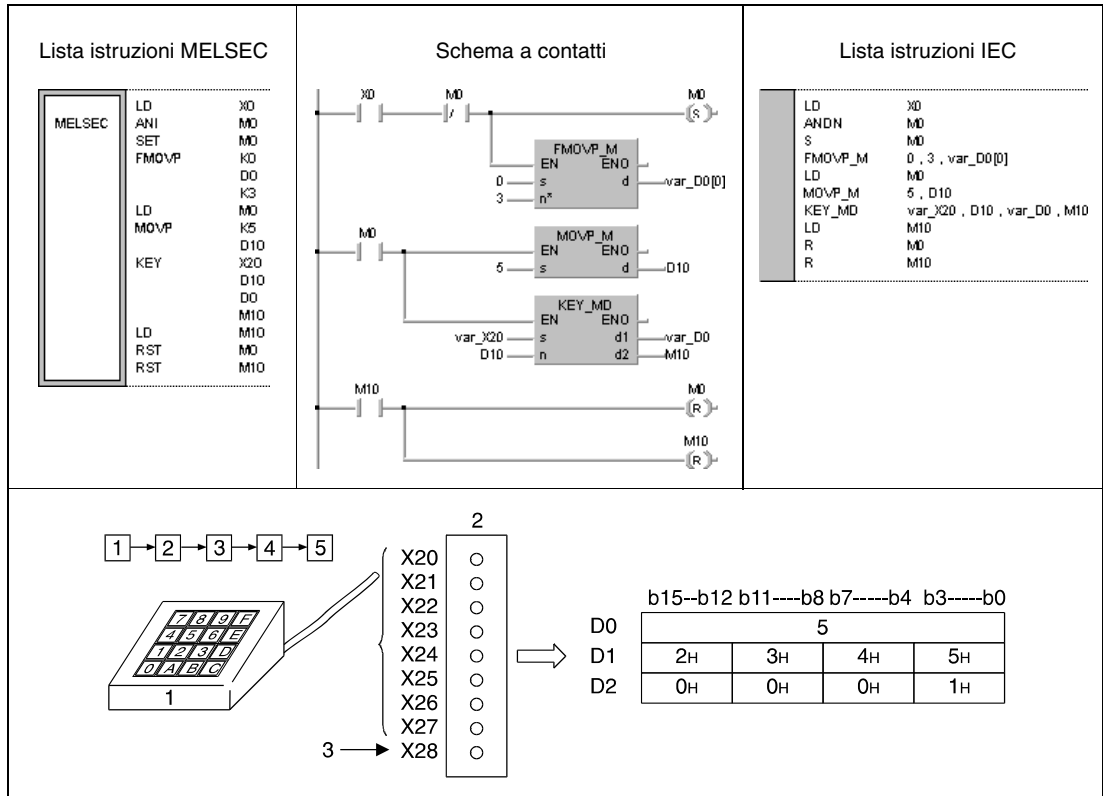
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- L'operando indicato da s non è un ingresso (X) (codice di errore 4100).
- Il numero di caratteri specificato da n non è compreso da 1 a 8.

Esempio di programma

Il programma seguente abilita l'input da tastiera di fino a 5 valori numerici tramite gli ingressi da X20 (var_X20[0]) a X27 (var_X20[7]). I valori vengono memorizzati nei registri D1 (var_D0[1]) e D2 (var_D0[2]) codificati in binario in formato esadecimale. Il numero di valori già inseriti è contenuto in D0 (var_D0[0]). Prima dell'esecuzione dell'istruzione KEY, i registri da D0 (var_D0[0]) a D2 (var_D0[2]) vengono cancellati e viene inserito il numero di valori da inserire (5). Al termine dell'esecuzione dell'istruzione KEY il relé M10 (input completato) viene disattivato. Il segnale di strobe viene inviato sull'ingresso X28 (var_X20[8]).



- ¹ Tastierino numerico
- ² Modulo ingressi
- ³ Segnale di strobe

7.18.8 ZPUSH, ZPUSHP, ZPOP, ZPOPP

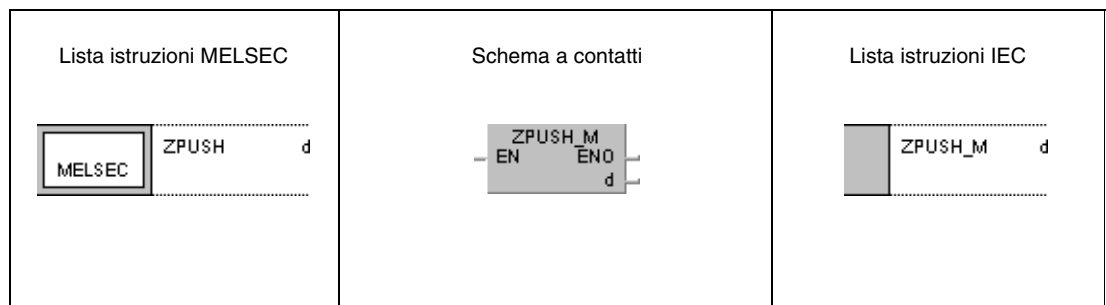
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

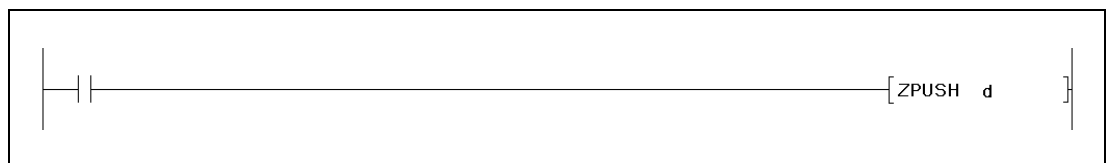
Operandi MELSEC Q

d	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
	—	●	●	—	—	—	—	—	SM0	3	

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
d	Indirizzo del primo dispositivo che memorizza il contenuto di registri indice.	BIN 16-bit

Funzioni Salvataggio e ripristino del contenuto di un gruppo di registri indice

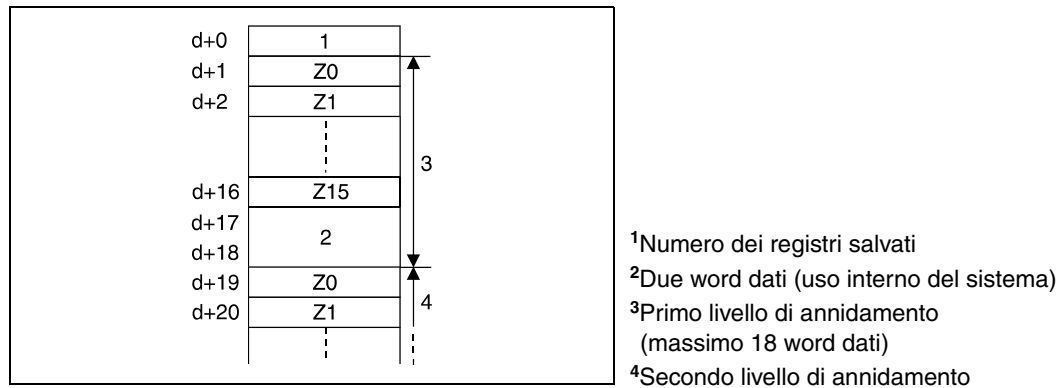
ZPUSH Salvataggio di un gruppo di registri indice

L'istruzione ZPUSH salva il contenuto dei registri indice da Z0 a Z15 a partire dall'operando indicato da d.

Questi dati possono essere recuperati con l'istruzione ZPOP. L'istruzione può essere usata nei diversi livelli di annidamento compresi nel loop ZPUSH / ZPOP.

Per l'esecuzione in livelli diversi di annidamento, considerare che l'istruzione ZPUSH richiede 18 registri a 16 bit nell'area degli operandi specificata a partire da d. Per l'esecuzione dell'istruzione ZPUSH è quindi necessaria la disponibilità della quantità adeguata di area di memoria.

La figura seguente mostra l'organizzazione dell'area di memoria a partire da d:



ZPOP Ripristino di un gruppo di registri indice

L'istruzione ZPOP recupera il contenuto dei registri indice salvati con l'istruzione ZPUSH. Il contenuto dell'area di memoria specificata a partire da d viene letto e riscritto nei registri indice corrispondenti.

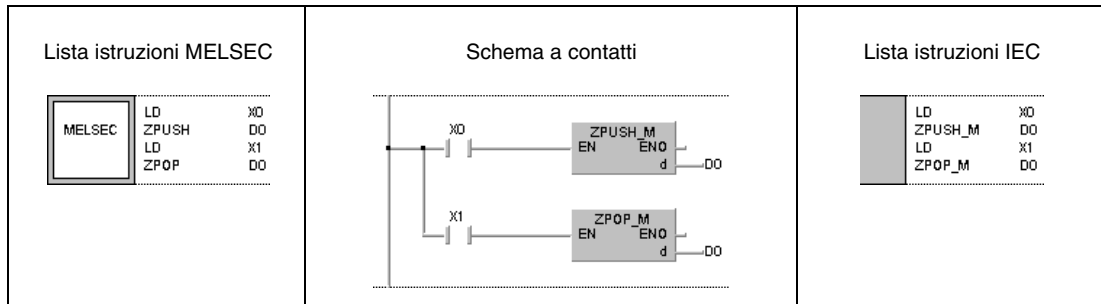
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- L'area di memoria specificata a partire da d supera il campo operandi ammesso (codice di errore 4101).
- Il contenuto dell'operando contenuto in d+0 è 0 (numero dei registri salvati) (codice di errore 4100).

Esempio di programma ZPUSH/ZPOP

Se X20 è attivo, il programma seguente salva il contenuto dei registri indice nell'area di memoria a partire da D0. Viene poi richiamato il sottoprogramma corrispondente a label_0, che usa i registri indice.



7.18.9 EROMWR, EROMWRP

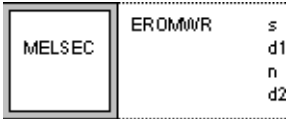
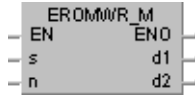
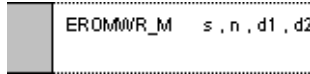
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	

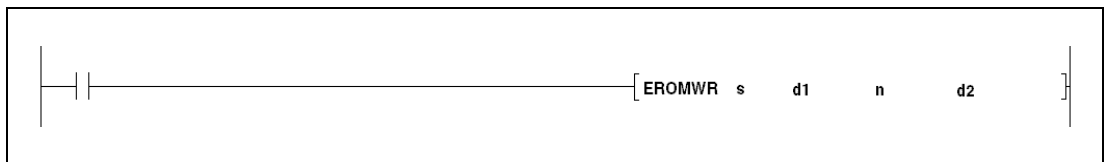
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□□□	Registro indice Zn	Costanti K, H (16#)			Altro U
	Bit	Word		Bit	Word						
s	—	●	—	—	—	—	—	—	SM0	6	
d1	—	—	●	—	—	—	—	—			
n	●	●	●	●	●	●	●	—			
d2	●	—	—	—	—	—	—	—			

GX IEC Developer

Lista istruzioni MELSEC 	Schema a contatti 	Lista istruzioni IEC 
---	--	---

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che contiene i dati da scrivere.	BIN 16-bit
d1	Primo indirizzo del file registri EEPROM in cui scrivere.	
n	Numero di word dati da scrivere.	
d2	Operando da attivare al termine dell'operazione.	Bit

Funzioni Scrittura di un gruppo di dati in un file registri EEPROM**EROMWR Istruzione di scrittura**

L'istruzione EROMWR scrive il numero di word dati specificato da n, contenute nell'operando indicato da s, nel file registri EEPROM specificato da d1.

Al completamento dell'operazione di scrittura, viene attivato l'operando indicato da d2, per essere poi disattivato automaticamente alla successiva scansione di programma.

l'istruzione EROMWR viene eseguita fino all'elaborazione dell'istruzione END. In questo tempo si possono scrivere fino a 64 word dati in ciascuna scansione di programma. Il numero di scansioni necessarie risulta dal quoziente arrotondato per eccesso del numero di word specificato da n, diviso per 64. Il tempo di elaborazione può essere calcolato sulla base di un tempo di scansione di circa 10 ms.

Il dato specificato da s non deve variare durante l'operazione di scrittura, altrimenti i dati possono essere perduti.

**Errori di
esecuzione**

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- L'area di memoria necessaria per il numero di word dati specificati da n supera il campo indirizzi ammesso per gli operandi s e d1 (codice di errore 4101).
- Il file registri specificato da d1 non esiste o non è un file registri EEPROM. (codice di errore 4101).

8 Istruzioni per comunicazione dati

8.1 Fondamenti

Una CPU QnA(S) può essere usata in un sistema di rete MELSECNET(II)/B/10. Una CPU del System Q supporta i sistemi di rete MELSECNET/10 e MELSECNET/H.

NOTA

I termini MELSECNET/10 e MELSECNET/H usati in questo contesto si riferiscono ai sistemi di rete MELSECNET/10 e MELSECNET/H.

Il termine MELSECNET usato in questo contesto si riferisce ai sistemi di rete MELSECNET(I), MELSECNET(II), e MELSECNET/B.

Tramite le istruzioni di comunicazione, la CPU può scambiare dati con altre stazioni collegate a MELSECNET e MELSECNET/10.

8.2 Categorie di istruzioni

Le istruzioni di comunicazione sono suddivise nelle seguenti quattro categorie:

1. Istruzioni per rinfresco dati

Queste istruzioni rinfrescano i dati nei moduli di rete designati.

2. Istruzioni dedicate per comunicazione dati

Queste istruzioni vengono utilizzate con CPU QnA o del System Q. Per la comunicazione dati vengono usati più canali del modulo di rete.

3. Istruzioni per comunicazione dati compatibili con serie A

Queste istruzioni sono identiche alle istruzioni dedicate ACPU.

4. Lettura/scrittura dati di instradamento

Queste istruzioni leggono e scrivono parametri da e per stazioni di ripetizione e instradamento.

Sui sistemi MELSECNET e MELSECNET/10 si possono utilizzare solo istruzioni specifiche per i collegamenti dati. Le istruzioni utilizzabili su una rete MELSECNET/10 dipendono ulteriormente dal tipo di unità utilizzata (CPU System Q, CPU serie A, CPU QnA o stazione I/O remoto).

La tabella che segue offre una panoramica delle istruzioni di comunicazione dati:

Categoria	Significato
Istruzioni per rinfresco rete	Istruzioni per operazioni di rinfresco dati sui moduli di rete.
Istruzioni dedicate per collegamenti dati	Letture e scritture dati da e per stazioni target in reti target. Invio dati su moduli di rete di stazioni target in reti target. Lettura di dati CPU inviato con istruzione SEND. Richieste dati da stazioni diverse (operazioni di lettura/scrittura con dati orologio, operazioni RUN/STOP). Lettura e scrittura dati da e per moduli funzione speciali in stazioni I/O remote.
Istruzioni per comunicazione dati compatibili con serie A	Letture e scritture dati da e per stazioni target in reti diverse. Lettura e scrittura dati da e per stazioni locali (solo su stazione master). Lettura e scrittura dati da e per moduli funzione speciali in stazioni I/O remote.
Letture/scritture dati di instradamento	Letture e scritture di parametri di routing (numero di rete e numero stazione della stazione di smistamento, numero di stazione della stazione di routing)

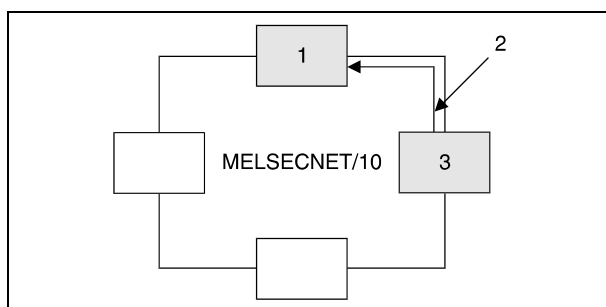
8.3 Campi di lettura e scrittura dei dati

8.3.1 MELSECNET/10

Con MELSECNET/10 una stazione host esegue operazioni di lettura/scrittura con stazioni della stessa rete o, tramite indirizzamento (parametri di routing) con stazioni di reti diverse.

Operazioni di lettura/scrittura con stazioni della stessa rete.

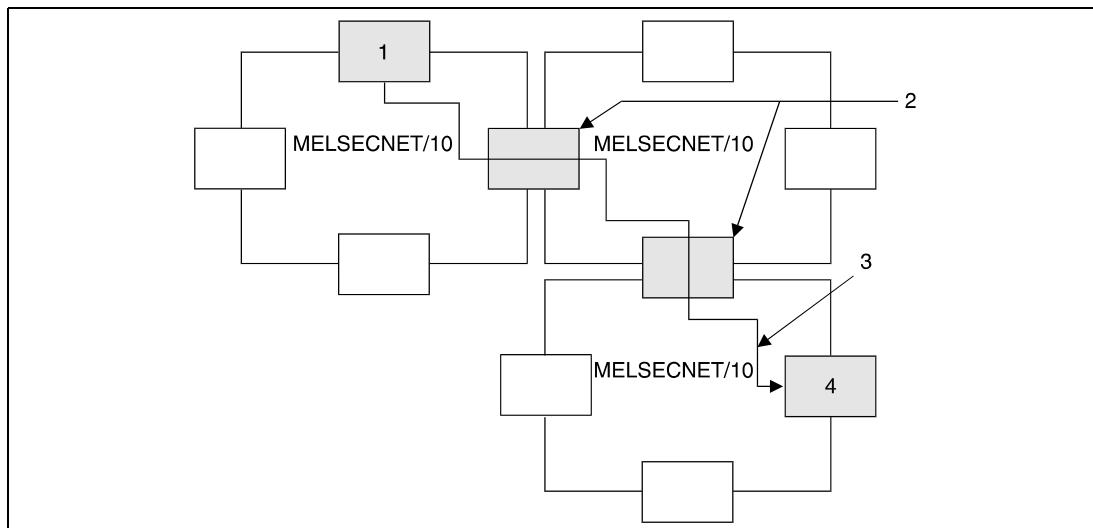
Il numero di rete della stazione target e quello del modulo di rete a cui la stazione host è collegata, devono essere uguali. Questa funzione legge e scrive dati da e per qualsiasi stazione all'interno della rete.



- ¹Stazione che esegue l'istruzione
- ²Operazione di lettura
- ³Stazione target

Operazioni di lettura/scrittura con stazioni di reti diverse

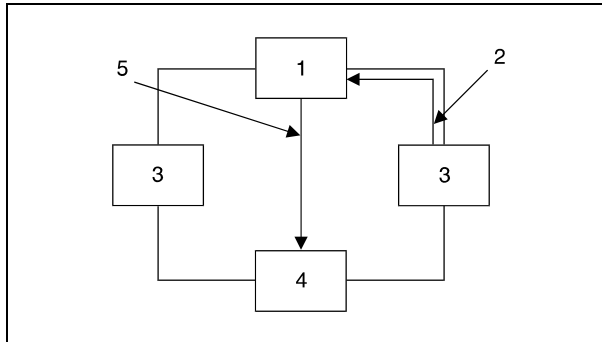
Il numero di rete della stazione target e quello del modulo di rete a cui la stazione host è collegata, devono essere diversi. Una stazione nella stessa rete della stazione host serve come stazione ripetitrice che instrada le operazioni di lettura/scrittura per la stazione target in una rete diversa.



- ¹ Stazione che esegue l'istruzione
- ² Stazione ripetitrice (necessari parametri di instradamento)
- ³ Operazione di lettura
- ⁴ Stazione target

8.3.2 MELSECNET

Con MELSECNET(I/II/B) una stazione master esegue operazioni di lettura/scrittura con stazioni locali e stazioni di I/O remote.



- 1Stazione master
- 2Operazione di lettura/scrittura
- 3Stazione locale
- 4Stazione I/O remoti
- 5Operazione di lettura/scrittura con moduli funzione speciali

8.4 Istruzioni dedicate per collegamenti dati

Vengono riportate di seguito alcune considerazioni per l'uso delle istruzioni dedicate per comunicazione dati per le CPU serie Q e System Q.

8.4.1 Esecuzione simultanea

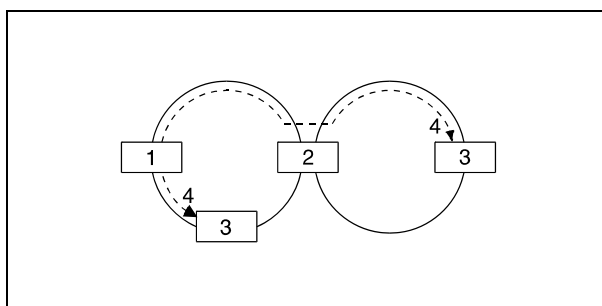
I moduli di rete dei sistemi MELSECNET/10 possiedono 8 zone per la comunicazione, usate dalle istruzioni di comunicazione. Questi moduli di rete non supportano l'esecuzione simultanea di più istruzioni di comunicazione all'interno della stessa zona. Se deve essere eseguita più di una istruzioni di comunicazione all'interno della stessa zona, è necessario serializzare l'esecuzione delle singole istruzioni ricorrendo agli operandi di completamento attivati al termine di ciascuna istruzione di lettura/scrittura.

8.4.2 Completamento della trasmissione

Nella programmazione delle istruzioni di comunicazione dedicate per la serie Q e System Q, è possibile specificare se il completamento della trasmissione deve essere confermato o meno.

Conferma del completamento trasmissione

La figura seguente mostra il modo in cui il completamento dell'esecuzione o della trasmissione dati viene confermato, quando i dati sono stati scritti sul canale indicato della stazione target designata (nelle operazioni di lettura è possibile utilizzare solo questa modalità).



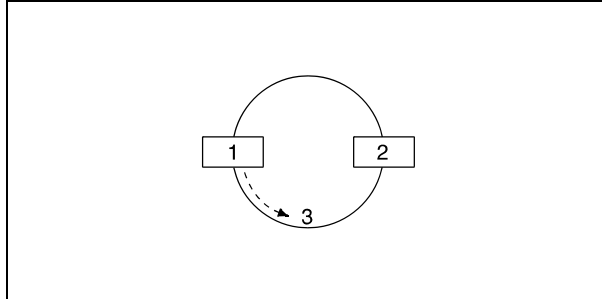
- 1Sorgente dell'esecuzione
- 2Stazione ripetitrice
- 3Stazione target
- 4Esecuzione/trasferimento completato

Nessuna conferma del completamento trasmissione

Le figure seguenti mostrano il modo in cui il completamento dell'esecuzione o della trasmissione dati può non essere confermato.

All'interno della stessa rete:

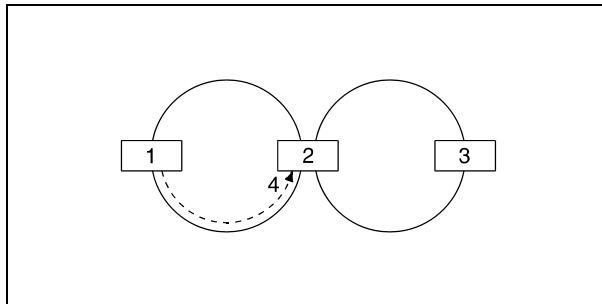
L'esecuzione o il trasferimento dati è completato quando la stazione host ha inviato tutti i dati.



- ¹Sorgente dell'esecuzione
- ²Stazione target
- ³Esecuzione/trasferimento completato

Fra reti diverse:

L'esecuzione o la trasmissione dati è completata quando i dati inviati sono arrivati ad una stazione ripetitrice sulla rete della stazione host.



- ¹Sorgente dell'esecuzione
- ²Stazione ripetitrice
- ³Stazione target
- ⁴Esecuzione/trasferimento completato

NOTA

Per migliorare l'integrità dei dati, si consiglia di scegliere la modalità con conferma del completamento trasmissione.

Se viene specificata la modalità senza conferma del completamento trasmissione, la trasmissione viene considerata conclusa al termine dell'invio, indipendentemente dagli eventuali errori occorsi durante la trasmissione. Inoltre, la stazione target restituisce un errore "buffer ricezione pieno" nel caso in cui diverse stazioni eseguano contemporaneamente istruzioni di comunicazione, anche se i dati vengono trasmessi correttamente. Ciò nonostante, la stazione trasmittente completa l'operazione anche in questo caso.

8.5 Istruzioni per rinfresco dati

Le istruzioni che seguono rinfrescano i dati sui moduli di rete. La tabella che segue offre una panoramica di queste istruzioni:

Funzione	MELSEC Istruzione in editor MELSEC	MELSEC Istruzione con Editor IEC	Stazione designata in MELSECNET/10			MELSECNET
			CPU QnA System Q	ACPU	Stazione I/O remoto	
Istruzioni per rinfresco dati	ZCOM	ZCOM_J_M	—	—	—	—
		ZCOM_JP_M				
		ZCOM_U_M				
		ZCOM_UP_M				

8.5.1 ZCOM

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

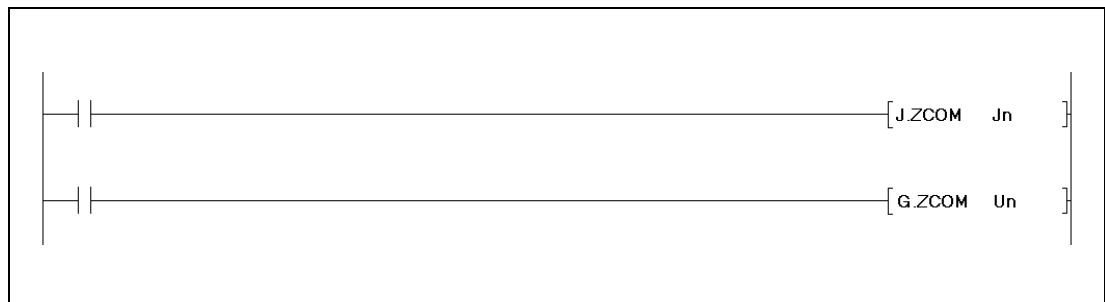
Operandi MELSEC Q

Operandi utilizzabili										Errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word							
—	—	—	—	—	—	—	—	—	—	SM0	6

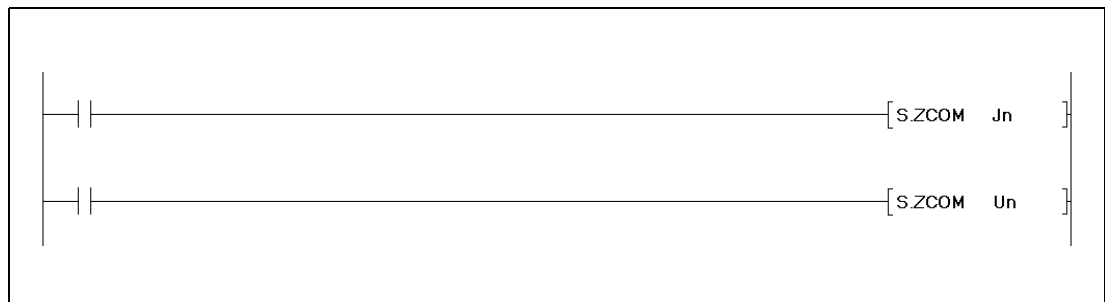
GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC								
<div style="border: 1px solid black; padding: 5px; display: inline-block; margin-bottom: 5px;">MELSEC</div> <table border="0" style="margin-left: 10px;"> <tr> <td>J.ZCOM</td> <td>Jn</td> </tr> <tr> <td>G.ZCOM</td> <td>Un</td> </tr> </table>	J.ZCOM	Jn	G.ZCOM	Un		<table border="0" style="margin-left: 10px;"> <tr> <td>ZCOM_J_M</td> <td>Jn</td> </tr> <tr> <td>ZCOM_U_M</td> <td>Un</td> </tr> </table>	ZCOM_J_M	Jn	ZCOM_U_M	Un
J.ZCOM	Jn									
G.ZCOM	Un									
ZCOM_J_M	Jn									
ZCOM_U_M	Un									

GX Developer (CPU QnA)



GX Developer (CPU System Q)



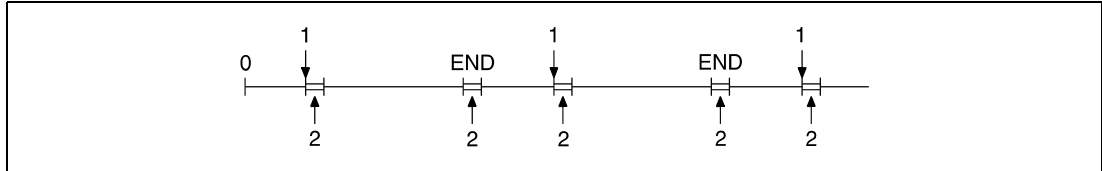
Variabili

Dati impostati	Significato	Tipo dati
Jn	Numero di rete della stazione host	BIN 16-bit
Un	Indirizzo intestazione I/O della rete della stazione host.	

Funzioni Rinfresco dati di rete

ZCOM Rinfresco dati su moduli di rete

A seguito dell'esecuzione dell'istruzione ZCOM, la CPU sospende l'elaborazione del programma di sequenza e rinfresca i dati nei moduli di rete indicati da Jn e Un.

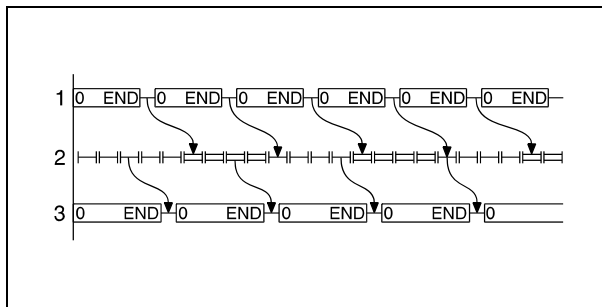


¹Esecuzione istruzione ZCOM

² Rinfresco dati

Nei casi in cui il tempo di scansione del programma di sequenza della stazione host supera il tempo di scansione delle altre stazioni, l'istruzione ZCOM assicura che i dati vengano incamerati correttamente dalle altre stazioni.

La figura seguente mostra un esempio di comunicazione dati senza l'uso dell'istruzione ZCOM:

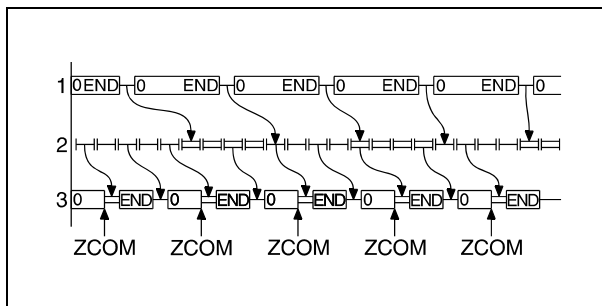


¹Programma della stazione di controllo

²Scansione programma della stazione collegata

³Programma della stazione normale

La figura seguente mostra un esempio di comunicazione dati con l'uso dell'istruzione ZCOM:

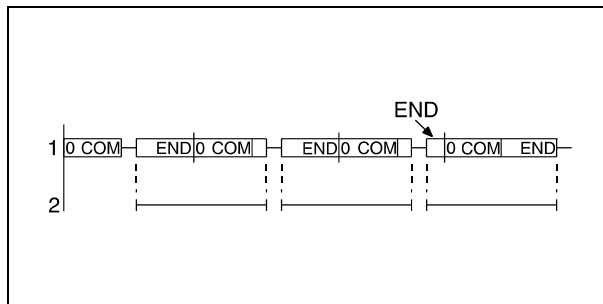


¹Programma della stazione di controllo

²Scansione programma della stazione collegata

³Programma della stazione normale

Nel caso in cui il tempo di scansione della stazione target è superiore a quello della stazione host, l'istruzione ZCOM non consente di migliorare la comunicazione dati.



¹Programma sequenza

²Tempo di scansione della stazione target

L'istruzione ZCOM può essere eseguita un numero qualsiasi di volte in un programma. Considerare tuttavia che ogni istruzione aumenta il tempo di scansione del programma sequenza del tempo di esecuzione del rinfresco dati.

L'istruzione ZCOM non può essere utilizzata nelle operazioni seguenti:

- Comunicazione fra CPU e unità periferiche.
- Monitoraggio di altre stazioni.
- Lettura di buffer di memoria di altri moduli funzione speciali, tramite modulo di collegamento a computer.

NOTA

Con le CPU serie Q e System Q, la designazione di "Un" fra gli argomenti, abilita l'accesso non solo ai moduli di rete, ma anche ai moduli funzione intelligenti. In questo caso il rinfresco automatico viene effettuato dal buffer di memoria del modulo funzione intelligente. (sostituisce le istruzioni FROM e TO).

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il numero di rete specificato non è collegato alla stazione host (codice di errore 4102).
- Il modulo per il numero di I/O specificato non è una unità di rete o una unità di comunicazione (codice di errore 2111).

NOTA

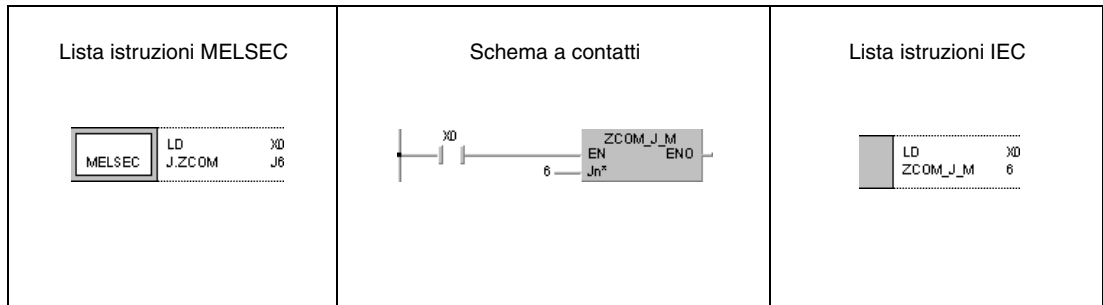
Nel caso di sola elaborazione di dati condivisi, utilizzare l'istruzione COM.

Notare che si possono verificare casi di non consistenza dei dati, cioè un operando può cambiare durante una scansione del programma.

Programma di esempio 1

J.ZCOM

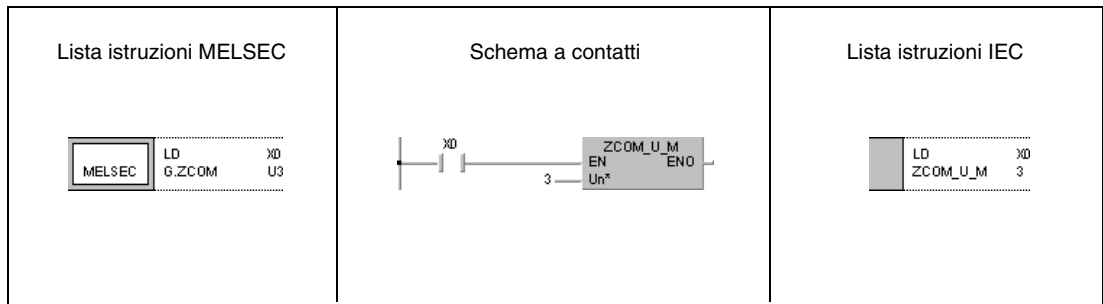
Con X0 attivo, il programma seguente rinfresca i dati nel modulo di rete con numero di rete 6.



Programma di esempio 2

G.ZCOM

Con X0 attivo, il programma seguente rinfresca i dati nel modulo di rete con gli indirizzi di I/O da X/Y30 a X/Y4F.



8.6 Istruzioni di comunicazione dedicate per la serie QnA

Queste istruzioni supportano la comunicazione dati tra stazioni con QCPU e fra CPU QnA e stazioni di I/O remoto nelle reti MELSECNET/10. La tabella seguente mostra una panoramica di queste istruzioni:

Funzione	MELSEC Istruzione in editor MELSEC	MELSEC Istruzione con Editor IEC	Stazione designata in MELSECNET/10			MELSECNET
			CPU QnA	ACPU	Stazione I/O remoto	
Lettura Dati CPU QnA da stazioni target in reti target	READ	READ_M	●	—	—	—
		READP_M				
		READ_JP_M				
		READ_UP_M				
	SREAD	SREAD_JP_M	●	—	—	—
		SREAD_UP_M				
Scrittura Dati CPU QnA su stazioni oggetto in reti target	WRITE	WRITE_JP_M	●	—	—	—
		WRITE_UP_M				
	SWRITE	SWRITE_M	●	—	—	—
		SWRITE_JP_M				
		SWRITE_UP_M				
	Invio dati a moduli di rete su stazioni target in reti target	SEND	SEND_M	●	—	—
SEND_4_M						
SEND_4_P_M						
SEND_JP_M						
SEND_UP_M						
Lettura dati CPU QnA inviati tramite istruzione SEND	RECV	RECV_M	●	—	—	—
		RECV_P_M				
		RECV_JP_M				
		RECV_UP_M				
Richiesta dati da altre stazioni (operazioni di lettura/scrittura con dati orologio, RUN/STOP remoto)	REQ	REQ_M	●	—	—	—
		REQP_M				
		REQ_JP_M				
		REQ_UP_M				
Lettura dati da moduli funzione speciali in stazioni I/O remote	ZNFR	ZNFR_JP_M	—	—	●	—
		ZNFR_UP_M				
Scrittura dati su moduli funzione speciali in stazioni I/O remote.	ZNT0	ZNT0_J_M	—	—	●	—
		ZNT0_U_M				
		ZNT0_JP_M				
		ZNT0_UP_M				

8.6.1 READ

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	

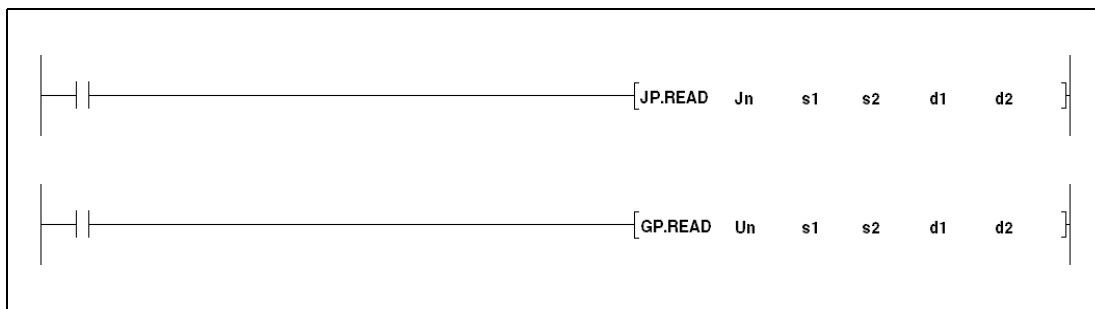
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	●	—	SM0	11
s2	—	●	●	—	—	—	—	●	—		
d1	—	●	●	—	—	—	—	—	—		
d2	●	●	●	—	—	—	—	—	—		

GX IEC Developer

<p style="text-align: center;">Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> <p>MELSEC</p> </div> <p>JP.READ Jn s1 s2 d1 d2</p> <p>GP.READ Un s1 s2 d1 d2</p>	<p style="text-align: center;">Schema a contatti</p> <div style="text-align: center; margin-bottom: 10px;"> <p>READ_JP_M</p> </div> <div style="text-align: center;"> <p>READ_UP_M</p> </div>	<p style="text-align: center;">Lista istruzioni IEC</p> <div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> <p>READ_JP_M Jn , s1 , s2 , d1 , d2</p> </div> <div style="border: 1px solid black; padding: 5px;"> <p>READ_UP_M Un , s1 , s2 , d1 , d2</p> </div>
---	---	--

GX Developer



Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
Jn	Numero di rete della stazione host ● ¹	BIN 16-bit	ANY16
Un	Indirizzo instazione I/O per unità di rete della stazione host. ● ²		
s1	Primo operando della stazione host che contiene i dati di controllo.	Indirizzo operando	Array [1...18] di ANY16
s2	Indirizzo del primo operando della stazione che contiene i dati da leggere.		ANY16
d1	Primo operando della stazione host che contiene i dati letti.		
d2	Operando attivato per una scansione al completamento dell'istruzione.	Bit	BOOL

NOTA

- ¹ Il numero di rete della stazione host può variare nel campo da 1 a 239. La rete di numero 254 viene configurata tramite parametrizzazione per consentire l'accesso alla stazione attiva da parte delle altre stazioni.
- ² L'indirizzo di intestazione di I/O dell'unità di rete per la stazione host deve essere nel campo da 0 a FE_H. Notare, che il compilatore considera Un come valore esadecimale. Un numero decimale viene convertito automaticamente in valore esadecimale.

L'istruzione READ può essere eseguita solo se la stazione target è una CPU QnA.

Con una ACPU su MELSECNET/10, l'istruzione READ non può essere usata.

Solo i numeri di stazione delle CPU QnA sono numeri validi per stazioni target.

Panoramica degli operandi per i dati di controllo

Operando	Significato	Funzione	Valore Campo	Imposta- to da
(s1)+0 Array_s1[1]	Modo di esecuzione	La conferma trasmissione è impostata: (Bit 0 (b0) = 1, fisso)	0001H 0081H	Utente
	Modo completa- mento errore	Memorizzazione dati impostazione orologio al termine dell'elaborazione errore: - nessuna memorizzazione dati orologio, Bit 7 (b7) = 0 - memorizzazione dati orologio, Bit 7 (b7) = 1 (dati orologio a partire da (s1)+11 (Array_s1[12]))		
(s1)+1 Array_s1[2]	Stato completa- mento esecuzione istruzione	Stato al completamente dell'esecuzione dell'istruzione: 0 = nessun errore (completamento normale) < > 0 = codice di errore ● ³	—	Sistema
(s1)+2 Array_s1[3]	Canale usato dalla stazione host	Designazione canale host.	da 1 a 8	Utente
(s1)+3 Array_s1[4]	Dummy	Non utilizzato	0	—
(s1)+4 Array_s1[5]	Numero di rete della stazione tar- get	Specifica il numero di rete della stazione da cui leg- gere.	da 1 a 239 254 ● ⁴	Utente
(s1)+5 Array_s1[6]	Numero stazione target	Specifica il numero della stazione target.	da 1 a 64	Utente
(s1)+6 Array_s1[7]	Dummy	Non utilizzato	—	—
(s1)+7 Array_s1[8]	Numero tentativi di trasmissione	Specifica il numero di tentativi per raggiungere il completamento dell'istruzione READ entro il tempo WDT memorizzato in (s1)+8 (Array_s1[9]).	da 1 a 15	Utente
	Numero tentativi di trasmissione ese- guiti	Contiene il numero dei tentativi di trasmissione ese- guiti.		Sistema
(s1)+8 Array_s1[9]	Impostazione tempo di trasmissione per WDT	Imposta il tempo di monitoraggio in secondi per le operazioni READ. Se l'operazione non viene terminata entro il tempo impostato, la trasmissione viene ripetuta per il numero di volte impostato in (s1)+7 (Array_s1[8]).	da 1 a 32767 0 = 10 s (fisso)	Utente
(s1)+9 Array_s1[10]	Lunghezza dati da leggere	Specifica la dimensione del blocco dati da leggere.	da 1 a 480	Utente
(s1)+10 Array_s1[11]	Dummy	Non utilizzato	—	—
(s1)+11 Array_s1[12]	Segnalazione orolo- gio impostato (attivato solo su errore)	Contiene lo stato di abilitazione/disabilitazione dati orologio impostato in (s1)+0 (Array_s1[1]): - Memorizzazione dati orologio disabilitata = 0 - Memorizzazione dati orologio abilitata = 1	—	Sistema
(s1)+12 Array_s1[13]	Dati orologio (attivato solo su errore)	Byte pesante = anno (da 0 a 99) Byte leggero = mese (da 1 a 12)	—	Sistema
(s1)+13 Array_s1[14]		Byte pesante = giorno (da 1 a 31) Byte leggero = ora (da 0 a 23)		
(s1)+14 Array_s1[15]		Byte pesante = minuto (da 0 a 59) Byte leggero = secondo (da 0 a 59)		
(s1)+15 Array_s1[16]		Byte pesante = 00H Byte leggero = giorno della settimana (da 0 a 6) (domenica = 0, sabato = 6)		
(s1)+16 Array_s1[17]	Numero della rete in cui si è verificato l'errore	Contiene il numero di rete della stazione in cui si è veri- ficato l'errore. Il numero di rete è compreso fra 1 e .	—	Sistema
(s1)+17 Array_s1[18]	Numero della sta- zione in cui si è verificato l'errore	Contiene il numero della stazione in cui si è veri- ficato l'errore. Il numero non viene memorizzato se lo stato di completamento dell'istruzione è "Chan- nel in use" (canale in uso). Il numero di stazione è compreso fra 1 e 64.	—	Sistema

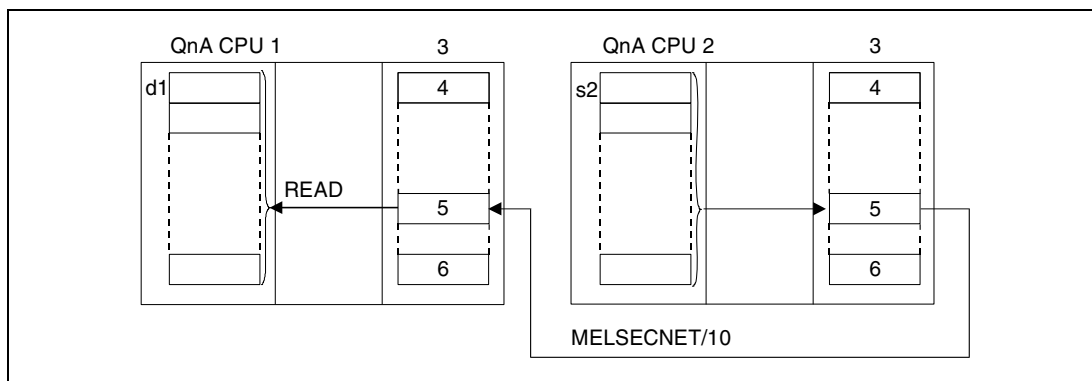
●³ Fare riferimento al manuale MELSECNET/10 per i sistemi di rete QnA per ulteriori dettagli.

●⁴ Il numero di rete 254 viene specificato inserendo 254 in Jn.

Funzioni Lettura dati operandi a word da una diversa stazione**READ Istruzione di lettura**

L'istruzione READ legge i dati contenuti a partire da s2 da una stazione collegata a MELSECNET/10. Il numero di stazione e di rete viene specificato nei dati di controllo. I dati letti dalla stazione vengono memorizzati a partire da d1 nella stazione host.

Al termine dell'operazione di lettura, viene attivato l'operando d2 della stazione target.



- ¹ Stazione host
- ² Stazione target
- ³ Modulo di rete
- ⁴ Canale 1
- ⁵ Canale n
- ⁶ Canale 8

Tramite una stazione ripetitrice e l'impostazione di parametri di instradamento, è possibile anche accedere a stazioni in reti diverse.

Le istruzioni di comunicazione non possono essere eseguite contemporaneamente con accesso comune allo stesso canale. In caso di esecuzione simultanea dell'istruzione da due o più locazioni, un meccanismo di interblocco fra le due stazioni attive impedisce l'esecuzione di ulteriori istruzioni di comunicazione.

Lo stato di esecuzione e lo stato di completamento (normale, anomalo) dell'istruzione READ può essere controllato tramite:

- il flag della direttiva di comunicazione (●⁵) del canale usato,
- l'operando di completamento della stazione host (d2) che viene attivato al termine dell'operazione,
- l'indicazione di stato del completamento operazione (completamento di trasmissione corretta o con errore) ((d2)+1)

come segue:

Flag direttiva di comunicazione

Questo flag è attivato durante l'esecuzione dell'istruzione READ. Il flag è azzerato con l'elaborazione dell'istruzione END alla fine della scansione di programma in cui viene completata l'operazione di lettura.

Operando di completamento stazione host

Questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione di lettura viene completata. L'operando viene azzerato con la successiva istruzione END.

Indicazione stato completamento dell'operazione

Questo operando viene attivato in funzione del risultato di completamento dell'istruzione di lettura.

Rimane a zero in caso di trasmissione normale (senza errori).

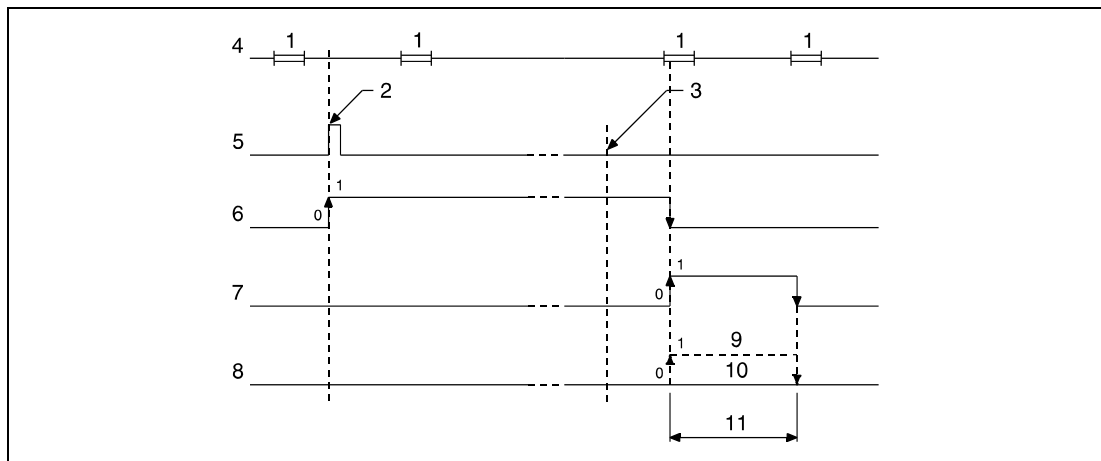
In caso di completamento con errore di una trasmissione, questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione READ viene completata. L'operando viene azzerato con la successiva istruzione END.

NOTA

●⁵ La tabella seguente assegna i numeri di canale secondo i flag di comunicazione del canale:

Numero canale	1	2	3	4	5	6	7	8
Flag canale di comunicazione	SB30	SB32	SB34	SB36	SB38	SB3A	SB3C	SB3E

La figura seguente mostra le operazioni della stazione host durante l'esecuzione di una istruzione READ:



- ¹ Elaborazione END
- ² Esecuzione istruzione READ
- ³ Completamento operazione
- ⁴ Programma della stazione host
- ⁵ Istruzione READ
- ⁶ Flag canale di comunicazione
- ⁷ Operando di completamento della stazione host (d2) attivato al termine dell'operazione
- ⁸ Indicazione stato completamento dell'operazione ((d2)+1)
- ⁹ Completamento trasmissione con errore
- ¹⁰ Completamento di una trasmissione senza errori
- ¹¹ Una scansione

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- I contenuti dei dati di controllo superano i campi ammessi (codice di errore 4100).
- La rete con il numero specificato da Jn non è collegata alla stazione (codice di errore 4102).
- Il modulo con l'indirizzo di I/O specificato da Un non è un modulo di rete (codice di errore 2111).

8.6.2 SREAD

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	

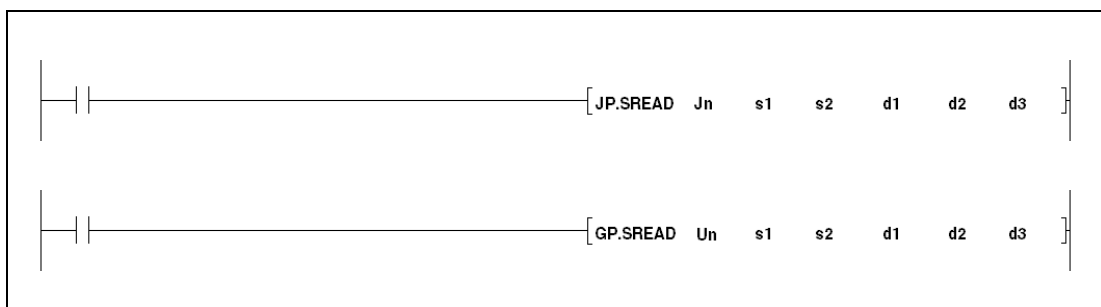
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	●	—	SM0	13
s2	—	●	●	—	—	—	—	●	—		
d1	—	●	●	—	—	—	—	—	—		
d2	●	●	●	—	—	—	—	—	—		
d3	●	●	●	—	—	—	—	—	—		

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
<div style="border: 1px solid black; padding: 5px; width: fit-content;">MELSEC</div> <p>JP.SREAD Jn s1 s2 d1 d2 d3</p> <p>GP.SREAD Un s1 s2 d1 d2 d3</p>		<div style="border: 1px solid black; padding: 5px; width: fit-content;">SREAD</div> <p>SREAD_JP_M Jn , s1 , s2 , d1 , d2 , d3</p> <p>SREAD_UP_M Un , s1 , s2 , d1 , d2 , d3</p>

GX Developer



Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
Jn	Numero di rete della stazione host ● ¹	BIN 16-bit	ANY16
Un	Indirizzo intestazione I/O per unità di rete della stazione host. ● ²		
s1	Primo operando della stazione host che contiene i dati di controllo.	Indirizzo operando	Array [1..18] di ANY16
s2	Indirizzo del primo operando della stazione che contiene i dati da leggere.		ANY16
d1	Primo operando della stazione host che contiene i dati letti.		
d2	Operando della stazione host attivato per una scansione al completamento dell'istruzione.	Bit	BOOL
d3	Operando della stazione target attivato per una scansione al completamento dell'istruzione.		

NOTA

- ¹ Il numero di rete della stazione host può variare nel campo da 1 a 239. La rete di numero 254 viene configurata tramite parametrizzazione per consentire l'accesso alla stazione attiva da parte delle altre stazioni.
- ² L'indirizzo di intestazione di I/O dell'unità di rete per la stazione host deve essere nel campo da 0 a FEH. Notare, che il compilatore considera Un come valore esadecimale. Un numero decimale viene convertito automaticamente in valore esadecimale.

L'istruzione SREAD può essere eseguita solo se la stazione target è una CPU QnA.

Con una ACPU su MELSECNET/10, l'istruzione SREAD non può essere usata.

Solo i numeri di stazione delle CPU QnA sono numeri validi per stazioni target.

Panoramica degli operandi per i dati di controllo

Operando	Significato	Funzione	Valore Campo	Imposta- to da
(s1)+0 Array_s1[1]	Modo di esecuzione	La conferma trasmissione è impostata: (Bit 0 (b0) = 1, fisso)	0001H 0081H	Utente
	Modo completa- mento errore	Memorizzazione dati impostazione orologio al termine dell'elaborazione errore: - nessuna memorizzazione dati orologio, Bit 7 (b7) = 0 - memorizzazione dati orologio, Bit 7 (b7) = 1 (dati orologio a partire da (s1)+11 (Array_s1[12]))		
(s1)+1 Array_s1[2]	Stato completa- mento esecuzione istruzione	Stato al completamente dell'esecuzione dell'istruzione: 0 = nessun errore (completamento normale) < > 0 = codice di errore ● ³	—	Sistema
(s1)+2 Array_s1[3]	Canale usato dalla stazione host	Designazione canale host.	da 1 a 8	Utente
(s1)+3 Array_s1[4]	Dummy	Non utilizzato	0	—
(s1)+4 Array_s1[5]	Numero di rete della stazione target	Specifica il numero di rete della stazione da cui leggere.	da 1 a 239 254 ● ⁴	Utente
(s1)+5 Array_s1[6]	Numero stazione target	Specifica il numero della stazione target.	da 1 a 64	Utente
(s1)+6 Array_s1[7]	Dummy	Non utilizzato	—	—
(s1)+7 Array_s1[8]	Numero tentativi di trasmissione	Specifica il numero di tentativi per raggiungere il completamento dell'istruzione READ entro il tempo WDT memorizzato in (s1)+8 (Array_s1[9]).	da 1 a 15	Utente
	Numero tentativi di trasmissione ese- guiti	Contiene il numero dei tentativi di trasmissione eseguiti.		Sistema
(s1)+8 Array_s1[9]	Impostazione tempo di trasmissione per WDT	Imposta il tempo di monitoraggio in secondi per le operazioni READ. Se l'operazione non viene terminata entro il tempo impostato, la trasmissione viene ripetuta per il numero di volte impostato in (s1)+7 (Array_s1[8]).	da 1 a 32767 0 = 10 s (fisso)	Utente
(s1)+9 Array_s1[10]	Lunghezza dati da leggere	Specifica la dimensione del blocco dati da leggere.	da 1 a 480	Utente
(s1)+10 Array_s1[11]	Dummy	Non utilizzato	—	—
(s1)+11 Array_s1[12]	Segnalazione orolo- gio impostato (attivato solo su errore)	Contiene lo stato di abilitazione/disabilitazione dati orologio impostato in (s1)+0 (Array_s1[1]): - Memorizzazione dati orologio disabilitata = 0 - Memorizzazione dati orologio abilitata = 1	—	Sistema
(s1)+12 Array_s1[13]	Dati orologio (attivato solo su errore)	Byte pesante = anno (da 0 a 99) Byte leggero = mese (da 1 a 12)	—	Sistema
(s1)+13 Array_s1[14]		Byte pesante = giorno (da 1 a 31) Byte leggero = ora (da 0 a 23)		
(s1)+14 Array_s1[15]		Byte pesante = minuto (da 0 a 59) Byte leggero = secondo (da 0 a 59)		
(s1)+15 Array_s1[16]		Byte pesante = 00H Byte leggero = giorno della settimana (da 0 a 6) (domenica = 0, sabato = 6)		
(s1)+16 Array_s1[17]	Numero della rete in cui si è verificato l'errore	Contiene il numero di rete della stazione in cui si è verificato l'errore. Il numero di rete è compreso fra 1 e 239.	—	Sistema
(s1)+17 Array_s1[18]	Numero della sta- zione in cui si è verificato l'errore	Contiene il numero della stazione in cui si è verificato l'errore. Il numero non viene memorizzato se lo stato di completamento dell'istruzione è "Channel in use" (canale in uso). Il numero di stazione è compreso fra 1 e 64.	—	Sistema

●³ Fare riferimento al manuale MELSECNET/10 per i sistemi di rete QnA per ulteriori dettagli.

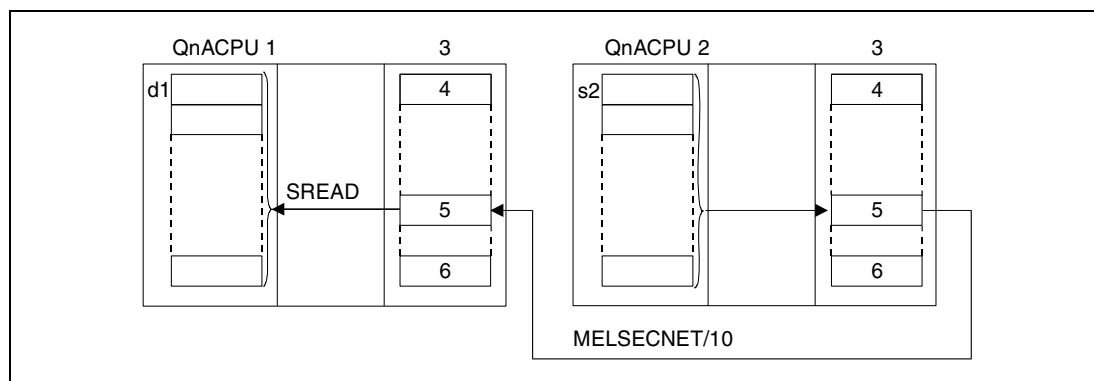
●⁴ Il numero di rete 254 viene specificato inserendo 254 in Jn.

Funzioni **Letture dati operandi a word da una diversa stazione**

SREAD Istruzione di lettura

L'istruzione SREAD legge i dati contenuti a partire da s2 da una stazione collegata a MELSECNET/10. Il numero di stazione e di rete viene specificato nei dati di controllo. I dati letti dalla stazione vengono memorizzati a partire da d1 nella stazione host.

Al termine dell'operazione di lettura vengono attivati gli operandi d2 della stazione host e d3 della stazione target.



- ¹ Stazione host
- ² Stazione target
- ³ Modulo di rete
- ⁴ Canale 1
- ⁵ Canale n
- ⁶ Canale 8

Tramite una stazione ripetitrice e l'impostazione di parametri di instradamento, è possibile anche accedere a stazioni in reti diverse.

Le istruzioni di comunicazione non possono essere eseguite contemporaneamente con accesso comune allo stesso canale. In caso di esecuzione simultanea dell'istruzione da due o più locazioni, un meccanismo di interblocco fra le due stazioni attive impedisce l'esecuzione di ulteriori istruzioni di comunicazione.

Lo stato di esecuzione e lo stato di completamento (normale, anomalo) dell'istruzione SREAD può essere controllato tramite:

- il flag della direttiva di comunicazione (●⁵) del canale usato,
- l'operando di completamento della stazione host (d2) e quello della stazione target (d3) che viene attivato al termine dell'operazione,
- l'indicazione di stato del completamento operazione (completamento di trasmissione corretta o con errore) ((d2)+1)

come segue:

Flag direttiva di comunicazione

Questo flag è attivato durante l'esecuzione dell'istruzione SREAD. Il flag è azzerato con l'elaborazione dell'istruzione END alla fine della scansione di programma in cui viene completata l'operazione di lettura.

Operando di completamento stazione host

Questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione di lettura viene completata. L'operando viene azzerato con la successiva istruzione END.

Indicazione stato completamento dell'operazione

Questo operando viene attivato in funzione del risultato di completamento dell'istruzione di lettura.

Rimane a zero in caso di trasmissione normale (senza errori).

In caso di completamento con errore di una trasmissione, questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione SREAD viene completata. L'operando viene azzerato con la successiva istruzione END.

Operando di completamento stazione target

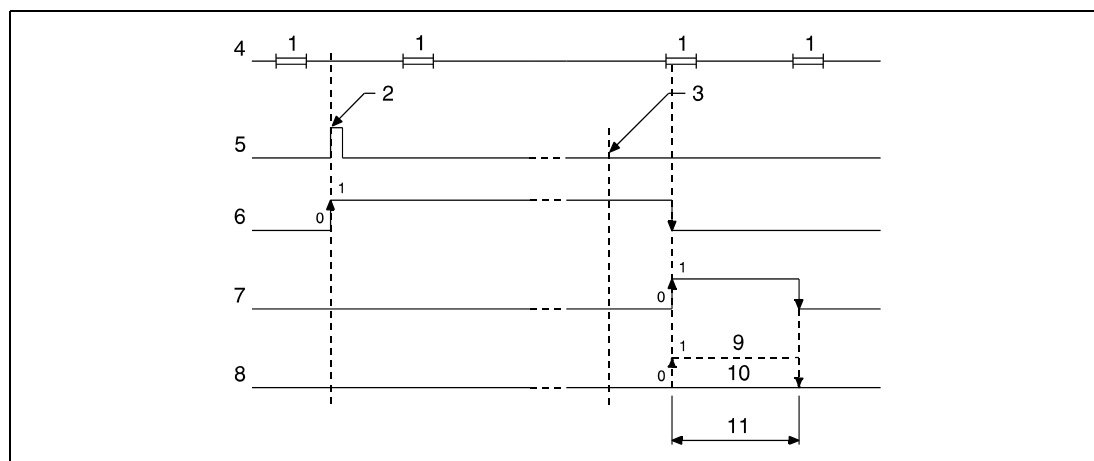
Questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione di lettura viene completata. L'operando viene azzerato con la successiva istruzione END.

NOTA

- ⁵ La tabella seguente assegna i numeri di canale secondo i flag di comunicazione del canale:

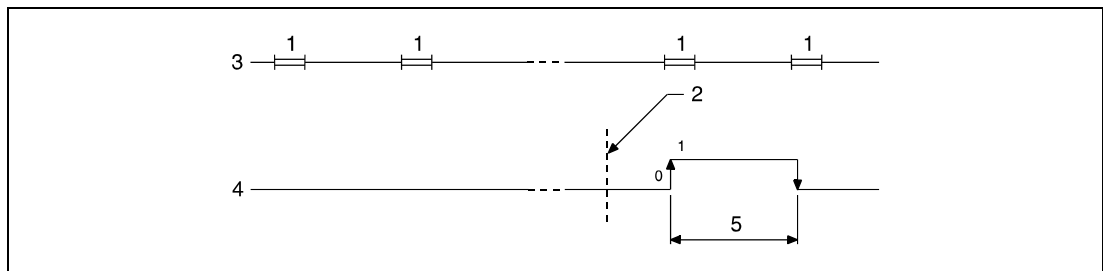
Numero canale	1	2	3	4	5	6	7	8
Flag canale di comunicazione	SB30	SB32	SB34	SB36	SB38	SB3A	SB3C	SB3E

La figura seguente mostra le operazioni della stazione host durante l'esecuzione di una istruzione SREAD:



- 1 Elaborazione END
- 2 Esecuzione istruzione SREAD
- 3 Completamento operazione
- 4 Programma della stazione host
- 5 Istruzione SREAD
- 6 Flag canale di comunicazione
- 7 Operando di completamento della stazione host (d2) attivato al termine dell'operazione
- 8 Indicazione stato completamento dell'operazione ((d2)+1)
- 9 Completamento trasmissione con errore
- 10 Completamento di una trasmissione senza errori
- 11 Una scansione

La figura seguente mostra le operazioni della stazione target durante l'esecuzione di una istruzione SREAD:



1 Elaborazione END

2 Completamento operazione

3 Programma della stazione target

4 Operando di completamento della stazione target (d3) attivato al termine dell'operazione

5 Una scansione

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- I contenuti dei dati di controllo superano i campi ammessi (codice di errore 4100).
- La rete con il numero specificato da Jn non è collegata alla stazione (codice di errore 4102).
- Il modulo con l'indirizzo di I/O specificato da Un non è un modulo di rete (codice di errore 2111).

8.6.3 WRITE

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	

Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	SM0	12	
s2	—	●	●	—	—	—	—	—			
d1	●	●	●	—	—	—	—	—			
d2	●	●	●	—	—	—	—	—			

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
<div style="border: 1px solid black; padding: 5px; width: fit-content;">MELSEC</div> <p>JP.WRITE Jn s1 s2 d1 d2</p> <p>GP.WRITE Un s1 s2 d1 d2</p>	<p style="text-align: center;">WRITE_JP_M</p> <p style="text-align: center;">WRITE_UP_M</p>	<p>WRITE_JP_M Jn , s1 , s2 , d1 , d2</p> <p>WRITE_UP_M Un , s1 , s2 , d1 , d2</p>

Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
Jn	Numero di rete della stazione host ● ¹	BIN 16-bit	ANY16
Un	Indirizzo intestazione I/O per unità di rete della stazione host. ● ²		
s1	Primo operando della stazione host che contiene i dati di controllo.	Indirizzo operando	Array [1..18] di ANY16
s2	Primo operando della stazione che contiene i dati da scrivere.		ANY16
d1	Primo operando della stazione target che contiene i dati scritti.		
d2	Operando attivato per una scansione al completamento dell'istruzione.	Bit	BOOL

NOTA

- ¹ Il numero di rete della stazione host può variare nel campo da 1 a 239. La rete di numero 254 viene configurata tramite parametrizzazione per consentire l'accesso alla stazione attiva da parte delle altre stazioni.
- ² L'indirizzo di intestazione di I/O dell'unità di rete per la stazione host deve essere nel campo da 0 a FF_H. Notare, che il compilatore considera Un come valore esadecimale. Un numero decimale viene convertito automaticamente in valore esadecimale.

L'istruzione WRITE può essere eseguita solo se la stazione target è una CPU QnA.

Con una ACPU su MELSECNET/10, l'istruzione WRITE non può essere usata.

L'istruzione WRITE può utilizzare l'indirizzo "FF_H" (tutte le stazioni della rete target) solo su reti che collegano solo CPU QnA. L'indirizzo "FF_H" non può essere usato per reti che collegano CPU QnA e serie A.

Panoramica degli operandi per i dati di controllo

Operando	Significato	Funzione	Valore Campo	Impostato da
(s1)+0 Array_s1[1]	Modo di esecuzione	Conferma del completamento trasmissione = imposta il bit 0 su 0 Nessuna conferma del completamento trasmissione = imposta il bit 0 su 1	0000H 0001H 0080H 0081H	Utente
	Modo completamento errore	Memorizzazione dati impostazione orologio al termine dell'elaborazione errore: - nessuna memorizzazione dati orologio, Bit 7 (b7) = 0 - memorizzazione dati orologio, Bit 7 (b7) = 1 (dati orologio a partire da (s1)+10 (Array_s1[11]))		
(s1)+1 Array_s1[2]	Stato completamento esecuzione istruzione	Stato al completamento dell'esecuzione dell'istruzione: 0 = nessun errore (completamento normale) < > 0 = codice di errore ● ³	—	Sistema
(s1)+2 Array_s1[3]	Canale usato dalla stazione host	Designazione canale host.	da 1 a 8	Utente
(s1)+3 Array_s1[4]	Dummy	Non utilizzato	0	—
(s1)+4 Array_s1[5]	Numero di rete della stazione target	Specifica il numero di rete della stazione target.	da 1 a 239 254 ● ⁴	Utente
(s1)+5 Array_s1[6]	Numero stazione target	Specifica il numero della stazione target.	Numero stazione: da 1 a 64 Indicazione gruppo: da 81H a 89H Tutte le stazioni della rete target: FFH	Utente
(s1)+6 Array_s1[7]	Dummy	Non utilizzato	—	—
(s1)+7 Array_s1[8]	Numero tentativi di trasmissione	Specifica il numero di tentativi per raggiungere il completamento dell'istruzione WRITE entro il tempo WDT memorizzato in (s1)+8 (Array_s1[9]). Attivo solo se il modo esecuzione in (s1)+0 Array_s1[1] è attivo (1).	da 1 a 15	Utente
	Numero tentativi di trasmissione eseguiti	Contiene il numero dei tentativi di trasmissione eseguiti.		Sistema
(s1)+8 Array_s1[9]	Impostazione tempo di trasmissione per WDT	Imposta il tempo di monitoraggio in secondi per le operazioni WRITE. Se l'operazione non viene terminata entro il tempo impostato, la trasmissione viene ripetuta per il numero di volte impostato in (s1)+7 (Array_s1[8]).	da 1 a 32767 0 = 10 s (fisso) Attivo solo se il modo esecuzione in (s1)+0 Array_s1[1] è attivo (1).	Utente
(s1)+9 Array_s1[10]	Lunghezza dati inviati	Specifica il numero di blocchi dati da scrivere.	da 1 a 480	Utente
(s1)+10 Array_s1[11]	Dummy	Non utilizzato	—	—
(s1)+11 Array_s1[12]	Segnalazione orologio impostato (attivato solo su errore)	Contiene lo stato di abilitazione/disabilitazione dati orologio impostato in (s1)+0 (Array_s1[1]): - Memorizzazione dati orologio disabilitata = 0 - Memorizzazione dati orologio abilitata = 1	—	Sistema

Operando	Significato	Funzione	Valore Campo	Impostato da
(s1)+12 Array_s1[13]	Dati orologio (attivato solo su errore)	Byte pesante = anno (da 0 a 99) Byte leggero = mese (da 1 a 12)	—	Sistema
(s1)+13 Array_s1[14]		Byte pesante = giorno (da 1 a 31) Byte leggero = ora (da 0 a 23)		
(s1)+14 Array_s1[15]		Byte pesante = minuto (da 0 a 59) Byte leggero = secondo (da 0 a 59)		
(s1)+15 Array_s1[16]		Byte pesante = 00H Byte leggero = giorno della settimana (da 0 a 6) (domenica = 0, sabato = 6)		
(s1)+16 Array_s1[17]	Numero della rete in cui si è verificato l'errore	Contiene il numero di rete della stazione in cui si è verificato l'errore. Il numero di rete è compreso fra 1 e 239.	—	Sistema
(s1)+17 Array_s1[18]	Numero della stazione in cui si è verificato l'errore	Contiene il numero della stazione in cui si è verificato l'errore. Il numero non viene memorizzato se lo stato di completamento dell'istruzione è "Channel in use" (canale in uso). Il numero di stazione è compreso fra 1 e 64.	—	Sistema

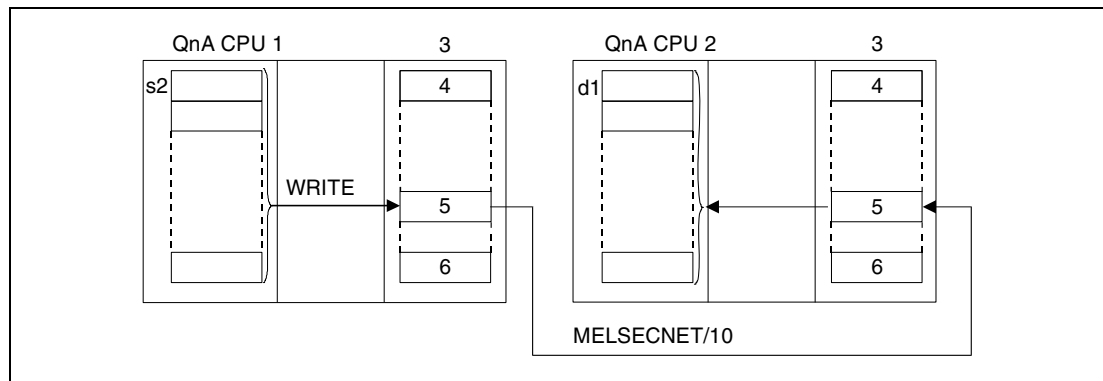
●³ Fare riferimento al manuale MELSECNET/10 per i sistemi di rete QnA per ulteriori dettagli.

●⁴ Il numero di rete 254 viene specificato inserendo 254 in Jn.

Funzioni Scrittura dati operando a word su una diversa stazione**WRITE Istruzione di scrittura**

L'istruzione WRITE scrive i dati contenuti a partire da s2 di una stazione host in una stazione collegata a MELSECNET/10. Il numero di stazione e di rete viene specificato nei dati di controllo. I dati vengono memorizzati nella stazione target a partire da d1.

Al termine dell'operazione di scrittura, viene attivato l'operando d2 della stazione target.



- ¹ Stazione host
- ² Stazione target
- ³ Modulo di rete
- ⁴ Canale 1
- ⁵ Canale n
- ⁶ Canale 8

Tramite una stazione ripetitrice e l'impostazione di parametri di instradamento, è possibile anche accedere a stazioni in reti diverse.

Le istruzioni di comunicazione non possono essere eseguite contemporaneamente da più locazioni con accesso comune allo stesso canale. In caso di esecuzione simultanea dell'istruzione da due o più locazioni, un meccanismo di interblocco fra le due stazioni attive impedisce l'esecuzione di ulteriori istruzioni di comunicazione.

Lo stato di esecuzione e lo stato di completamento (normale, anomalo) dell'istruzione WRITE può essere controllato tramite:

- il flag della direttiva di comunicazione (●⁵) del canale usato,
- l'operando di completamento della stazione host (d2) che viene attivato al termine dell'operazione,
- l'indicazione di stato del completamento operazione (completamento di trasmissione corretta o con errore) ((d2)+1)

come segue:

Flag direttiva di comunicazione

Questo flag è attivato durante l'esecuzione dell'istruzione WRITE. Il flag è azzerato con l'elaborazione dell'istruzione END alla fine della scansione di programma in cui viene completata l'operazione di scrittura.

Operando di completamento stazione host

Questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione di scrittura viene completata. L'operando viene azzerato con la successiva istruzione END.

Indicazione stato completamento dell'operazione

Questo operando viene attivato in funzione del risultato di completamento dell'istruzione di scrittura.

Rimane a zero in caso di trasmissione normale (senza errori).

In caso di completamento con errore di una trasmissione, questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione WRITE viene completata. L'operando viene azzerato con la successiva istruzione END.

Operando di completamento stazione target

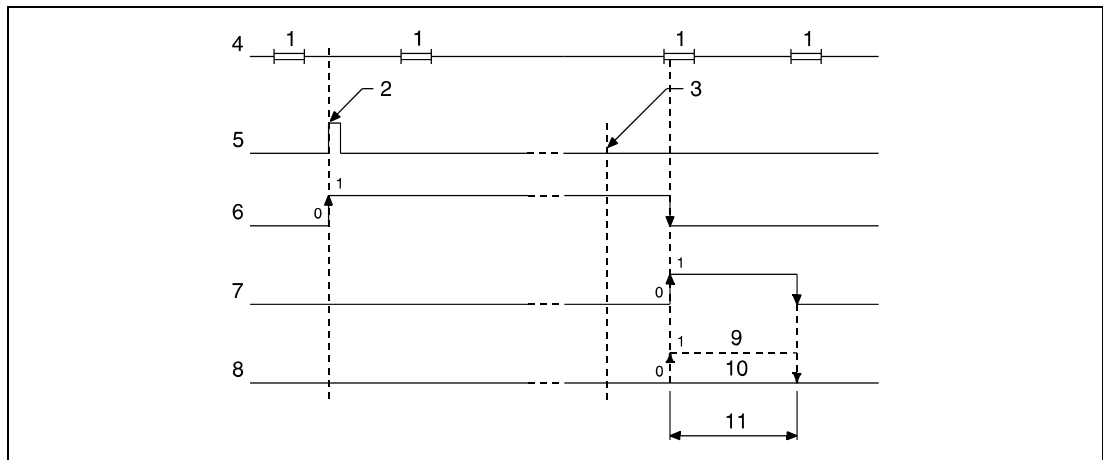
Questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione di lettura viene completata. L'operando viene azzerato con la successiva istruzione END.

NOTA

●⁵ La tabella seguente assegna i numeri di canale secondo i flag di comunicazione del canale:

Numero canale	1	2	3	4	5	6	7	8
Flag canale di comunicazione	SB30	SB32	SB34	SB36	SB38	SB3A	SB3C	SB3E

La figura seguente mostra le operazioni della stazione host durante l'esecuzione di una istruzione WRITE:



- 1 Elaborazione END
- 2 Esecuzione istruzione WRITE
- 3 Completamento operazione
- 4 Programma della stazione host
- 5 Istruzione WRITE
- 6 Flag canale di comunicazione
- 7 Operando di completamento della stazione host (d2) attivato al termine dell'operazione
- 8 Indicazione stato completamento dell'operazione ((d2)+1)
- 9 Completamento trasmissione con errore
- 10 Completamento di una trasmissione senza errori
- 11 Una scansione

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- I contenuti dei dati di controllo superano i campi ammessi (codice di errore 4100).
- La rete con il numero specificato da Jn non è collegata alla stazione (codice di errore 4102).
- Il modulo con l'indirizzo di I/O specificato da Un non è un modulo di rete (codice di errore 2111).

8.6.4 SWRITE

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	

Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	SM0	13	
s2	—	●	●	—	—	—	—	—			
d1	●	●	●	—	—	—	—	—			
d2	●	●	●	—	—	—	—	—			
d3	●	●	●	—	—	—	—	—			

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
<div style="border: 1px solid black; padding: 5px; width: fit-content;">MELSEC</div> <p>JP.SWRITE Jn s1 s2 d1 d2 d3</p> <p>GP.SWRITE Un s1 s2 d1 d2 d3</p>		<p>SWRITE_JP_M Jn , s1 , s2 , d1 , d2 , d3</p> <p>SWRITE_UP_M Un , s1 , s2 , d1 , d2 , d3</p>

Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
Jn	Numero di rete della stazione host ● ¹	BIN 16-bit	ANY16
Un	Indirizzo intestazione I/O per unità di rete della stazione host. ● ²		
s1	Primo operando della stazione host che contiene i dati di controllo.	Indirizzo operando	Array [1..18] di ANY16
s2	Primo operando della stazione che contiene i dati da scrivere.		ANY16
d1	Primo operando della stazione target che contiene i dati scritti.	Bit	BOOL
d2	Operando attivato per una scansione al completamento dell'istruzione.		
d3	Operando della stazione target attivato per una scansione al completamento dell'istruzione.		

NOTA

- ¹ Il numero di rete della stazione host può variare nel campo da 1 a 239. La rete di numero 254 viene configurata tramite parametrizzazione per consentire l'accesso alla stazione attiva da parte delle altre stazioni.
- ² L'indirizzo di intestazione di I/O dell'unità di rete per la stazione host deve essere nel campo da 0 a FF_H. Notare, che il compilatore considera Un come valore esadecimale. Un numero decimale viene convertito automaticamente in valore esadecimale.

L'istruzione SWRITE può essere eseguita solo se la stazione target è una CPU QnA.

Con una ACPU su MELSECNET/10, l'istruzione SWRITE non può essere usata.

L'istruzione SWRITE può utilizzare l'indirizzo "FF_H" (tutte le stazioni della rete target) solo su reti che collegano solo CPU QnA. L'indirizzo "FF_H" non può essere usato per reti che collegano CPU QnA e serie A.

Panoramica degli operandi per i dati di controllo

Operando	Significato	Funzione	Valore Campo	Impostato da
(s1)+0 Array_s1[1]	Modo di esecuzione	Conferma del completamento trasmissione = imposta il bit 0 su 0 Nessuna conferma del completamento trasmissione = imposta il bit 0 su 1	0000H 0001H 0080H 0081H	Utente
	Modo completamento errore	Memorizzazione dati impostazione orologio al termine dell'elaborazione errore: - nessuna memorizzazione dati orologio, Bit 7 (b7) = 0 - memorizzazione dati orologio, Bit 7 (b7) = 1 (dati orologio a partire da (s1)+10 (Array_s1[11]))		
(s1)+1 Array_s1[2]	Stato completamento esecuzione istruzione	Stato al completamento dell'esecuzione dell'istruzione: 0 = nessun errore (completamento normale) < > 0 = codice di errore ● ³	—	Sistema
(s1)+2 Array_s1[3]	Canale usato dalla stazione host	Designazione canale host.	da 1 a 8	Utente
(s1)+3 Array_s1[4]	Dummy	Non utilizzato	0	—
(s1)+4 Array_s1[5]	Numero di rete della stazione target	Specifica il numero di rete della stazione target.	da 1 a 239 254 ● ⁴	Utente
(s1)+5 Array_s1[6]	Numero stazione target	Specifica il numero della stazione target.	Numero stazione: da 1 a 64 Indicazione gruppo: da 81H a 89H Tutte le stazioni della rete target: FFH	Utente
(s1)+6 Array_s1[7]	Dummy	Non utilizzato	—	—
(s1)+7 Array_s1[8]	Numero tentativi di trasmissione	Specifica il numero di tentativi per raggiungere il completamento dell'istruzione WRITE entro il tempo WDT memorizzato in (s1)+8 (Array_s1[9]). Attivo solo se il modo esecuzione in (s1)+0 Array_s1[1] è attivo (1).	da 1 a 15	Utente
	Numero tentativi di trasmissione eseguiti	Contiene il numero dei tentativi di trasmissione eseguiti.		Sistema
(s1)+8 Array_s1[9]	Impostazione tempo di trasmissione per WDT	Imposta il tempo di monitoraggio in secondi per le operazioni WRITE. Se l'operazione non viene terminata entro il tempo impostato, la trasmissione viene ripetuta per il numero di volte impostato in (s1)+7 (Array_s1[8]).	da 1 a 32767 0 = 10 s (fisso) Attivo solo se il modo esecuzione in (s1)+0 Array_s1[1] è attivo (1).	Utente
(s1)+9 Array_s1[10]	Lunghezza dati inviati	Specifica il numero di blocchi dati da scrivere.	da 1 a 480	Utente
(s1)+10 Array_s1[11]	Dummy	Non utilizzato	—	—
(s1)+11 Array_s1[12]	Segnalazione orologio impostato (attivato solo su errore)	Contiene lo stato di abilitazione/disabilitazione dati orologio impostato in (s1)+0 (Array_s1[1]): - Memorizzazione dati orologio disabilitata = 0 - Memorizzazione dati orologio abilitata = 1	—	Sistema

Operando	Significato	Funzione	Valore Campo	Impostato da
(s1)+12 Array_s1[13]	Dati orologio (attivato solo su errore)	Byte pesante = anno (da 0 a 99) Byte leggero = mese (da 1 a 12)	—	Sistema
(s1)+13 Array_s1[14]		Byte pesante = giorno (da 1 a 31) Byte leggero = ora (da 0 a 23)		
(s1)+14 Array_s1[15]		Byte pesante = minuto (da 0 a 59) Byte leggero = secondo (da 0 a 59)		
(s1)+15 Array_s1[16]		Byte pesante = 00h Byte leggero = giorno della settimana (da 0 a 6) (domenica = 0, sabato = 6)		
(s1)+16 Array_s1[17]	Numero della rete in cui si è verificato l'errore	Contiene il numero di rete della stazione in cui si è verificato l'errore. Il numero di rete è compreso fra 1 e 239.	—	Sistema
(s1)+17 Array_s1[18]	Numero della stazione in cui si è verificato l'errore	Contiene il numero della stazione in cui si è verificato l'errore. Il numero non viene memorizzato se lo stato di completamento dell'istruzione è "Channel in use" (canale in uso). Il numero di stazione è compreso fra 1 e 64.	—	Sistema

●³ Fare riferimento al manuale MELSECNET/10 per i sistemi di rete QnA per ulteriori dettagli.

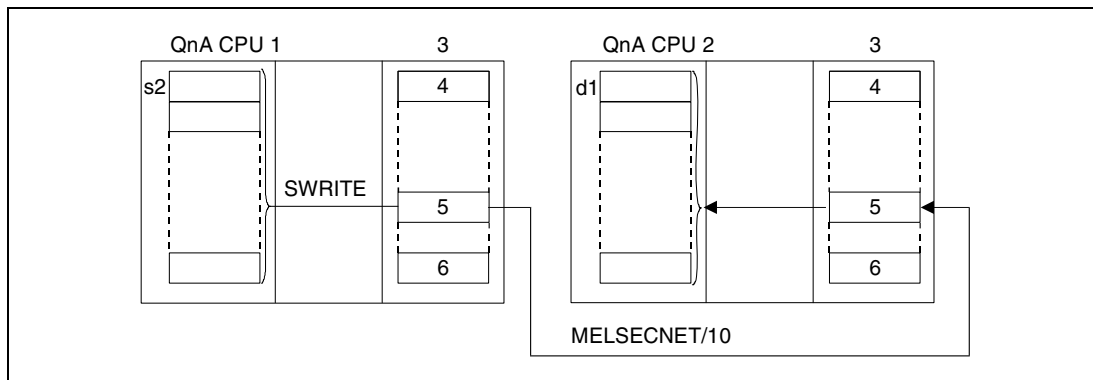
●⁴ Il numero di rete 254 viene specificato inserendo 254 in Jn.

Funzioni Scrittura dati operando a word su una diversa stazione

SWRITE Istruzione di scrittura

L'istruzione SWRITE scrive i dati contenuti a partire da s2 di una stazione host in una stazione collegata a MELSECNET/10. Il numero di stazione e di rete viene specificato nei dati di controllo. I dati vengono memorizzati nella stazione target a partire da d1.

Al termine dell'operazione di scrittura vengono attivati gli operandi d2 della stazione host e d3 della stazione target.



- 1 Stazione host
- 2 Stazione target
- 3 Modulo di rete
- 4 Canale 1
- 5 Canale n
- 6 Canale 8

Tramite una stazione ripetitrice e l'impostazione di parametri di instradamento, è possibile anche accedere a stazioni in reti diverse.

Le istruzioni di comunicazione non possono essere eseguite contemporaneamente da più locazioni con accesso comune allo stesso canale. In caso di esecuzione simultanea dell'istruzione da due o più locazioni, un meccanismo di interblocco fra le due stazioni attive impedisce l'esecuzione di ulteriori istruzioni di comunicazione.

Lo stato di esecuzione e lo stato di completamento (normale, anomalo) dell'istruzione SWRITE può essere controllato tramite:

- il flag della direttiva di comunicazione (●⁵) del canale usato,
- l'operando di completamento della stazione host (d2) e quello della stazione target (d3) che viene attivato al termine dell'operazione,
- l'indicazione di stato del completamento operazione (completamento di trasmissione corretta o con errore) ((d2)+1)

come segue:

Flag direttiva di comunicazione

Questo flag è attivato durante l'esecuzione dell'istruzione SWRITE. Il flag è azzerato con l'elaborazione dell'istruzione END alla fine della scansione di programma in cui viene completata l'operazione di scrittura.

Operando di completamento stazione host

Questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione di scrittura viene completata. L'operando viene azzerato con la successiva istruzione END.

Indicazione stato completamento dell'operazione

Questo operando viene attivato in funzione del risultato di completamento dell'istruzione di scrittura.

Rimane a zero in caso di trasmissione normale (senza errori).

In caso di completamento con errore di una trasmissione, questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione SWRITE viene completata. L'operando viene azzerato con la successiva istruzione END.

Operando di completamento stazione target

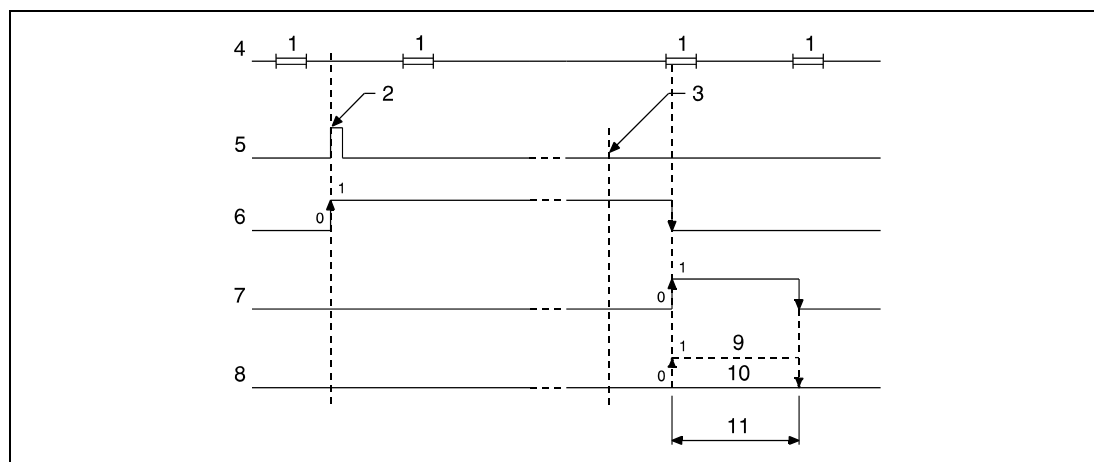
Questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione di lettura viene completata. L'operando viene azzerato con la successiva istruzione END.

NOTA

- ⁵ La tabella seguente assegna i numeri di canale secondo i flag di comunicazione del canale:

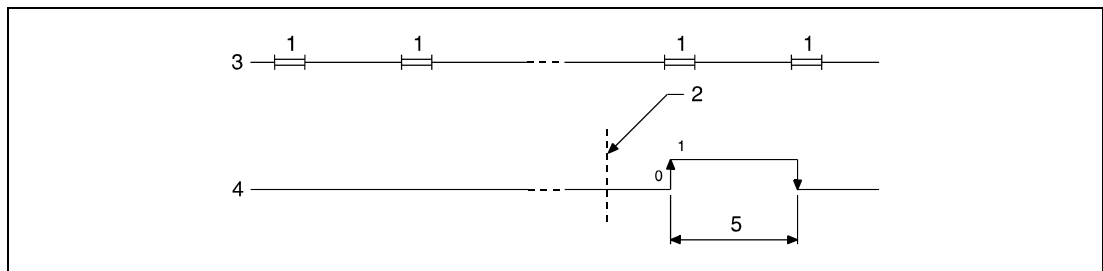
Numero canale	1	2	3	4	5	6	7	8
Flag canale di comunicazione	SB30	SB32	SB34	SB36	SB38	SB40	SB42	SB44

La figura seguente mostra le operazioni della stazione host durante l'esecuzione di una istruzione SWRITE:



- 1 Elaborazione END
- 2 Esecuzione istruzione SWRITE
- 3 Completamento operazione
- 4 Programma della stazione host
- 5 Istruzione SWRITE
- 6 Flag canale di comunicazione
- 7 Operando di completamento della stazione host (d2) attivato al termine dell'operazione
- 8 Indicazione stato completamento dell'operazione ((d2)+1)
- 9 Completamento trasmissione con errore
- 10 Completamento di una trasmissione senza errori
- 11 Una scansione

La figura seguente mostra le operazioni della stazione target durante l'esecuzione di una istruzione SWRITE:



- 1 Elaborazione END
- 2 Completamento operazione
- 3 Programma della stazione target
- 4 Operando di completamento della stazione target (d3) attivato al termine dell'operazione
- 5 Una scansione

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- I contenuti dei dati di controllo superano i campi ammessi (codice di errore 4100).
- La rete con il numero specificato da Jn non è collegata alla stazione (codice di errore 4102).
- Il modulo con l'indirizzo di I/O specificato da Un non è un modulo di rete (codice di errore 2111).

8.6.5 SEND

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	

Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	SM0	10	
s2	—	●	●	—	—	—	—	—			
d	●	●	●	—	—	—	—	—			

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
<div style="border: 1px solid black; padding: 5px; width: fit-content;">MELSEC</div> <p>JP.SEND Jn s1 s2 d</p> <p>GP.SEND Un s1 s2 d</p>		<p>SEND_JP_M Jn , s1 , s2 , d</p> <p>SEND_UP_M Un , s1 , s2 , d</p>

Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
Jn	Numero di rete della stazione host ● ¹	BIN 16-bit	ANY16
Un	Indirizzo intestazione I/O per unità di rete della stazione host. ● ²		
s1	Indirizzo del primo operando che memorizza i dati di controllo.	Indirizzo operando	Array [1..18] di ANY16
s2	Indirizzo del primo operando che contiene i dati da inviare.		ANY16
d	Operando attivato per una scansione al completamento dell'istruzione.	Bit	BOOL

NOTA

- ¹ Il numero di rete della stazione host può variare nel campo da 1 a 239. La rete di numero 254 viene configurata tramite parametrizzazione per consentire l'accesso alla stazione attiva da parte delle altre stazioni.
- ² L'indirizzo di intestazione di I/O dell'unità di rete per la stazione host deve essere nel campo da 0 a FEH. Notare, che il compilatore considera Un come valore esadecimale. Un numero decimale viene convertito automaticamente in valore esadecimale.

Panoramica degli operandi per i dati di controllo

Operando	Significato	Funzione	Valore Campo	Impostato da
(s1)+0 Array_s1[1]	Modo di esecuzione	Conferma del completamento trasmissione = imposta il bit 0 su 0 Nessuna conferma del completamento trasmissione = imposta il bit 0 su 1	0000H 0001H 0080H 0081H	Utente
	Modo completamento errore	Memorizzazione dati impostazione orologio al termine dell'elaborazione errore: - nessuna memorizzazione dati orologio, Bit 7 (b7) = 0 - memorizzazione dati orologio, Bit 7 (b7) = 1 (dati orologio a partire da (s1)+10 (Array_s1[11]))		
(s1)+1 Array_s1[2]	Stato completamento esecuzione istruzione	Stato al completamento dell'esecuzione dell'istruzione: 0 = nessun errore (completamento normale) < > 0 = codice di errore ● ³	—	Sistema
(s1)+2 Array_s1[3]	Canale usato dalla stazione host	Designazione canale host.	da 1 a 8	Utente
(s1)+3 Array_s1[4]	Canale usato dalla stazione target	Designazione stazione target.	da 1 a 8	Utente
(s1)+4 Array_s1[5]	Numero di rete della stazione target	Specifica il numero di rete della stazione target.	da 1 a 239 254 ● ⁴	Utente
(s1)+5 Array_s1[6]	Numero stazione target	Specifica il numero della stazione target.	Numero stazione: da 1 a 64 Indicazione gruppo: da 81H a 89H Tutte le stazioni della rete target: FFH	Utente
(s1)+6 Array_s1[7]	Dummy	Non utilizzato	—	—
(s1)+7 Array_s1[8]	Numero tentativi di trasmissione	Specifica il numero di tentativi per raggiungere il completamento dell'istruzione WRITE entro il tempo WDT memorizzato in (s1)+8 (Array_s1[9]). Attivo solo se il modo esecuzione in (s1)+0 Array_s1[1] è attivo (1).	da 1 a 15	Utente
	Numero tentativi di trasmissione eseguiti	Contiene il numero dei tentativi di trasmissione eseguiti.		Sistema
(s1)+8 Array_s1[9]	Impostazione tempo di trasmissione per WDT	Imposta il tempo di monitoraggio in secondi per le operazioni WRITE. Se l'operazione non viene terminata entro il tempo impostato, la trasmissione viene ripetuta per il numero di volte impostato in (s1)+7 (Array_s1[8]).	da 1 a 32767 0 = 10 s (fisso) Attivo solo se il modo esecuzione in (s1)+0 Array_s1[1] è attivo (1).	Utente
(s1)+9 Array_s1[10]	Lunghezza dati inviati	Specifica il numero di blocchi dati da scrivere.	da 1 a 480	Utente
(s1)+10 Array_s1[11]	Dummy	Non utilizzato	—	—
(s1)+11 Array_s1[12]	Segnalazione orologio impostato (attivato solo su errore)	Contiene lo stato di abilitazione/disabilitazione dati orologio impostato in (s1)+0 (Array_s1[1]): - Memorizzazione dati orologio disabilitata = 0 - Memorizzazione dati orologio abilitata = 1	—	Sistema

Operando	Significato	Funzione	Valore Campo	Impostato da
(s1)+12 Array_s1[13]	Dati orologio (attivato solo su errore)	Byte pesante = anno (da 0 a 99) Byte leggero = mese (da 1 a 12)	—	Sistema
(s1)+13 Array_s1[14]		Byte pesante = giorno (da 1 a 31) Byte leggero = ora (da 0 a 23)		
(s1)+14 Array_s1[15]		Byte pesante = minuto (da 0 a 59) Byte leggero = secondo (da 0 a 59)		
(s1)+15 Array_s1[16]		Byte pesante = 00h Byte leggero = giorno della settimana (da 0 a 6) (domenica = 0, sabato = 6)		
(s1)+16 Array_s1[17]	Numero della rete in cui si è verificato l'errore	Contiene il numero di rete della stazione in cui si è verificato l'errore. Il numero di rete è compreso fra 1 e 239.	—	Sistema
(s1)+17 Array_s1[18]	Numero della stazione in cui si è verificato l'errore	Contiene il numero della stazione in cui si è verificato l'errore. Il numero non viene memorizzato se lo stato di completamento dell'istruzione è "Channel in use" (canale in uso). Il numero di stazione è compreso fra 1 e 64.	—	Sistema

●³ Fare riferimento al manuale MELSECNET/10 per i sistemi di rete QnA per ulteriori dettagli.

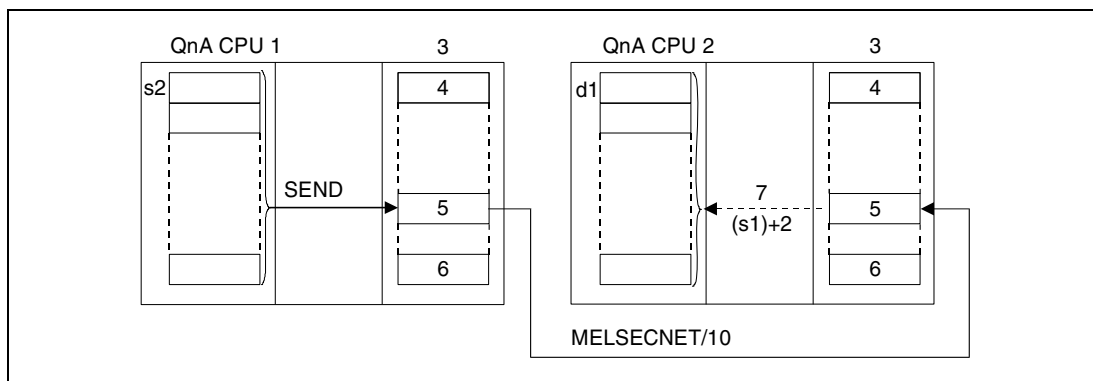
●⁴ Il numero di rete 254 viene specificato inserendo 254 in Jn.

Funzioni Invio dati ad altre stazioni

SEND Istruzione di invio

L'istruzione SEND invia i dati contenuti a partire da s2 da una stazione host ad una stazione collegata a MELSECNET/10. Il canale di trasferimento è specificato da (s1)+2. Il numero di stazione e di rete viene specificato nei dati di controllo.

Al termine dell'operazione di scrittura, viene attivato l'operando d2 della stazione target.



- 1 Stazione host
- 2 Stazione target
- 3 Modulo di rete
- 4 Canale 1
- 5 Canale n
- 6 Canale 8
- 7 L'operazione di lettura viene attivata dall'istruzione RECV

Tramite una stazione ripetitrice e l'impostazione di parametri di instradamento, è possibile anche accedere a stazioni in reti diverse.

Le istruzioni di comunicazione non possono essere eseguite contemporaneamente da più locazioni con accesso comune allo stesso canale. In caso di esecuzione simultanea dell'istruzione da due o più locazioni, un meccanismo di interblocco fra le due stazioni attive impedisce l'esecuzione di ulteriori istruzioni di comunicazione.

Lo stato di esecuzione e lo stato di completamento (normale, anomalo) dell'istruzione SEND può essere controllato tramite:

- il flag della direttiva di comunicazione (●⁵) del canale usato,
- l'operando di completamento (d) che viene attivato al termine dell'operazione,
- l'indicazione di stato del completamento operazione (completamento di trasmissione corretta o con errore) (d+1)

come segue:

Flag direttiva di comunicazione

Questo flag è attivato durante l'esecuzione dell'istruzione SEND. Il flag è azzerato con l'elaborazione dell'istruzione END alla fine della scansione di programma in cui viene completata l'operazione.

Operando di completamento stazione host

Questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione di scrittura viene completata. L'operando viene azzerato con la successiva istruzione END.

Indicazione stato completamento dell'operazione

Questo operando viene attivato in funzione del risultato di completamento dell'istruzione di scrittura.

Rimane a zero in caso di trasmissione normale (senza errori).

In caso di completamento con errore di una trasmissione, questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione SEND viene completata. L'operando viene azzerato con la successiva istruzione END.

Operando di completamento stazione target

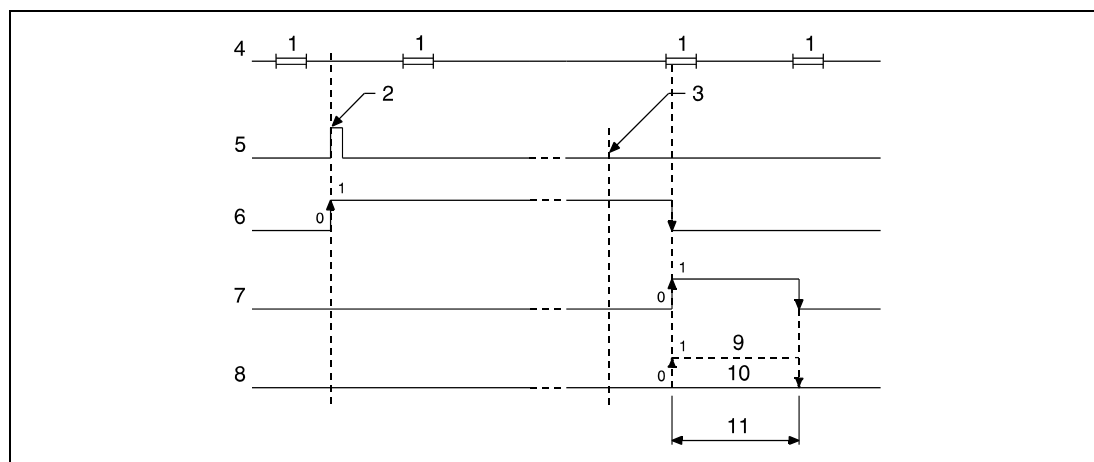
Questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione di lettura viene completata. L'operando viene azzerato con la successiva istruzione END.

NOTA

- ⁵ La tabella seguente assegna i numeri di canale secondo i flag di comunicazione del canale:

Numero canale	1	2	3	4	5	6	7	8
Flag canale di comunicazione	SB30	SB32	SB34	SB36	SB38	SB3A	SB3C	SB3E

La figura seguente mostra le operazioni della stazione host durante l'esecuzione di una istruzione SEND:



- 1 Elaborazione END
- 2 Esecuzione istruzione SEND
- 3 Completamento operazione
- 4 Programma della stazione host
- 5 Istruzione SEND
- 6 Flag canale di comunicazione
- 7 Operando di completamento della stazione host (d) attivato al termine dell'operazione
- 8 Indicazione stato completamento dell'operazione (d+1)
- 9 Completamento trasmissione con errore
- 10 Completamento di una trasmissione senza errori
- 11 Una scansione

Errori di esecuzione

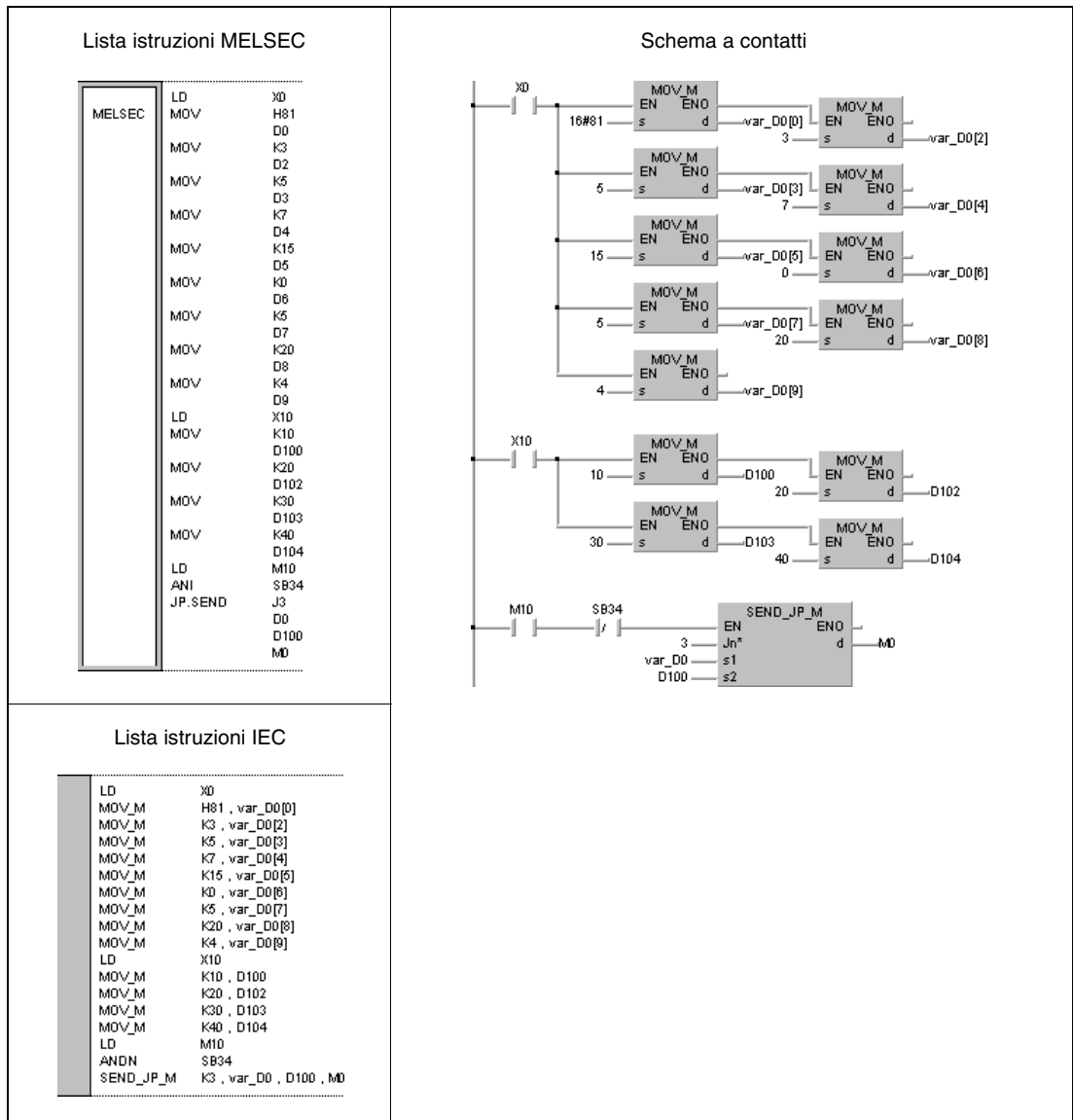
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- I contenuti dei dati di controllo superano i campi ammessi (codice di errore 4100).
- La rete con il numero specificato da Jn non è collegata alla stazione (codice di errore 4102).
- Il modulo con l'indirizzo di I/O specificato da Un non è un modulo di rete (codice di errore 2111).

Esempio di programma JP.SEND

Con il fronte positivo di M10, il programma seguente invia dalla stazione host ad una stazione target. L'esecuzione dell'istruzione SEND è interbloccata tramite il contatto normalmente aperto del flag SB34. La tabella seguente contiene ulteriori informazioni su stazione host, stazione target e istruzioni MOV utilizzate.

Operando/Istruzione	Significato/Funzione
Stazione host	–
Rete host	7
Canale host	3
Flag canale di comunicazione	SB34
Stazione target	15
Rete target	5
Canale target	5
1. Istruzione MOV	Imposta le condizioni di ingresso ed i dati orologio
2. Istruzione MOV	Imposta il canale della stazione HOST
3. Istruzione MOV	Imposta il canale della stazione target
4. Istruzione MOV	Specifica il numero di rete della stazione target.
5. Istruzione MOV	Specifica il numero della stazione target.
6. Istruzione MOV	–
7. Istruzione MOV	Imposta il numero di tentativi di trasmissione
8. Istruzione MOV	Imposta il tempo del WDT (20s)
9. Istruzione MOV	Imposta il numero di blocchi da inviare (4)
10. Istruzione MOV	Imposta i dati da inviare
11. Istruzione MOV	
12. Istruzione MOV	
13. Istruzione MOV	



NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

8.6.6 RECV

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	

Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	—	SM0	9	
d1	—	●	●	—	—	—	—	—			
d2	●	●	●	—	—	—	—	—			

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
<div style="border: 1px solid black; padding: 5px; width: fit-content;">MELSEC</div> <p>JP.RECV Jn s d1 d2</p> <p>GP.RECV Un s d1 d2</p>		<p>RECV_JP_M Jn, s, d1, d2</p> <p>RECV_UP_M Un, s, d1, d2</p>

Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
Jn	Numero di rete della stazione host ● ¹	BIN 16-bit	ANY16
Un	Indirizzo intestazione I/O per unità di rete della stazione host. ● ²		
s	Indirizzo del primo operando che memorizza i dati di controllo.	Indirizzo operando	Array [1..16] di ANY16
d1	Indirizzo del primo operando che contiene i dati da inviare.		ANY16
d2	Operando attivato per una scansione al completamento dell'istruzione.	Bit	BOOL

NOTA

- ¹ Il numero di rete della stazione host può variare nel campo da 1 a 239. La rete di numero 254 viene configurata tramite parametrizzazione per consentire l'accesso alla stazione attiva da parte delle altre stazioni.
- ² L'indirizzo di intestazione di I/O dell'unità di rete per la stazione host deve essere nel campo da 0 a FEH. Notare, che il compilatore considera Un come valore esadecimale. Un numero decimale viene convertito automaticamente in valore esadecimale.

Panoramica degli operandi per i dati di controllo

Operando	Significato	Funzione	Valore Campo	Impostato da
(s1)+0 Array_s1[1]	Modo di esecuzione	Attesa dati (in questo modo si attendono ripetutamente i dati per un tempo impostato) (bit 0 (b0) = 0, fisso)	0000H 0080H	Utente
	Modo completamento errore	Memorizzazione dati impostazione orologio al termine dell'elaborazione errore: - nessuna memorizzazione dati orologio, Bit 7 (b7) = 0 - memorizzazione dati orologio, Bit 7 (b7) = 1 (dati orologio a partire da (s1)+10 (Array_s1[11]))		
(s1)+1 Array_s1[2]	Stato completamento esecuzione istruzione	Stato al completamente dell'esecuzione dell'istruzione: 0 = nessun errore (completamento normale) < > 0 = codice di errore ● ³	—	Sistema
(s1)+2 Array_s1[3]	Canale usato dalla stazione host	Imposta il canale che contiene i dati da ricevere.	da 1 a 8	Utente
(s1)+3 Array_s1[4]	Canale usato dalla stazione target	Memorizza il canale per la stazione mittente.	da 1 a 8	Sistema
(s1)+4 Array_s1[5]	Numero di rete della stazione target	Memorizza il numero di rete per la stazione mittente.	da 1 a 239	Sistema
(s1)+5 Array_s1[6]	Numero stazione target	Memorizza il numero di stazione per la stazione mittente.	da 1 a 64	Sistema
(s1)+6 Array_s1[7]	Dummy	Non utilizzato	—	—
(s1)+7 Array_s1[8]	Dummy	Non utilizzato	—	—
(s1)+8 Array_s1[9]	Impostazione tempo di trasmissione per WDT	Imposta il tempo di monitoraggio in secondi per l'operazione.	da 1 a 32767 0 = 10 s (fisso) Attivo solo se attivo il modo esecuzione (1).	Utente
(s1)+9 Array_s1[10]	Lunghezza dati inviati	Contiene il numero di blocchi dati ricevuti.	da 1 a 480	Sistema
(s1)+10 Array_s1[11]	Dummy	Non utilizzato	—	—
(s1)+11 Array_s1[12]	Segnalazione orologio impostato (attivato solo su errore)	Contiene lo stato di abilitazione/disabilitazione dati orologio impostato in (s1)+0 (Array_s1[1]): - Memorizzazione dati orologio disabilitata = 0 - Memorizzazione dati orologio abilitata = 1	—	Sistema
(s1)+12 Array_s1[13]	Dati orologio (attivato solo su errore)	Byte pesante = anno (da 0 a 99) Byte leggero = mese (da 1 a 12)	—	Sistema
(s1)+13 Array_s1[14]		Byte pesante = giorno (da 1 a 31) Byte leggero = ora (da 0 a 23)		
(s1)+14 Array_s1[15]		Byte pesante = minuto (da 0 a 59) Byte leggero = secondo (da 0 a 59)		
(s1)+15 Array_s1[16]		Byte pesante = 00H Byte leggero = giorno della settimana (da 0 a 6) (domenica = 0, sabato = 6)		

●³ Fare riferimento al manuale MELSECNET/10 per i sistemi di rete QnA per ulteriori dettagli.

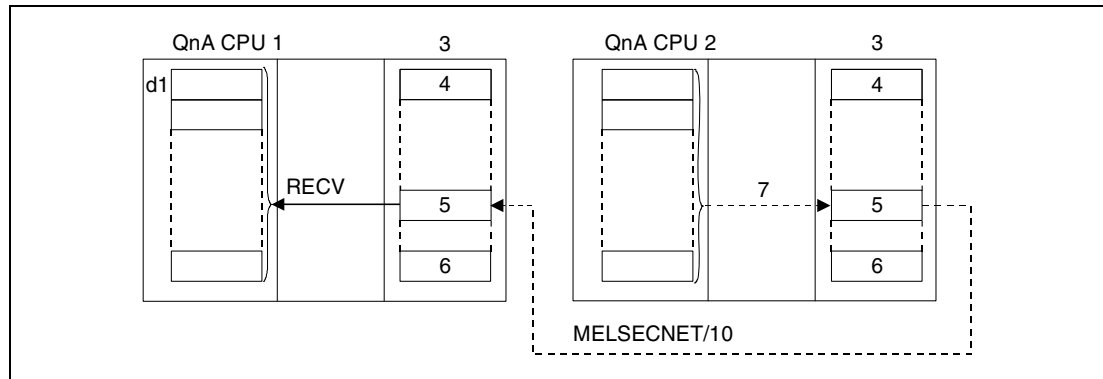
Funzioni

Ricezione dati inviati da altre stazioni

RECV Istruzione di ricezione

L'istruzione RECV riceve i dati inviati da una stazione collegata a MELSECNET/10 tramite l'istruzione SEND. Il numero di stazione e di rete viene specificato nei dati di controllo. I dati vengono memorizzati a partire da d1.

Al termine dell'operazione viene attivato l'operando specificato da d2.



¹ Stazione host

² Stazione target

³ Modulo di rete

⁴ Canale 1

⁵ Canale n

⁶ Canale 8

⁷ L'operazione di scrittura viene attivata dall'istruzione SEND.

Tramite una stazione ripetitrice e l'impostazione di parametri di instradamento, è possibile anche accedere a stazioni in reti diverse.

Le istruzioni di comunicazione non possono essere eseguite contemporaneamente da più locazioni con accesso comune allo stesso canale. In caso di esecuzione simultanea dell'istruzione da due o più locazioni, un meccanismo di interblocco fra le due stazioni attive impedisce l'esecuzione di ulteriori istruzioni di comunicazione.

Lo stato di esecuzione e lo stato di completamento (normale, anomalo) dell'istruzione RECV può essere controllato tramite:

- il flag della direttiva di comunicazione (●⁴) del canale usato,
- l'operando di completamento (d2) che viene attivato al termine dell'operazione,
- l'indicazione di stato del completamento operazione (completamento di trasmissione corretta o con errore) ((d2)+1)

come segue:

Flag direttiva di comunicazione

Questo flag è attivato durante l'esecuzione dell'istruzione RECV. Il flag è azzerato con l'elaborazione dell'istruzione END alla fine della scansione di programma in cui viene completata l'operazione.

Operando di completamento stazione host

Questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione di scrittura viene completata. L'operando viene azzerato con la successiva istruzione END.

Indicazione stato completamento dell'operazione

Questo operando viene attivato in funzione del risultato di completamento dell'istruzione.

Rimane a zero in caso di trasmissione normale (senza errori).

In caso di completamento con errore di una trasmissione, questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione RECV viene completata. L'operando viene azzerato con la successiva istruzione END.

Operando di completamento stazione target

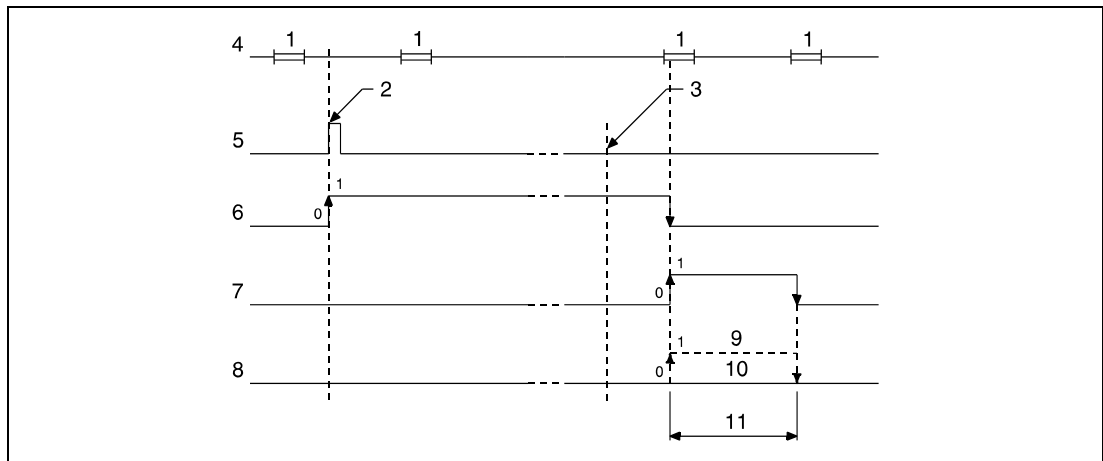
Questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione di lettura viene completata. L'operando viene azzerato con la successiva istruzione END.

NOTA

●⁴ La tabella seguente assegna i numeri di canale secondo i flag di comunicazione del canale:

Numero canale	1	2	3	4	5	6	7	8
Flag canale di comunicazione	SB30	SB32	SB34	SB36	SB38	SB3A	SB3C	SB3E

La figura seguente mostra le operazioni della stazione host durante l'esecuzione di una istruzione RECV:



- 1 Elaborazione END
- 2 Esecuzione istruzione RECV
- 3 Completamento operazione
- 4 Programma della stazione host
- 5 Istruzione RECV
- 6 Flag canale di comunicazione
- 7 Operando di completamento della stazione host (d2) attivato al termine dell'operazione
- 8 Indicazione stato completamento dell'operazione ((d2)+1)
- 9 Completamento trasmissione con errore
- 10 Completamento di una trasmissione senza errori
- 11 Una scansione

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

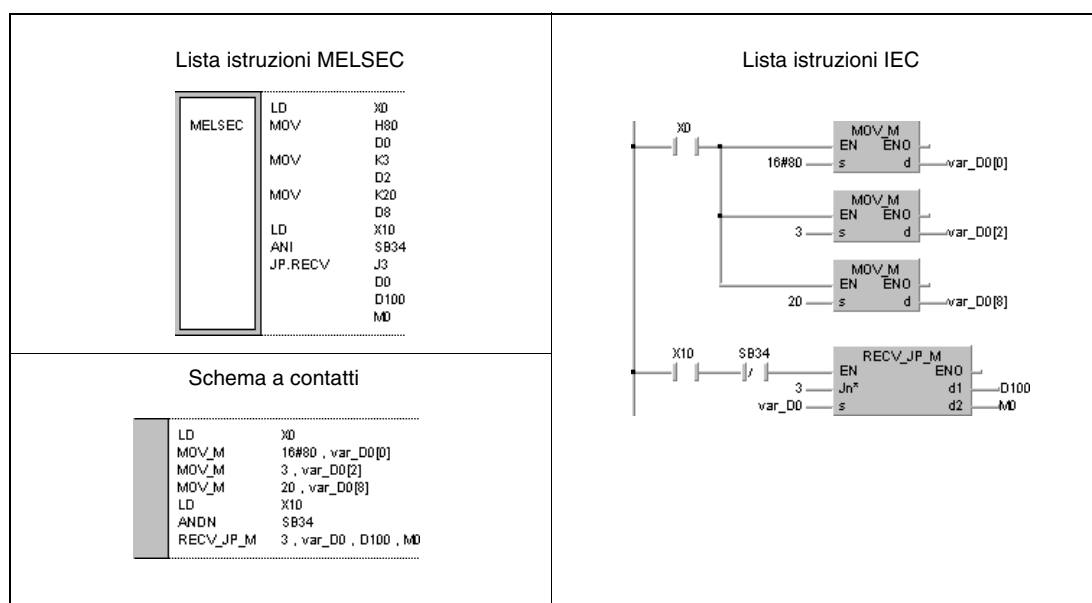
- I contenuti dei dati di controllo superano i campi ammessi (codice di errore 4100).
- La rete con il numero specificato da Jn non è collegata alla stazione (codice di errore 4102).
- Il modulo con l'indirizzo di I/O specificato da Un non è un modulo di rete (codice di errore 2111).

Esempio di programma

JP.RECV

Con il fronte positivo di X10, il programma seguente legge i dati inviati da una stazione con l'istruzione SEND. L'esecuzione dell'istruzione RECV è interbloccata tramite il contatto normalmente aperto del flag SB34. La tabella seguente contiene ulteriori informazioni su stazione host, stazione mittente e istruzioni MOV utilizzate.

Operando/Istruzione	Significato/Funzione
Stazione host	–
Rete host	–
Canale host	3
Flag canale di comunicazione	SB34
Stazione mittente	–
Rete della stazione mittente	3
Canale della stazione mittente	3
1. Istruzione MOV	Imposta i dati orologio
2. Istruzione MOV	Imposta il canale della stazione HOST
3. Istruzione MOV	Imposta il tempo del WDT (20s)

**NOTA**

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

8.6.7 REQ

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	

Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	SM0	10	
s2	—	●	●	—	—	—	—	—			
d1	—	●	●	—	—	—	—	—			
d2	●	●	●	—	—	—	—	—			

GX IEC Developer

<p>Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; margin: 5px;"> <p>MELSEC</p> </div> <p>JP.REQ Jn s1 s2 d1 d2</p> <p>GP.REQ Un s1 s2 d1 d2</p>	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p> <div style="border: 1px solid black; padding: 5px; margin: 5px;"> <p>REQ_JP_M Jn , s1 , s2 , d1 , d2</p> <p>REQ_UP_M Un , s1 , s2 , d1 , d2</p> </div>
---	--------------------------	--

Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
Jn	Numero di rete della stazione host ● ¹	BIN 16-bit	ANY16
Un	Indirizzo intestazione I/O per unità di rete della stazione host. ● ²		
s1	Indirizzo del primo operando che memorizza i dati di controllo.	Indirizzo operando	Array [1..18] di ANY16
s2	Indirizzo del primo operando che memorizza i dati richiesti.		Array [1..7] di ANY16
d1	Indirizzo del primo operando che memorizza i dati della risposta.		Array [1..4] di ANY16
d2	Operando attivato per una scansione al completamento dell'istruzione.	Bit	BOOL

NOTA

- ¹ Il numero di rete della stazione host può variare nel campo da 1 a 239. La rete di numero 254 viene configurata tramite parametrizzazione per consentire l'accesso alla stazione attiva da parte delle altre stazioni.
- ² L'indirizzo di intestazione di I/O dell'unità di rete per la stazione host deve essere nel campo da 0 a FE_H. Notare, che il compilatore considera Un come valore esadecimale. Un numero decimale viene convertito automaticamente in valore esadecimale.

L'istruzione REQ può essere eseguita solo se la stazione target è una CPU QnA.

Con una ACPU su MELSECNET/10, l'istruzione REQ non può essere usata.

Solo i numeri di stazione delle CPU QnA sono numeri validi per stazioni target.

Panoramica degli operandi per i dati di controllo

Operando	Significato	Funzione	Valore Campo	Impostato da
(s1)+0 Array_s1[1]	Modo di esecuzione	Conferma completamento trasmissione attivo (Bit 0 (b0) = 1, fisso)	0001H 0081H	Utente
	Modo completamento errore	Memorizzazione dati impostazione orologio al termine dell'elaborazione errore: - nessuna memorizzazione dati orologio, Bit 7 (b7) = 0 - memorizzazione dati orologio, Bit 7 (b7) = 1 (dati orologio a partire da (s1)+10 (Array_s1[11]))		
(s1)+1 Array_s1[2]	Stato completamento esecuzione istruzione	Stato al completamento dell'esecuzione dell'istruzione: 0 = nessun errore (completamento normale) < > 0 = codice di errore ● ¹	—	Sistema
(s1)+2 Array_s1[3]	Canale usato dalla stazione host	Imposta il canale usato dalla stazione host	da 1 a 8	Utente
(s1)+3 Array_s1[4]	Dummy	Non utilizzato	—	—
(s1)+4 Array_s1[5]	Numero di rete della stazione target.	Specifica il numero di rete della stazione da cui leggere.	da 1 a 239 254 ● ²	Utente
(s1)+5 Array_s1[6]	Numero della stazione target.	Imposta il numero di rete della stazione target.	da 1 a 64	Utente
(s1)+6 Array_s1[7]	Dummy	Non utilizzato	—	—
(s1)+7 Array_s1[8]	Numero tentativi di trasmissione	Specifica il numero di tentativi per raggiungere il completamento dell'istruzione REQ entro il tempo WDT memorizzato in (s1)+8 (Array_s1[9]).	da 1 a 15	Utente
	Numero tentativi di trasmissione eseguiti	Contiene il numero dei tentativi di trasmissione eseguiti.	—	Sistema
(s1)+8 Array_s1[9]	Impostazione tempo di trasmissione per WDT	Imposta il tempo di monitoraggio in secondi per l'operazione. Se l'operazione non viene terminata entro il tempo impostato, la trasmissione viene ripetuta per il numero di volte impostato in (s1)+7 (Array_s1[8]).	da 1 a 32767 0 = 10 secondi (fisso)	Utente
(s1)+9 Array_s1[10]	Lunghezza dei dati richiesti.	Imposta la lunghezza dei dati richiesti Lettura dati orologio = 2 Scrittura dati orologio = 7 Durante RUN/STOP remoto = 4	2, 7, 4	Utente
(s1)+10 Array_s1[11]	Lunghezza dei dati risposta	Imposta la lunghezza dei dati della risposta Lettura dati orologio = 2	0, 4	Utente
(s1)+11 Array_s1[12]	Segnalazione orologio impostato (attivato solo su errore)	Contiene lo stato di abilitazione/disabilitazione dati orologio impostato in (s1)+0 (Array_s1[1]): - Memorizzazione dati orologio disabilitata = 0 - Memorizzazione dati orologio abilitata = 1	—	Sistema
(s1)+12 Array_s1[13]	Dati orologio (attivato solo su errore)	Byte pesante = anno (da 0 a 99) Byte leggero = mese (da 1 a 12)	—	Sistema
(s1)+13 Array_s1[14]		Byte pesante = giorno (da 1 a 31) Byte leggero = ora (da 0 a 23)		
(s1)+14 Array_s1[15]		Byte pesante = minuto (da 0 a 59) Byte leggero = secondo (da 0 a 59)		
(s1)+15 Array_s1[16]		Byte pesante = 00H Byte leggero = giorno della settimana (da 0 a 6) (domenica = 0, sabato = 6)		

Operando	Significato	Funzione	Valore Campo	Impostato da
(s1)+16 Array_s1[17]	Numero della rete in cui si è verificato l'errore	Contiene il numero di rete della stazione in cui si è verificato l'errore. Il numero di rete varia da 1 a 239. Il numero non viene memorizzato se lo stato di completamento dell'istruzione è "Channel in Use (F7C1H)" (canale in uso).	—	Sistema
(s1)+17 Array_s1[18]	Numero della stazione in cui si è verificato l'errore	Contiene il numero della stazione in cui si è verificato l'errore. Il numero di rete varia da 1 a 64. Il numero non viene memorizzato se lo stato di completamento dell'istruzione è "Channel in Use (F7C1H)" (canale in uso).	—	Sistema

●¹ Fare riferimento al manuale MELSECNET/10 per i sistemi di rete QnA per ulteriori dettagli.

●² Il numero di rete 254 viene specificato inserendo 254 in Jn.

Richiesta/risposta dati durante operazioni di scrittura/lettura di dati orologio Richiesta dati

Operando	Significato	Funzione	Letture Dati orologio	Scrittura Dati orologio										
(s2)+0 Array_s2[1]	Tipo richiesta	0001H = Lettura dati orologio 0011H = Scrittura dati orologio	●	●										
(s2)+1 Array_s2[2]	Tipo richiesta sotto-programma	0002H = Lettura dati orologio 0001H = Scrittura dati orologio	●	●										
(s2)+2 Array_s2[3]	Maschera aggiornamento	Indica gli elementi dei dati orologio da (s2)+3 (Array_s2[4]) a (s2)+6 (Array_s2[7]) da aggiornare. Se l'operando è attivo (1) l'elemento dei dati orologio viene aggiornato. b15 ----- b7 b6 b5 b4 b3 b2 b1 b0 <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 10px;">0</td> <td style="width: 10px;"></td> <td style="width: 10px;">0</td> <td style="width: 10px;">W</td> <td style="width: 10px;">Sec</td> <td style="width: 10px;">Min</td> <td style="width: 10px;">H</td> <td style="width: 10px;">D</td> <td style="width: 10px;">M</td> <td style="width: 10px;">Y</td> </tr> </table>	0		0	W	Sec	Min	H	D	M	Y		●
0		0	W	Sec	Min	H	D	M	Y					
(s2)+3 Array_s2[4]	Mese e anno da aggiornare	Mese e anno memorizzati in codice BCD (ultime due cifre). b15 ----- b8 b7 ----- b0 <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 10px;">M</td> <td style="width: 10px;">(01H - 12H)</td> <td style="width: 10px;">Y</td> <td style="width: 10px;">(00H - 99H)</td> </tr> </table>	M	(01H - 12H)	Y	(00H - 99H)		●						
M	(01H - 12H)	Y	(00H - 99H)											
(s2)+4 Array_s2[5]	Ora e giorno da aggiornare	Ora e giorno memorizzati in codice BCD. b15 ----- b8 b7 ----- b0 <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 10px;">H</td> <td style="width: 10px;">(00H - 23H)</td> <td style="width: 10px;">D</td> <td style="width: 10px;">(01H - 31H)</td> </tr> </table>	H	(00H - 23H)	D	(01H - 31H)		●						
H	(00H - 23H)	D	(01H - 31H)											
(s2)+5 Array_s2[6]	Minuto e secondo da aggiornare	Secondo e minuto memorizzati in codice BCD. b15 ----- b8 b7 ----- b0 <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 10px;">Sec</td> <td style="width: 10px;">(00H - 59H)</td> <td style="width: 10px;">Min</td> <td style="width: 10px;">(00H - 59H)</td> </tr> </table>	Sec	(00H - 59H)	Min	(00H - 59H)		●						
Sec	(00H - 59H)	Min	(00H - 59H)											
(s2)+6 Array_s2[7]	Giorno della settimana da aggiornare	Giorno della settimana memorizzato in codice BCD (00H = domenica, 06H = sabato). b15 ----- b8 b7 ----- b0 <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="width: 10px;">00H</td> <td style="width: 10px;">W</td> <td style="width: 10px;">(00H - 06H)</td> </tr> </table>	00H	W	(00H - 06H)		●							
00H	W	(00H - 06H)												

M = mese
 Y = anno
 H = ora
 D = giorno
 Sec = secondo
 Min = minuto
 W = Giorno della settimana

Dati risposta

Operando	Significato	Funzione	Letture Dati orolo- gio	Scrittura Dati orolo- gio
(d1)+0 Array-d1[1]	Mese e anno in lettura	Mese e anno memorizzati in codice BCD (ultime due cifre). I b15 ----- b8 b7 ----- b0 M (01H - 12H) Y (00H - 99H)	●	
(d1)+1 Array_d1[2]	Ora e giorno in lettura	Ora e giorno memorizzati in codice BCD. b15 ----- b8 b7 ----- b0 H (00H - 23H) D (01H - 31H)	●	
(d1)+2 Array_d1[3]	Minuto e secondo in lettura	Secondo e minuto memorizzati in codice BCD. b15 ----- b8 b7 ----- b0 Sec (00H - 59H) Min (00H - 59H)	●	
(d1)+3 Array_d1[4]	Giorno della set- timana in lettura	Giorno della settimana memorizzato in codice BCD (00H = domenica, 06H = sabato). b15 ----- b8 b7 ----- b0 00H W (00H - 06H)	●	

M = mese
Y = anno
H = ora
D = giorno
Sec = secondo
Min = minuto
W = Giorno della settimana

NOTA

Le operazioni di scrittura/lettura vengono disabilitate se la funzione "Memory Protect" (protezione memoria) è attiva sulla CPU della stazione target (commutatore di sistema 1, SW5 (QnA, Q4AR), SW1 (QnAS) impostati ON).

Richiesta dati durante operazione RUN/STOP su stazione remota**Richiesta dati**

Operando	Significato	Funzione	RUN	STOP
(s2)+0 Array-s2[1]	Tipo richiesta	0010H	●	●
(s2)+1 Array_s2[2]	Tipo richiesta sotto-programma	0001H = operazione RUN su stazione remota 0002H = operazione STOP su stazione remota	●	●
(s2)+2 Array_s2[3]	Modo	Impostazione RUN forzato su stazione remota: 0001H = RUN non forzato 0003H = RUN forzato (impostato durante STOP remoto) Se la stazione che sta eseguendo una operazione di STOP su una stazione remota non può eseguire una operazione RUN, l'operazione di RUN remoto può essere eseguita da una stazione diversa.	●	●
(s2)+3 Array_s2[4]	Modo cancellazione	Imposta lo stato di memoria della CPU durante l'esecuzione dell'operazione RUN su una stazione remota: 0000H = Nessuna cancellazione (impostato durante STOP remoto) 0001H = Cancellazione (tranne zone retentive) 0002H = Cancellazione (comprese zone retentive)	●	●

NOTA

La funzione RUN/STOP può essere eseguita solo se il commutatore RUN/STOP della CPU sulla stazione target è impostato su RUN.

Le operazioni di scrittura/lettura vengono disabilitate se la funzione "Memory Protect" (protezione memoria) è attiva sulla CPU della stazione target (commutatore di sistema 1, SW5 (QnA, Q4AR), SW1 (QnAS) impostati ON).

Se la stazione target è già stata posta in modo STOP/PAUSE da un'altra stazione, l'operazione RUN può essere forzata solo se il modo in (s2)+2 è impostato su "do not force RUN (0001H)" (RUN non forzato).

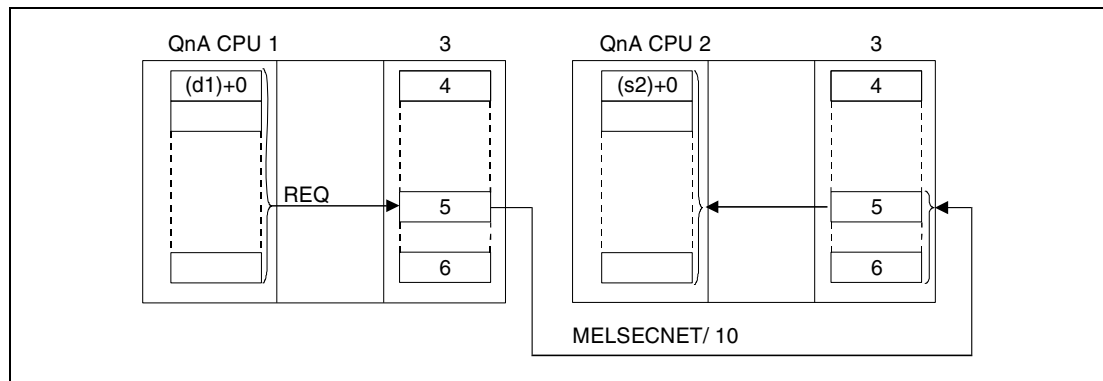
Se la CPU QnA della stazione target che esegue l'operazione RUN/STOP viene resettata, le informazioni relative all'operazione RUN/STOP sulla stazione target vengono perse.

Funzioni **Richiesta dati da altre stazioni**

REQ Istruzione di richiesta

L'istruzione REQ trasferisce i dati richiesti contenuti a partire da $(d1)+0$ (Array `_d1[1]`) da una stazione collegata a MELSECNET/10. Il numero di stazione e di rete viene specificato nei dati di controllo. I dati vengono memorizzati a partire da $(s2)+0$ (Array `_s2[1]`).

Al termine dell'operazione viene attivato l'operando specificato da `d2`.



- ¹ Stazione host
- ² Stazione target
- ³ Modulo di rete
- ⁴ Canale 1
- ⁵ Canale n
- ⁶ Canale 8

Tramite una stazione ripetitrice e l'impostazione di parametri di instradamento, è possibile anche accedere a stazioni in reti diverse.

Le istruzioni di comunicazione non possono essere eseguite contemporaneamente da più locazioni con accesso comune allo stesso canale. In caso di esecuzione simultanea dell'istruzione da due o più locazioni, un meccanismo di interblocco fra le due stazioni attive impedisce l'esecuzione di ulteriori istruzioni di comunicazione.

Lo stato di esecuzione e lo stato di completamento (normale, anomalo) dell'istruzione REQ può essere controllato tramite:

- il flag della direttiva di comunicazione (●³) del canale usato,
- l'operando di completamento (`d2`) che viene attivato al termine dell'operazione,
- l'indicazione di stato del completamento operazione (completamento di trasmissione corretta o con errore) ($(d2)+1$)

come segue:

Flag direttiva di comunicazione

Questo flag è attivato durante l'esecuzione dell'istruzione REQ. Il flag è azzerato con l'elaborazione dell'istruzione END alla fine della scansione di programma in cui viene completata l'operazione.

Operando di completamento stazione host

Questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione di scrittura viene completata. L'operando viene azzerato con la successiva istruzione END.

Indicazione stato completamento dell'operazione

Questo operando viene attivato in funzione del risultato di completamento dell'istruzione.

Rimane a zero in caso di trasmissione normale (senza errori).

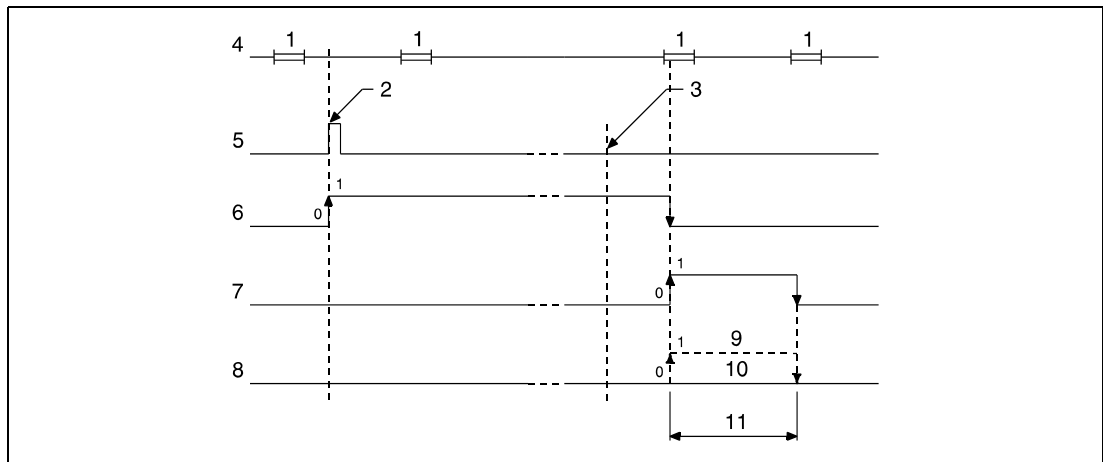
In caso di completamento con errore di una trasmissione, questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione REQ viene completata. L'operando viene azzerato con la successiva istruzione END.

NOTA

- ³ La tabella seguente assegna i numeri di canale secondo i flag di comunicazione del canale:

Numero canale	1	2	3	4	5	6	7	8
Flag canale di comunicazione	SB30	SB32	SB34	SB36	SB38	SB3A	SB3C	SB3E

La figura seguente mostra le operazioni della stazione host durante l'esecuzione di una istruzione REQ:



- 1 Elaborazione END
- 2 Esecuzione istruzione REQ
- 3 Completamento operazione
- 4 Programma della stazione host
- 5 Istruzione REQ
- 6 Flag canale di comunicazione
- 7 Operando di completamento della stazione host (d2) attivato al termine dell'operazione
- 8 Indicazione stato completamento dell'operazione ((d2)+1)
- 9 Completamento trasmissione con errore
- 10 Completamento di una trasmissione senza errori
- 11 Una scansione

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

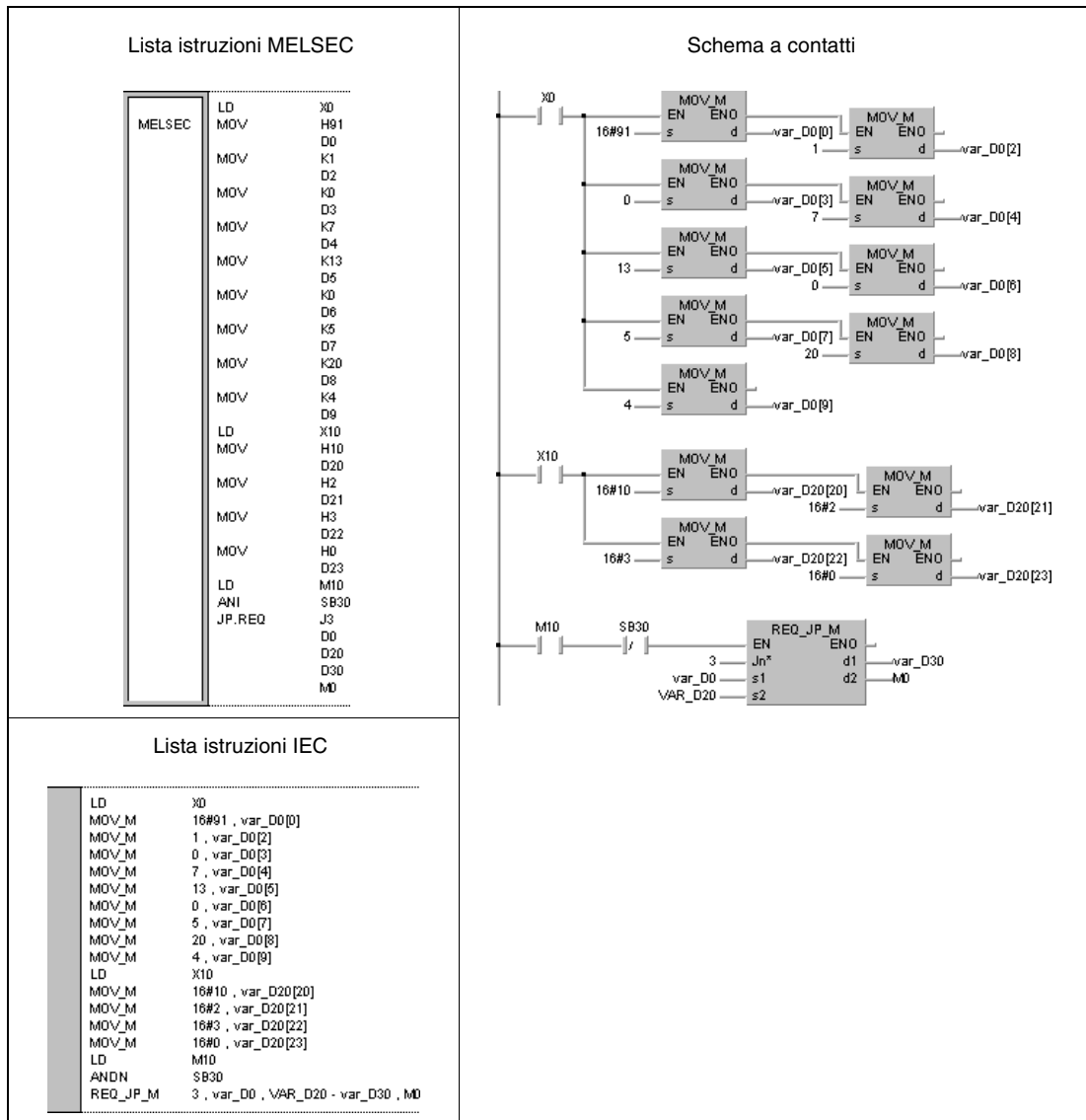
- I contenuti dei dati di controllo superano i campi ammessi (codice di errore 4100).
- La rete con il numero specificato da Jn non è collegata alla stazione (codice di errore 4102).
- Il modulo con l'indirizzo di I/O specificato da Un non è un modulo di rete (codice di errore 2111).

Esempio di programma

JP.REQ

Con il fronte di salita di X10, il programma seguente esegue una operazione di STOP su una stazione target. L'esecuzione dell'istruzione REQ è interbloccata tramite il contatto normalmente aperto del flag SB30. La tabella seguente contiene ulteriori informazioni su stazione host, stazione mittente e istruzioni MOV utilizzate.

Operando/Istruzione	Significato/Funzione
Stazione host	–
Rete host	–
Canale host	1
Flag canale di comunicazione	SB30
Stazione target	13
Rete target	7
Canale target	–
1. Istruzione MOV	Imposta i dati orologio
2. Istruzione MOV	Imposta il canale della stazione HOST
3. Istruzione MOV	–
4. Istruzione MOV	Specifica il numero di rete della stazione target.
5. Istruzione MOV	Specifica il numero della stazione target.
6. Istruzione MOV	–
7. Istruzione MOV	Imposta il numero di tentativi di trasmissione
8. Istruzione MOV	Imposta il tempo del WDT (20s)
9. Istruzione MOV	Imposta il numero di blocchi da inviare (4)
10. Istruzione MOV	Imposta il tipo di richiesta
11. Istruzione MOV	Imposta il tipo di richiesta del sottoprogramma
12. Istruzione MOV	Imposta il modo
13. Istruzione MOV	Imposta il modo cancellazione

**NOTA**

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

8.6.8 ZNFR

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	

Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	—	9	
s2	—	● ¹	—	—	—	—	—	—			
d	●	—	—	—	—	—	—	—			

¹ Solo registri di comunicazione

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
<div style="border: 1px solid black; padding: 5px; width: fit-content;">MELSEC</div> <p>JP.ZNFR Jn s1 s2 d</p> <p>GP.ZNFR Un s1 s2 d</p>		<p>ZNFR_JP_M Jn, s1, s2, d</p> <p>ZNFR_UP_M Un, s1, s2, d</p>

Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
Jn	Numero di rete della stazione host ● ¹	BIN 16-bit	ANY16
Un	Indirizzo intestazione I/O per unità di rete della stazione host. ● ²		
s1	Indirizzo del primo operando che memorizza i dati di controllo.	Indirizzo operando	Array [1..15] di ANY16
s2	Indirizzo del primo registro di comunicazione (W) della stazione host che contiene il dato letto.		ANY16
d	Operando attivato per una scansione al completamento dell'istruzione.	Bit	BOOL

NOTA

- ¹ Il numero di rete della stazione host può variare nel campo da 1 a 239. La rete di numero 254 viene configurata tramite parametrizzazione per consentire l'accesso alla stazione attiva da parte delle altre stazioni.
- ² L'indirizzo di intestazione di I/O dell'unità di rete per la stazione host deve essere nel campo da 0 a FEH. Notare, che il compilatore considera Un come valore esadecimale. Un numero decimale viene convertito automaticamente in valore esadecimale.

Panoramica degli operandi per i dati di controllo

Operando	Significato	Funzione	Valore Campo	Impostato da
(s1)+0 Array_s1[1]	Modo di esecuzione	Conferma completamento trasmissione attivo (Bit 0 (b0) = 1, fisso)	0001H 0081H	Utente
	Modo completamento errore	Memorizzazione dati impostazione orologio al termine dell'elaborazione errore: - nessuna memorizzazione dati orologio, Bit 7 (b7) = 0 - memorizzazione dati orologio, Bit 7 (b7) = 1 (dati orologio a partire da (s1)+11 (Array_s1[12]))		
(s1)+1 Array_s1[2]	Stato completamento esecuzione istruzione	Stato al completamento dell'esecuzione dell'istruzione: 0 = nessun errore (completamento normale) < > 0 = codice di errore ● ³	—	Sistema
(s1)+2 Array_s1[3]	Dummy	Non utilizzato	—	—
(s1)+3 Array_s1[4]	Indirizzo buffer di memoria	Imposta il primo indirizzo del buffer di memoria.	● ⁴	Utente
(s1)+4 Array_s1[5]	Dummy	Non utilizzato	—	—
(s1)+5 Array_s1[6]	Numero della stazione target.	Imposta il numero della stazione di I/O remoto che legge i dati.	da 1 a 64	Utente
(s1)+6 Array_s1[7]	Posizione del modulo funzione speciale	Imposta la posizione del modulo funzione speciale nella serie di moduli funzione speciali installati sulla stazione target.	—	Utente
(s1)+7 Array_s1[8]	Dummy	Non utilizzato	—	—
(s1)+8 Array_s1[9]	Dummy	Non utilizzato	—	—
(s1)+9 Array_s1[10]	Lunghezza dei dati	Specifica il numero di dati da leggere.	da 1 a 256	Utente
(s1)+10 Array_s1[11]	Segnalazione orologio impostato (attivato solo su errore)	Contiene lo stato di abilitazione/disabilitazione dati orologio impostato in (s1)+0 (Array_s1[1]): - Memorizzazione dati orologio disabilitata = 0 - Memorizzazione dati orologio abilitata = 1	—	Sistema
(s1)+11 Array_s1[12]	Dati orologio (attivato solo su errore)	Byte pesante = anno (da 0 a 99) Byte leggero = mese (da 1 a 12)	—	Sistema
(s1)+12 Array_s1[13]		Byte pesante = giorno (da 1 a 31) Byte leggero = ora (da 0 a 23)		
(s1)+13 Array_s1[14]		Byte pesante = minuto (da 0 a 59) Byte leggero = secondo (da 0 a 59)		
(s1)+14 Array_s1[15]		Byte pesante = 00H Byte leggero = giorno della settimana (da 0 a 6) (domenica = 0, sabato = 6)		

●³ Fare riferimento al manuale MELSECNET/10 per i sistemi di rete QnA per ulteriori dettagli.

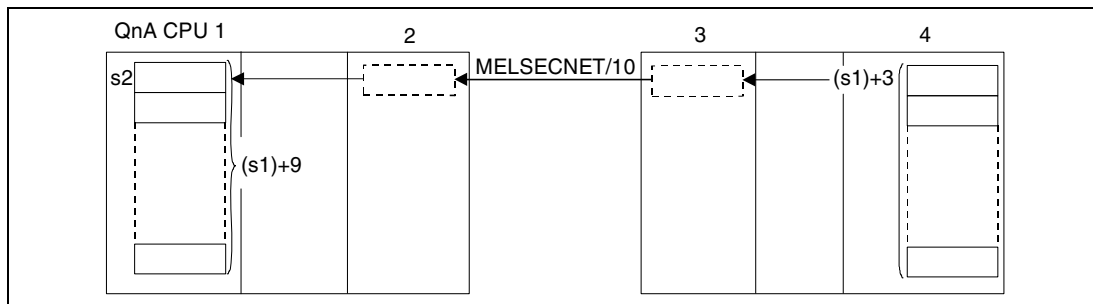
●⁴ Per ulteriori dettagli, fare riferimento al manuale relativo al modulo funzione speciale corrispondente.

Funzioni **Letture dati da moduli funzione speciali in stazioni I/O remote**

ZNFR Istruzione di lettura

L'istruzione ZNFR legge i dati contenuti nel buffer di memoria di un modulo funzione speciale all'interno di una stazione di I/O remoto collegata a MELSECNET/10. La stazione di I/O remoto è specificata nei dati di controllo. I dati letti dal modulo vengono memorizzati a partire da $s2$ nella stazione host.

Al termine dell'operazione viene attivato l'operando specificato da d .



- 1 Stazione host / stazione master
- 2 Modulo di rete (stazione host / stazione master)
- 3 Stazione I/O remoti (stazione target)
- 4 Modulo funzione speciale (stazione target / stazione I/O remoto)

L'operazione di lettura di una stazione di I/O remoto può essere eseguita tramite un modulo di rete collegato alla stessa rete della stazione di I/O remoto collegata a MELSECNET/10.

L'istruzione ZNFR non può essere eseguita contemporaneamente da più di una locazione dallo stesso modulo funzione speciale. In caso di esecuzione simultanea dell'istruzione da due o più locazioni, un meccanismo di interblocco fra le due stazioni attive impedisce l'esecuzione di ulteriori istruzioni ZNFR.

Il segnale di interblocco inviato durante l'esecuzione dell'istruzione ZNFR contiene

- segnali di richiesta di lettura/scrittura
- segnali di completamento lettura/scrittura
- operando di completamento stazione host (d)
- indicazione dello stato di completamento operazione (completamento di una operazione con e senza errori) ($d+1$).

I segnali e operandi descritti di seguito:

Segnali di richiesta di lettura/scrittura

Questo segnale viene attivato con l'esecuzione delle istruzioni dedicate di comunicazione della serie QnA. Il segnale viene azzerato con l'elaborazione di END del ciclo in cui le operazioni di lettura/scrittura vengono completate.

Segnali di completamento lettura/scrittura

Questo segnale viene attivato con l'esecuzione delle istruzioni dedicate di comunicazione della serie QnA. Il segnale viene azzerato con l'elaborazione di END del ciclo in cui le operazioni di lettura/scrittura vengono completate.

Operando di completamento stazione host

Questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione ZNFR viene completata. L'operando viene azzerato con la successiva istruzione END.

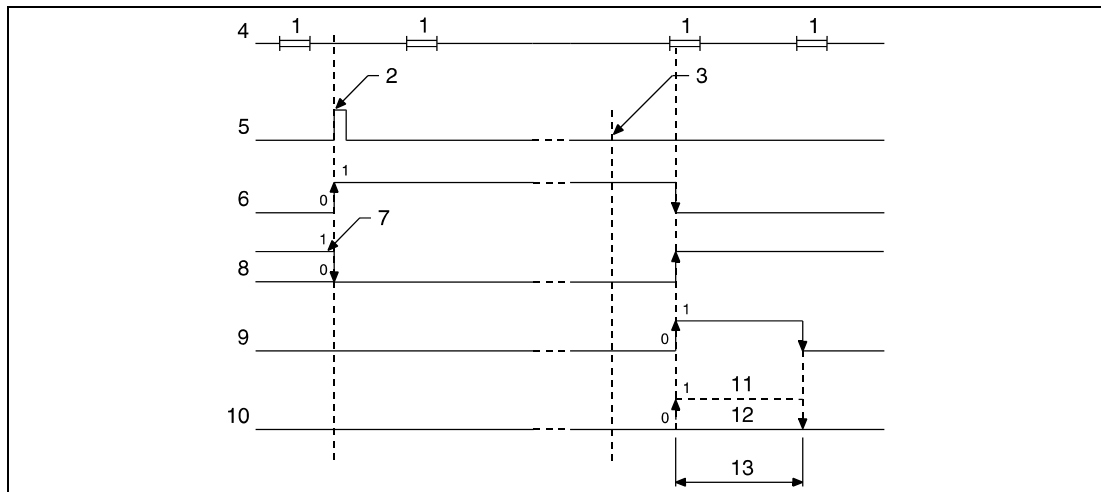
Indicazione stato completamento dell'operazione

Questo operando viene attivato in funzione del risultato di completamento dell'istruzione.

Rimane a zero in caso di trasmissione normale (senza errori).

In caso di completamento con errore di una trasmissione, questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione ZNFR viene completata. L'operando viene azzerato con la successiva istruzione END.

La figura seguente mostra le operazioni della stazione host durante l'esecuzione di una istruzione ZNFR:



- 1 Elaborazione END
- 2 Esecuzione istruzione ZNFR
- 3 Completamento operazione
- 4 Programma della stazione host
- 5 Istruzione ZNFR
- 6 Segnale di richiesta di lettura/scrittura
- 7 Al termine dell'esecuzione dell'istruzione di comunicazione dedicata della serie Q
- 8 Completamento dell'operazione di lettura/scrittura
- 9 Operando di completamento della stazione host (d) attivato al termine dell'operazione
- 10 Indicazione stato completamento dell'operazione (d+1)
- 11 Completamento trasmissione con errore
- 12 Completamento di una trasmissione senza errori
- 13 Una scansione

I registri di comunicazione in s2 vengono impostati con i parametri di rete "M ← R (da stazione master a stazione di I/O remoto)" e sono allocati all'interno del campo specificato dai parametri di rinfresco della comunicazione.

L'esecuzione dell'istruzione ZNFR comporta l'uso di relé e registri di comunicazione da parte del sistema operativo. Il numero di relé e registri di comunicazione usati dal sistema operativo in funzione del modulo funzione speciale è il seguente:

Per $M \rightarrow R$ (da stazione master a stazione di I/O remoto):

Relé di comunicazione = 4, registri di comunicazione = 4

Per $M \leftarrow R$ (a stazione master da stazione di I/O remoto):

Relé di comunicazione = 4, registri di comunicazione = 4

Errori di esecuzione

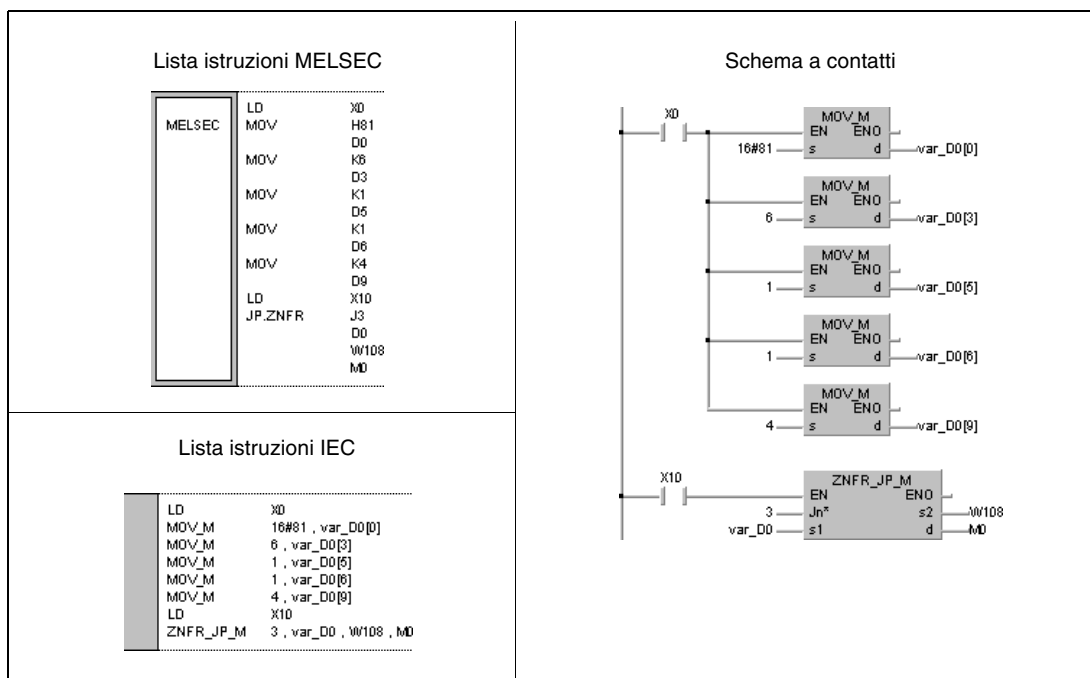
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- I contenuti dei dati di controllo superano i campi ammessi (codice di errore 4100).
- La rete con il numero specificato da Jn non è collegata alla stazione (codice di errore 4102).
- Il modulo con l'indirizzo di I/O specificato da Un non è un modulo di rete (codice di errore 2111).

Esempio di programma JP.ZNFR

Con il fronte positivo di X10, il programma seguente legge gli indirizzi da 6 a 9 del buffer di memoria di un modulo funzione speciale all'interno di una stazione di I/O remoto. I dati letti vengono memorizzati nei registri di comunicazione da W108 a W10B. Ulteriori dettagli sulla stazione di I/O remoto e le istruzioni MOV utilizzate, sono contenute nella tabella seguente:

Operando/Istruzione	Significato/Funzione
Stazione di I/O	1R1
Rete della stazione di I/O	3
Modulo funzione speciale	1
1. Istruzione MOV	Imposta i dati orologio
2. Istruzione MOV	Imposta il primo indirizzo del buffer di memoria (6)
3. Istruzione MOV	Imposta il numero della stazione di I/O
4. Istruzione MOV	Imposta la posizione del modulo funzione speciale in ordine di sequenza
5. Istruzione MOV	Specifica la lunghezza dei dati da leggere.

**NOTA**

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

8.6.9 ZNTO

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	

Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	—	9	
s2	—	● ¹	—	—	—	—	—	—			
d	●	—	—	—	—	—	—	—			

¹ Solo registri di comunicazione

GX IEC Developer

<p>Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <p>MELSEC</p> </div> <p>JP.ZNTO Jn s1 s2 d</p> <p>GP.ZNTO Un s1 s2 d</p>	<p>Schema a contatti</p> <div style="margin: 10px 0;"> </div> <div style="margin: 10px 0;"> </div>	<p>Lista istruzioni IEC</p> <div style="margin: 10px 0;"> </div>
--	--	--

Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
Jn	Numero di rete della stazione host ● ¹	BIN 16-bit	ANY16
Un	Indirizzo intestazione I/O per unità di rete della stazione host. ● ²		
s1	Indirizzo del primo operando che memorizza i dati di controllo.	Indirizzo operando	Array [1..16] di ANY16
s2	Indirizzo del primo registro di comunicazione (W) della stazione host che contiene i dati da scrivere.		ANY16
d	Operando attivato per una scansione al completamento dell'istruzione.	Bit	BOOL

NOTA

- ¹ Il numero di rete della stazione host può variare nel campo da 1 a 239. La rete di numero 254 viene configurata tramite parametrizzazione per consentire l'accesso alla stazione attiva da parte delle altre stazioni.
- ² L'indirizzo di intestazione di I/O dell'unità di rete per la stazione host deve essere nel campo da 0 a FEH. Notare, che il compilatore considera Un come valore esadecimale. Un numero decimale viene convertito automaticamente in valore esadecimale.

Panoramica degli operandi per i dati di controllo

Operando	Significato	Funzione	Valore Campo	Impostato da
(s1)+0 Array_s1[1]	Modo di esecuzione	Conferma completamento trasmissione attivo (Bit 0 (b0) = 1, fisso)	0001H 0081H	Utente
	Modo completamento errore	Memorizzazione dati impostazione orologio al termine dell'elaborazione errore: - nessuna memorizzazione dati orologio, Bit 7 (b7) = 0 - memorizzazione dati orologio, Bit 7 (b7) = 1 (dati orologio a partire da (s1)+11 (Array_s1[12]))		
(s1)+1 Array_s1[2]	Stato completamento esecuzione istruzione	Stato al completamento dell'esecuzione dell'istruzione: 0 = nessun errore (completamento normale) < > 0 = codice di errore ● ³	—	Sistema
(s1)+2 Array_s1[3]	Dummy	Non utilizzato	—	—
(s1)+3 Array_s1[4]	Indirizzo buffer di memoria	Imposta il primo indirizzo del buffer di memoria.	● ⁴	Utente
(s1)+4 Array_s1[5]	Dummy	Non utilizzato	—	—
(s1)+5 Array_s1[6]	Numero della stazione target.	Imposta il numero di rete della stazione target.	da 1 a 64	Utente
(s1)+6 Array_s1[7]	Posizione del modulo funzione speciale	Imposta la posizione del modulo funzione speciale nella serie di moduli funzione speciali installati sulla stazione target.	—	Utente
(s1)+7 Array_s1[8]	Dummy	Non utilizzato	—	—
(s1)+8 Array_s1[9]	Dummy	Non utilizzato	—	—
(s1)+9 Array_s1[10]	Lunghezza dei dati	Specifica il numero di dati da scrivere.	da 1 a 256	Utente
(s1)+10 Array_s1[11]	Dummy	Non utilizzato	—	—
(s1)+11 Array_s1[12]	Segnalazione orologio impostato (attivato solo su errore)	Contiene lo stato di abilitazione/disabilitazione dati orologio impostato in (s1)+0 (Array_s1[1]): - Memorizzazione dati orologio disabilitata = 0 - Memorizzazione dati orologio abilitata = 1	—	Sistema
(s1)+12 Array_s1[13]	Dati orologio (attivato solo su errore)	Byte pesante = anno (da 0 a 99) Byte leggero = mese (da 1 a 12)	—	Sistema
(s1)+13 Array_s1[14]		Byte pesante = giorno (da 1 a 31) Byte leggero = ora (da 0 a 23)		
(s1)+14 Array_s1[15]		Byte pesante = minuto (da 0 a 59) Byte leggero = secondo (da 0 a 59)		
(s1)+15 Array_s1[16]		Byte pesante = 00H Byte leggero = giorno della settimana (da 0 a 6) (domenica = 0, sabato = 6)		

●³ Fare riferimento al manuale MELSECNET/10 per i sistemi di rete QnA per ulteriori dettagli.

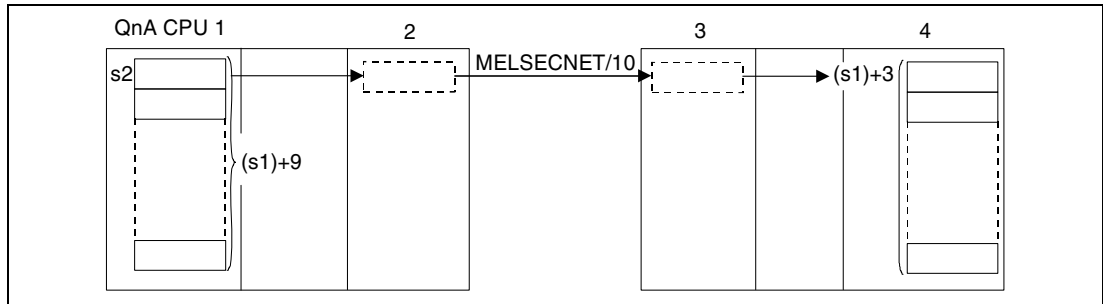
●⁴ Per ulteriori dettagli, fare riferimento al manuale relativo al modulo funzione speciale corrispondente.

Funzioni **Scrittura dati su moduli funzione speciali in stazioni I/O remote**

ZNT0 Istruzione di scrittura

L'istruzione ZNT0 scrive i dati contenuti nella stazione host a partire da s2 nel buffer di memoria di un modulo funzione speciale all'interno di una stazione di I/O remoto collegata a MELSECNET/10. La stazione di I/O remoto è specificata nei dati di controllo.

Al termine dell'operazione viene attivato l'operando specificato da d.



¹ Stazione host / stazione master

² Modulo di rete (stazione host / stazione master)

³ Stazione I/O remoti (stazione target)

⁴ Modulo funzione speciale (stazione target / stazione I/O remoto)

L'operazione di scrittura può essere eseguita solo da una stazione master collegata a MELSECNET/10 verso una stazione di I/O remoto collegata alla stessa rete.

L'istruzione ZNT0 non può essere eseguita contemporaneamente da più di una locazione dallo stesso modulo funzione speciale. In caso di esecuzione simultanea dell'istruzione da due o più locazioni, un meccanismo di interblocco fra le due stazioni attive impedisce l'esecuzione di ulteriori istruzioni ZNT0.

Il segnale di interblocco inviato durante l'esecuzione dell'istruzione ZNT0 contiene

- segnali di richiesta di lettura/scrittura
- segnali di completamento lettura/scrittura
- operando di completamento stazione host (d)
- indicazione dello stato di completamento operazione (completamento di una operazione con e senza errori) (d+1).

I segnali e operandi descritti di seguito:

Segnali di richiesta di lettura/scrittura

Questo segnale viene attivato con l'esecuzione delle istruzioni dedicate di comunicazione della serie QnA. Il segnale viene azzerato con l'elaborazione di END del ciclo in cui le operazioni di lettura/scrittura vengono completate.

Segnali di completamento lettura/scrittura

Questo segnale viene attivato con l'esecuzione delle istruzioni dedicate di comunicazione della serie QnA. Il segnale viene azzerato con l'elaborazione di END del ciclo in cui le operazioni di lettura/scrittura vengono completate.

Operando di completamento stazione host

Questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione ZNT0 viene completata. L'operando viene azzerato con la successiva istruzione END.

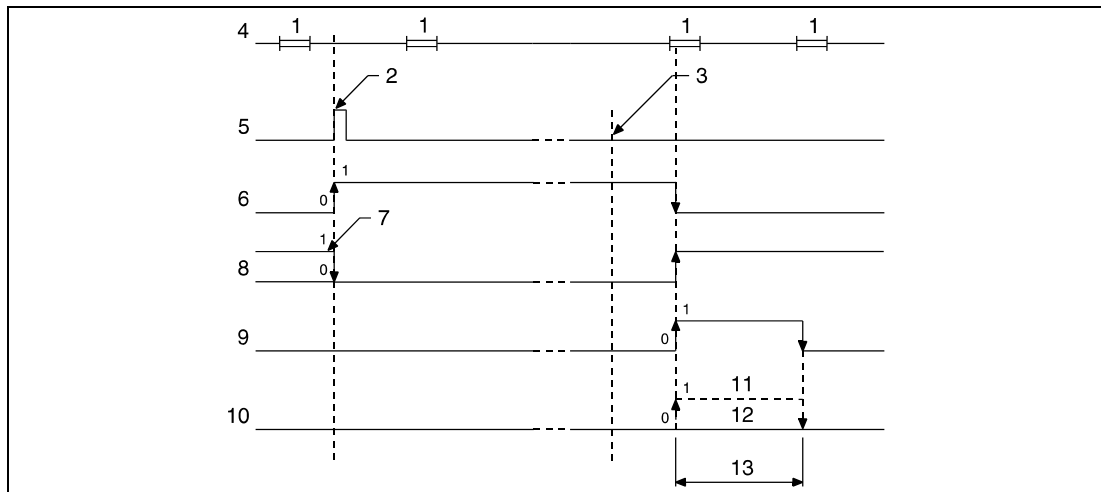
Indicazione stato completamento dell'operazione

Questo operando viene attivato in funzione del risultato di completamento dell'istruzione.

Rimane a zero in caso di trasmissione normale (senza errori).

In caso di completamento con errore di una trasmissione, questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione ZNTO viene completata. L'operando viene azzerato con la successiva istruzione END.

La figura seguente mostra le operazioni della stazione host durante l'esecuzione di una istruzione ZNTO:



- 1 Elaborazione END
- 2 Esecuzione istruzione ZNTO
- 3 Completamento operazione
- 4 Programma della stazione host
- 5 Istruzione ZNTO
- 6 Segnale di richiesta di lettura/scrittura
- 7 Al termine dell'esecuzione dell'istruzione di comunicazione dedicata della serie QnA
- 8 Completamento dell'operazione di lettura/scrittura
- 9 Operando di completamento della stazione host (d) attivato al termine dell'operazione
- 10 Indicazione stato completamento dell'operazione (d+1)
- 11 Completamento trasmissione con errore
- 12 Completamento di una trasmissione senza errori
- 13 Una scansione

I registri di comunicazione in s2 vengono impostati con i parametri di rete "M ← R (da stazione master a stazione di I/O remoto)" e sono allocati all'interno del campo specificato dai parametri di rinfresco della comunicazione.

L'esecuzione dell'istruzione ZNTO comporta l'uso di relé e registri di comunicazione da parte del sistema operativo. Il numero di relé e registri di comunicazione usati dal sistema operativo in funzione del modulo funzione speciale è il seguente:

Per $M \rightarrow R$ (da stazione master a stazione di I/O remoto):

Relé di comunicazione = 4, registri di comunicazione = 4

Per $M \leftarrow R$ (a stazione master da stazione di I/O remoto):

Relé di comunicazione = 4, registri di comunicazione = 4

Errori di esecuzione

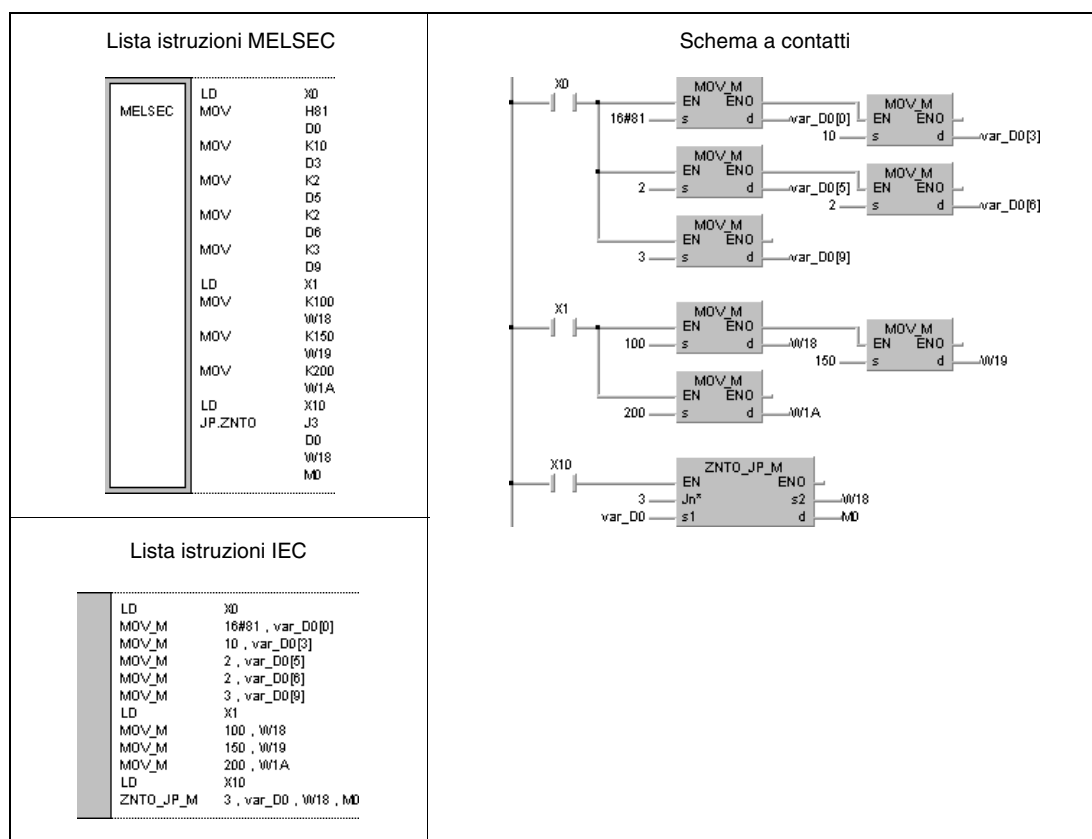
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- I contenuti dei dati di controllo superano i campi ammessi (codice di errore 4100).
- La rete con il numero specificato da Jn non è collegata alla stazione (codice di errore 4102).
- Il modulo con l'indirizzo di I/O specificato da Un non è un modulo di rete (codice di errore 2111).

Esempio di programma JP.ZNTO

Con il fronte positivo di X10, il programma seguente scrive dati negli indirizzi da 10 a 12 del buffer di memoria di un modulo funzione speciale all'interno di una stazione di I/O remoto. I dati da scrivere sono contenuti nella stazione host all'interno dei registri di comunicazione da W18 a W1A. Ulteriori dettagli sulla stazione di I/O remoto e le istruzioni MOV utilizzate, sono contenute nella tabella seguente:

Operando/Istruzione	Significato/Funzione
Stazione di I/O	1R2
Rete della stazione di I/O	3
Modulo funzione speciale	2
1. Istruzione MOV	Imposta i dati orologio
2. Istruzione MOV	Imposta il primo indirizzo del buffer di memoria (10)
3. Istruzione MOV	Imposta il numero della stazione di I/O
4. Istruzione MOV	Imposta la posizione del modulo funzione speciale in ordine di sequenza
5. Istruzione MOV	Specifica la lunghezza dei dati da leggere.
6. Istruzione MOV	Scrive dati nei registri di comunicazione da W18 a W1A
7. Istruzione MOV	
8. Istruzione MOV	

**NOTA**

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

8.7 Istruzioni di comunicazione dati compatibili con serie A

Queste istruzioni supportano la comunicazione dati tra stazioni con CPU QnA, tra stazioni con CPU QnA e ACPU e fra CPU QnA o ACPU e stazioni di I/O remoto collegate a MELSECNET e MELSECNET/10. La tabella seguente mostra una panoramica di queste istruzioni:

Funzione	MELSEC Istruzione in editor MELSEC	MELSEC Istruzione con Editor IEC	Stazione designata in MELSECNET/10			MELSECNET
			CPU QnA	ACPU	Stazione I/O remoto	
Legge dati QnA da stazioni target in reti target	J.ZNRD	ZNRD_J_M	●	●	—	—
	JP.ZNRD	ZNRD_JP_M				
Lettura dati da stazioni locali (solo stazioni ma- ster)	J.ZNRD	ZNRD_J_M	—	—	—	●
	JP.ZNRD	ZNRD_JP_M				
Scrive dati QnA su stazioni oggetto in reti target	J.ZNWR	ZNWR_J_M	●	●	—	—
	JP.ZNWR	ZNWR_JP_M				
Scrive dati su stazioni locali (solo stazioni ma- ster)	J.ZNWR	ZNWR_J_M	—	—	—	●
	JP.ZNWR	ZNWR_JP_M				
Solo serie A: Lettura dati da stazioni locali (solo stazioni ma- ster)	LRDP	LRDP_M	—	—	—	●
		LRDP_MD				
		LRDP_P_MD				
Solo serie A: Scrive dati su stazioni locali (solo stazioni ma- ster)	LWTP	LWTP_M	—	—	—	●
		LWTP_MD				
		LWTP_M_MD				
Lettura dati da moduli funzione speciali in stazioni I/O remote	RFRP/ G.RFRP	RFRP_U_M	—	—	—	●
		RFRP_UP_M				
Scrittura dati su moduli funzione speciali in stazioni I/O remote.	RTOP G.RTOP	RTOP_U_M	—	—	—	●
		RTOP_UP_M				

8.7.1 ZNRD

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	

Operandi
MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
n1	●	●	●	—	—	—	—	●	—	SM0	12
s	—	● ¹	—	—	—	—	—	—	—		
d1	—	●	●	—	—	—	—	—	—		
n2	●	●	●	—	—	—	—	●	—		
d2	●	●	●	—	—	—	—	—	—		

¹ solo T, C, D, e W

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
<div style="border: 1px solid black; padding: 5px; width: fit-content;">MELSEC</div> <p>J.ZNRD Jn n1 s d1 n2 d2</p>		<p>ZNRD_J_M Jn , n1 , s , n2 , d1 , d2</p>

Variabili

Dati impostati	Significato	Tipo dati
Jn	Numero di rete della stazione host ● ¹	BIN 16-bit
n1	Numero della stazione target.	
s	Indirizzo del primo operando della stazione che contiene i dati da leggere.	
d1	Primo operando della stazione host che contiene i dati letti.	
n2	Lunghezza dati da leggere	Bit
d2	Operando attivato per una scansione al completamento dell'istruzione.	

NOTA

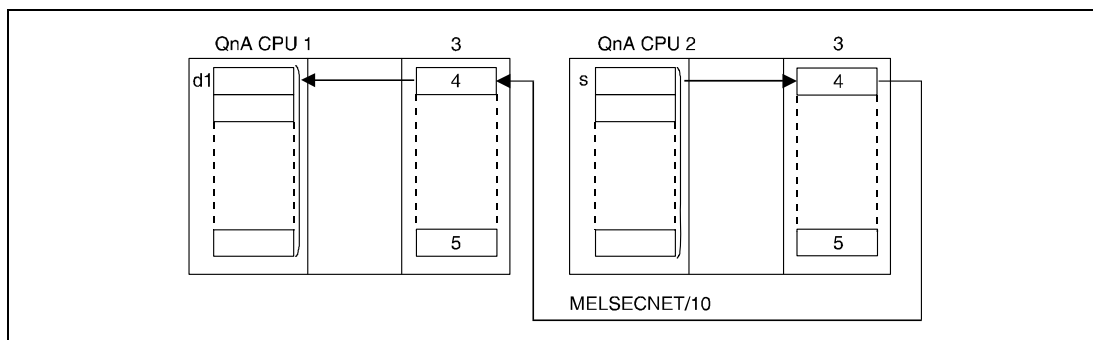
●¹ Specificare J0 nelle istruzioni usate con MELSECNET(I/II/B).

Funzioni **Richiesta dati da altre stazioni**

ZNRD Istruzione di lettura

L'istruzione ZNRD legge il numero di word dati specificato da n2 e contenute nella stazione target di MELSECNET/10. Il numero di stazione è specificato da n1. Il numero di rete è contenuto in Jn. I dati letti vengono memorizzati nella stazione host a partire da d1.

Al termine dell'operazione viene attivato l'operando specificato da d2.



- 1 Stazione host
- 2 Stazione target
- 3 Modulo di rete
- 4 Canale 1
- 5 Canale 8

L'operazione di lettura può essere eseguita solo con una stazione target collegata alla stessa rete MELSECNET/10 della stazione host.

L'operazione di lettura da una stazione locale può essere eseguita solo con una stazione master collegata a una rete MELSECNET.

Il numero di rete Jn può variare tra 1 e 239. La specifica del numero di rete 0 (J0) è simile alla specifica nel sistema MELSECNET.

Nel sistema MELSECNET il numero della rete target (Jn) è fissato a 0 (J0). I numeri delle reti target (Jn) da 1 a 239 vengono usati con MELSECNET/10.

Il numero di stazione n1 può variare fra 1 e 64.

Nel sistema MELSECNET/B il numero di stazione può variare tra 1 e 31.

La lunghezza dei dati da leggere n2 (numero di word dati) può variare fra 1 e 230.

Le operazioni di lettura da altre stazioni tramite l'istruzione ZNRD possono essere eseguite allo stesso modo da stazioni con CPU AnU e QnA.

Le istruzioni di comunicazione non possono essere eseguite contemporaneamente da più locazioni con accesso comune allo stesso canale. L'esecuzione simultanea da due o più locazioni viene evitata da un meccanismo di interblocco fra le due stazioni attive.

Entrambe le stazioni, host e target, usano il canale 1 del modulo di rete per l'esecuzione dell'istruzione ZNRD. Per poter eseguire più istruzioni ZNRD, si deve poter accedere diverse volte al canale 1, mentre il canale 1 del modulo di rete può essere usato solo una volta per istruzione. Per evitare l'esecuzione simultanea di più istruzioni, si deve stabilire un interblocco tra i segnali di richiesta di lettura/scrittura e l'operando di completamento operazione.

L'esecuzione e il completamento dell'istruzione ZNRD viene indicato tramite il flag della direttiva di comunicazione (SB30) e l'operando di completamento della stazione host (d2) come segue:

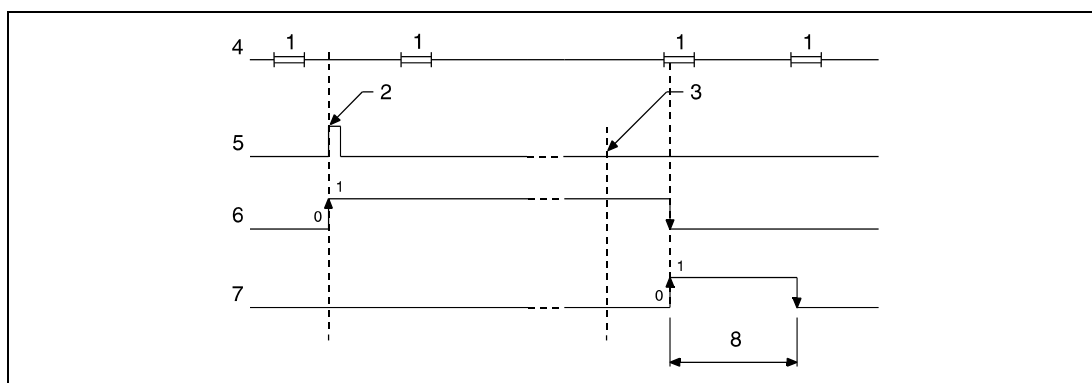
Flag direttiva di comunicazione

Questo flag è attivato durante l'esecuzione dell'istruzione ZNRD. Il flag è azzerato con l'elaborazione dell'istruzione END alla fine della scansione di programma in cui viene completata l'operazione di lettura.

Operando di completamento stazione host

Questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione viene completata. L'operando viene azzerato con la successiva istruzione END.

La figura seguente mostra le operazioni della stazione host durante l'esecuzione di una istruzione ZNRD:



- 1 Elaborazione END
- 2 Esecuzione istruzione ZNRD
- 3 Completamento operazione
- 4 Programma della stazione host
- 5 Istruzione ZNRD
- 6 Flag direttiva di comunicazione (SB30)
- 7 Operando di completamento della stazione host (d2) attivato al termine dell'operazione
- 8 Una scansione

Gli stati di esecuzione e completamento (normale, anomalo) dell'istruzione ZNRD sono indicati tramite il registro di completamento operazione dell'istruzione ZNRD (SW31) come segue:

In caso di completamento senza errori (normale) dell'operazione, il contenuto del registro SW31 è 0.

In caso di completamento con errore (anomalo) dell'operazione, il codice di errore corrispondente viene memorizzato nel registro SW31.

NOTA

Fare riferimento al manuale MELSECNET/10 per i sistemi di rete QnA per ulteriori dettagli.

Errori di esecuzione

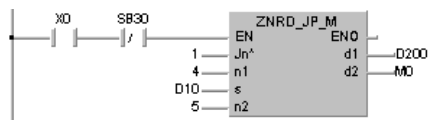
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- La lunghezza dei dati da leggere specificata da n2 supera il campo ammesso per l'operando indicato da s1 (codice di errore 4101).
- La rete con il numero specificato da Jn non esiste (codice di errore 4102).
- La stazione con il numero specificato da n1 non esiste (codice di errore 4102).
- La lunghezza dei dati da leggere specificata da n2 non è nel campo da 1 a 230 (codice di errore 4100).

Esempio di programma

JP.ZNRD

Con il fronte positivo di X0, il programma seguente legge i dati dai registri da D10 a D14 nella stazione numero 4. I dati letti vengono memorizzati nei registri da D200 a D204 della stazione host. Stazione host e stazione target sono collegate alla rete numero 1.

<p style="text-align: center;">Lista istruzioni MELSEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20%; text-align: center;">MELSEC</td> <td style="width: 30%; border-top: 1px dotted black;">LD</td> <td style="width: 50%; border-top: 1px dotted black;">X0</td> </tr> <tr> <td></td> <td style="border-top: 1px dotted black;">ANI</td> <td style="border-top: 1px dotted black;">S830</td> </tr> <tr> <td></td> <td style="border-top: 1px dotted black;">JP.ZNRD</td> <td style="border-top: 1px dotted black;">J1</td> </tr> <tr> <td></td> <td></td> <td style="border-top: 1px dotted black;">K4</td> </tr> <tr> <td></td> <td></td> <td style="border-top: 1px dotted black;">D10</td> </tr> <tr> <td></td> <td></td> <td style="border-top: 1px dotted black;">D200</td> </tr> <tr> <td></td> <td></td> <td style="border-top: 1px dotted black;">K5</td> </tr> <tr> <td></td> <td></td> <td style="border-top: 1px dotted black;">M0</td> </tr> </table>	MELSEC	LD	X0		ANI	S830		JP.ZNRD	J1			K4			D10			D200			K5			M0	<p style="text-align: center;">Schema a contatti</p> 
MELSEC	LD	X0																							
	ANI	S830																							
	JP.ZNRD	J1																							
		K4																							
		D10																							
		D200																							
		K5																							
		M0																							
<p style="text-align: center;">Lista istruzioni IEC</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20%; border-top: 1px dotted black;">LD</td> <td style="width: 30%; border-top: 1px dotted black;">X0</td> </tr> <tr> <td style="border-top: 1px dotted black;">ANDN</td> <td style="border-top: 1px dotted black;">S830</td> </tr> <tr> <td style="border-top: 1px dotted black;">ZNRD JP_M</td> <td style="border-top: 1px dotted black;">1, 4, D10, 5, D200, M0</td> </tr> </table>	LD	X0	ANDN	S830	ZNRD JP_M	1, 4, D10, 5, D200, M0																			
LD	X0																								
ANDN	S830																								
ZNRD JP_M	1, 4, D10, 5, D200, M0																								

8.7.2 ZNWR

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	

Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
n1	●	●	●	—	—	—	—	●	—	SM0	12
d1	—	● ¹	—	—	—	—	—	—	—		
s	—	●	●	—	—	—	—	—	—		
n2	—	●	●	—	—	—	—	●	—		
d2	●	●	●	—	—	—	—	—	—		

¹ solo T, C, D, e W

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
<div style="border: 1px solid black; padding: 5px; width: fit-content;">MELSEC</div> <p>J.ZNWR Jn n1 d1 s n2 d2</p>		<div style="border: 1px solid black; padding: 5px; width: fit-content;">ZNWR_J_M</div> <p>Jn , n1 , s , n2 , d1 , d2</p>

Variabili

Dati impostati	Significato	Tipo dati
Jn	Numero di rete della stazione host ● ¹	BIN 16-bit
n1	Numero della stazione target.	
d1	Primo operando della stazione target in cui vengono scritti i dati.	
s	Primo operando della stazione host che contiene i dati da scrivere.	
n2	Lunghezza dati inviati.	
d2	Operando attivato per una scansione al completamento dell'istruzione.	Bit

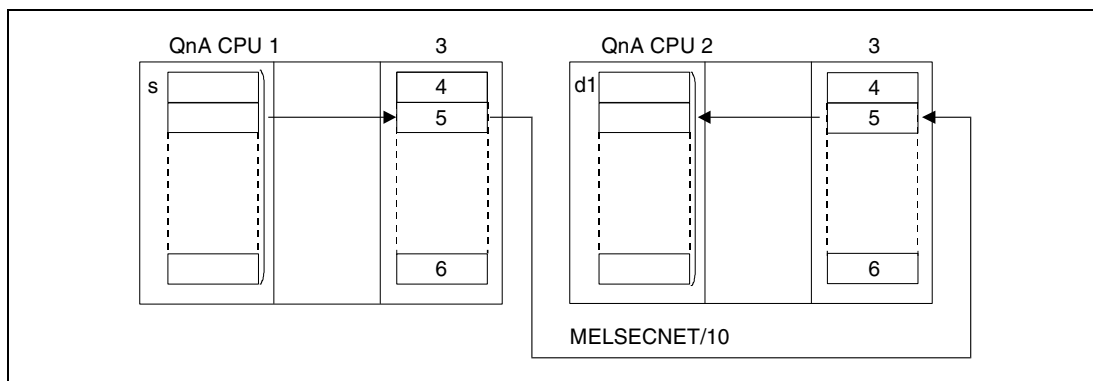
NOTA

●¹ Specificare J0 nelle istruzioni usate con MELSECNET(I/II/B).

Funzioni **Scrittura dati su altre stazioni**
ZNWR **Istruzione di scrittura**

L'istruzione ZNWR scrive il numero di word dati specificato da n2 e contenute in s nella stazione host, all'interno di una stazione target di MELSECNET/10. Il numero di stazione è specificato da n1. Il numero di rete è contenuto in Jn.

Al termine dell'operazione viene attivato l'operando specificato da d2.



- ¹ Stazione host
- ² Stazione target
- ³ Modulo di rete
- ⁴ Canale 1
- ⁵ Canale 2
- ⁶ Canale 8

L'operazione di scrittura può essere eseguita solo con una stazione target collegata alla stessa rete MELSECNET/10 della stazione host.

L'operazione di scrittura in una stazione locale può essere eseguita solo con una stazione master collegata a una rete MELSECNET.

Il numero di rete Jn può variare tra 1 e 239. La specifica del numero di rete 0 (J0) è simile alla specifica nel sistema MELSECNET.

Nel sistema MELSECNET il numero della rete target (Jn) è fissato a 0 (J0). I numeri delle reti target (Jn) da 1 a 239 vengono usati con MELSECNET/10.

Il numero di stazione n1 può variare fra 1 e 64.

Nel sistema MELSECNET/B il numero di stazione può variare tra 1 e 31.

La lunghezza dei dati da scrivere n2 (numero di word dati) può variare fra 1 e 230.

Le operazioni di lettura da altre stazioni tramite l'istruzione ZNWR possono essere eseguite allo stesso modo da stazioni con CPU AnU e QnA.

Entrambe le stazioni, host e target, usano il canale 2 del modulo di rete per l'esecuzione dell'istruzione ZNWR. Per poter eseguire più istruzioni ZNWR, si deve poter accedere diverse volte al canale 2, mentre il canale 2 del modulo di rete può essere usato solo una volta per istruzione. Per evitare l'esecuzione simultanea di più istruzioni, si deve stabilire un interblocco tra i segnali di richiesta di lettura/scrittura e l'operando di completamento operazione.

L'esecuzione e il completamento dell'istruzione ZNWR viene indicato tramite il flag della direttiva di comunicazione (SB30) e l'operando di completamento della stazione host (d2) come segue:

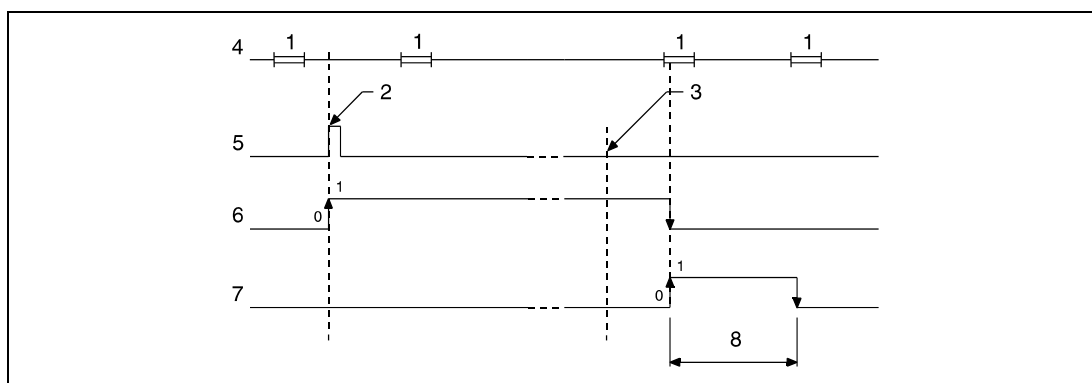
Flag direttiva di comunicazione

Questo flag è attivato durante l'esecuzione dell'istruzione ZNWR. Il flag è azzerato con l'elaborazione dell'istruzione END alla fine della scansione di programma in cui viene completata l'operazione di lettura.

Operando di completamento stazione host

Questo operando viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione viene completata. L'operando viene azzerato con la successiva istruzione END.

La figura seguente mostra le operazioni della stazione host durante l'esecuzione di una istruzione ZNWR:



- 1 Elaborazione END
- 2 Esecuzione istruzione ZNWR
- 3 Completamento operazione
- 4 Programma della stazione host
- 5 Istruzione ZNWR
- 6 Flag direttiva di comunicazione (SB32)
- 7 Operando di completamento della stazione host (d2) attivato al termine dell'operazione
- 8 Una scansione

Gli stati di esecuzione e completamento (normale, anomalo) dell'istruzione ZNWR sono indicati tramite il registro di completamento operazione dell'istruzione ZNWR (SW33) come segue:

In caso di completamento senza errori (normale) dell'operazione, il contenuto del registro SW33 è 0.

In caso di completamento con errore (anomalo) dell'operazione, il codice di errore corrispondente viene memorizzato nel registro SW33.

NOTA

Fare riferimento al manuale MELSECNET/10 per i sistemi di rete QnA per ulteriori dettagli.

Errori di esecuzione

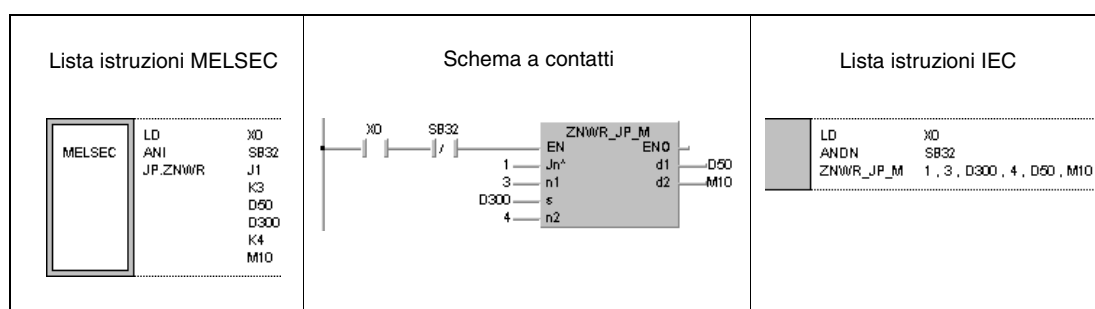
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- La lunghezza dei dati da inviare specificata da n2 supera il campo ammesso per l'operando indicato da s1 (codice di errore 4101).
- La rete con il numero specificato da Jn non esiste (codice di errore 4102).
- La stazione con il numero specificato da n1 non esiste (codice di errore 4102).
- La lunghezza dei dati da inviare specificata da n2 non è nel campo da 1 a 230 (codice di errore 4100).

Esempio di programma

JP.ZNWR

Con il fronte di salita di X0 il programma seguente scrive i dati contenuti nei registri da D300 a D303 della stazione host nei registri da D50 a D53 della stazione numero 3. La stazione host e la stazione target sono collegate alla rete numero 1.



8.7.3 LRDP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●		

Operandi MELSEC A

	Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Riporto	Errore		
	Operandi a bit							Operandi a word (16-bit)							Costanti						Punta-tore	
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z				V	K	H (16#)	P
n1																●	●					
s							●	●	●	●												
d							●	●	●	●												
n2																●	●					

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
<div style="border: 1px solid black; padding: 5px; display: inline-block;">MELSEC</div> LRDP n1 s d n2		LRDP_M s , n1 , n2 , d

Variabili

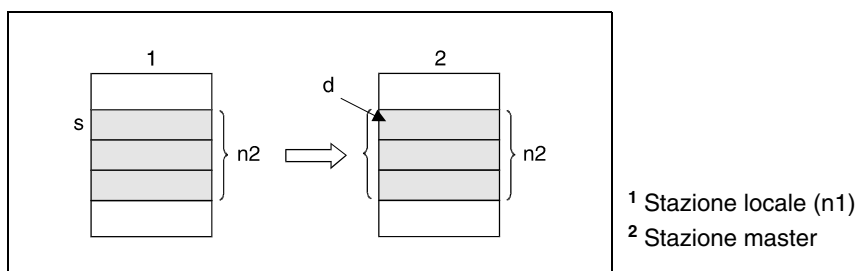
Dati impostati	Significato	Tipo dati
n1	Numero della stazione locale.	BIN 16-bit
s	Indirizzo iniziale dell'area dati della stazione locale che si deve leggere.	
d	Indirizzo dell'area della stazione master che riceve i dati letti.	
n2	Lunghezza dati da leggere	

Funzioni **Letture dati da una stazione locale****LRDP** **Istruzione di lettura**

L'istruzione LRDP legge dati da una stazione locale e li memorizza in un'area specificata della stazione master. L'indirizzo iniziale dell'area dati da leggere è specificato da s. La quantità dei dati (da 1 a 32) è specificata da n2. Il numero della stazione locale è specificato da n1. L'indirizzo dell'area di memoria interna della stazione master in cui trasferire i dati letti è indicato da d.

Durante l'esecuzione dell'istruzione LRDP, il relé speciale M9200 della stazione master viene attivato. Al termine dell'istruzione, M9201 viene attivato. Entrambi i relé speciali rimangono attivi dopo l'esecuzione dell'istruzione. Devono essere azzerati dal programma di sequenza.

Non è possibile eseguire contemporaneamente due o più istruzioni LRDP. L'istruzione LRDP non può nemmeno accedere ad una stazione locale contemporaneamente ad una istruzione LWTP.

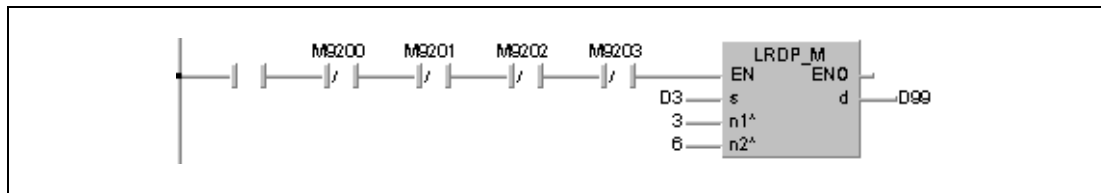
**NOTA**

I relé speciali M9200, M9201, M9202 e M9203 dovrebbero essere utilizzati come interblocchi e condizioni di ingresso per le istruzioni LRDP e LWTP per assicurare che non ci sia contemporaneità di esecuzione con altre istruzioni LRDP o LWTP.

Il risultato dell'esecuzione dell'istruzione LRDP viene restituito tramite il valore contenuto in D9200 (vedi tabella seguente):

Valore	Significato
LRDP D9200	
0	Completamento istruzione senza errori.
2	Errore di esecuzione per indirizzo non valido: Gli indirizzi in s e d superano i campi operando ammessi. Il valore contenuto in n1 supera il campo da 1 a 64. Il valore contenuto in n2 supera il campo da 1 a 32.
3	La stazione locale indirizzata è offline e non accessibile
4	Nessuna stazione locale corrisponde al numero stazione specificato (errore di elaborazione).

La figura seguente mostra un interblocco dell'istruzione LRDP:



Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

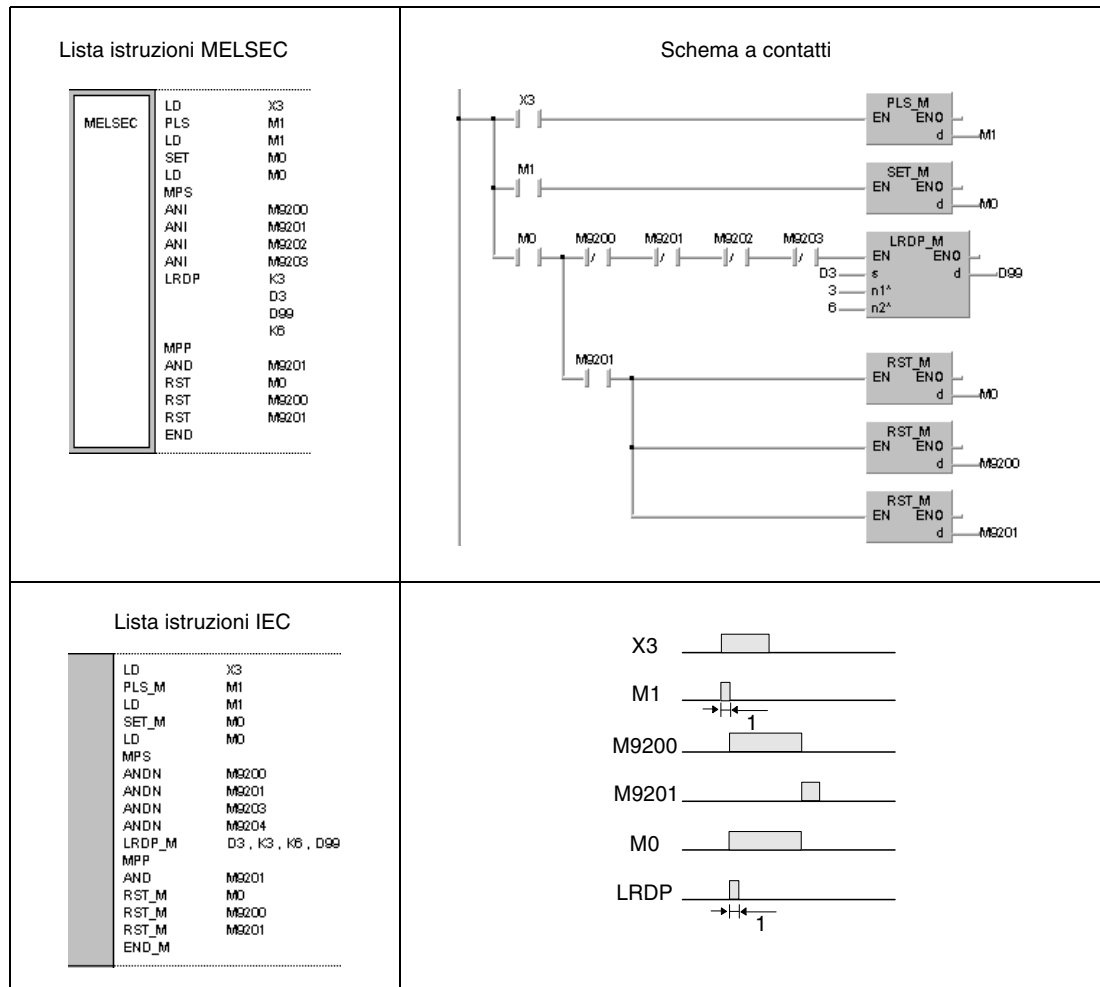
- Non esiste nessuna stazione locale con il numero specificato da n1, oppure il numero supera il campo da 1 a 64.
- Gli indirizzi in s e d superano i campi operando ammessi.
- La quantità di dati specificata da n2 supera il campo da 1 a 32.
- L'istruzione LRDP viene eseguita nel programma di una stazione locale.

NOTA

Se la CPU non supporta istruzioni di comunicazione dati, l'istruzione LRDP non viene eseguita. Nessun errore di funzionamento. M9200 è comunque attivo.

Esempio di programma LRDP

Il programma seguente legge i dati da D3 a D8 nella stazione locale 3 e li memorizza da D99 a D104 nella stazione master. Dopo l'attivazione di X3, M0 si attiva e viene eseguita l'istruzione LRDP. All'inizio del trasferimento dati viene attivato M9200. Al termine del trasferimento, si attiva M9201. L'istruzione LRDP non viene eseguita se un'altra istruzione LRDP o LWTP è già in esecuzione. Al termine del trasferimento (M9201 attivo) M0, M9200 e M9201 vengono azzerati nella prosecuzione del programma.



¹ Esecuzione singola

Il contatto corrispondente a M1 deve essere convertito in impulso. Altrimenti l'istruzione LRDP non viene eseguita completamente.

Il contatto corrispondente a M0 deve essere attivato con l'istruzione SET. Se viene programmata una istruzione OUT o PLS invece dell'istruzione SET, si possono verificare errori nell'esecuzione dell'istruzione LRDP.

Per evitare l'esecuzione simultanea di due istruzioni LRDP, si deve realizzare un interblocco per mezzo dei relé speciali M9200 e M9201.

Se all'interno dello stesso programma si accede ad una stazione locale con l'istruzione LWTP, è necessario programmare nell'interblocco anche i relé M9202 e M9203.

8.7.4 LWTP

CPU


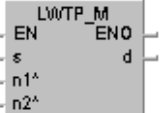

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●		

Operandi MELSEC A

	Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Riporto	Errore					
	Operandi a bit								Operandi a word (16-bit)													Costanti		Puntatore		Livello
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V				K	H (16#)	P	I	N	M9012	M9010 M9011
n1																●	●									
d							●	●	●	●																●
s							●	●	●	●																
n2																●	●									

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
 <div style="display: inline-block; vertical-align: middle; margin-left: 10px;"> LWTP n1 d s n2 </div>	 <div style="display: inline-block; vertical-align: middle; margin-left: 10px;"> LWTP_M - EN ENO - s d - n1¹ - n2¹ </div>	 <div style="display: inline-block; vertical-align: middle; margin-left: 10px;"> LWTP_M s , n1 , n2 , d </div>

Variabili

Dati impostati	Significato	Tipo dati
n1	Numero della stazione locale.	BIN 16-bit
d	Indirizzo dell'area nella stazione locale in cui si deve scrivere.	
s	Indirizzo iniziale dell'area dati della stazione master contenente i dati da scrivere.	
n2	Lunghezza dati inviati.	

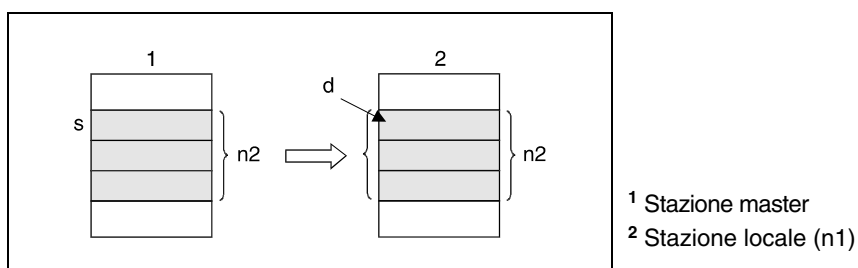
Funzioni Scrittura dati su una stazione locale

LWTP Istruzione di scrittura

L'istruzione LWTP scrive i dati della stazione master in un'area di indirizzo specificato di una stazione locale. L'indirizzo iniziale dell'area dati da scrivere è specificato da s. La quantità dei dati (da 1 a 32) è specificata da n2. Il numero della stazione locale è specificato da n1. L'indirizzo dell'area della stazione locale in cui si deve scrivere è specificato da d.

Durante l'esecuzione dell'istruzione LWTP, il relé speciale M9202 della stazione master viene attivato. Al termine dell'istruzione, M9203 viene attivato. Entrambi i relé speciali rimangono attivi dopo l'esecuzione dell'istruzione. Devono essere azzerati dal programma di sequenza.

Non è possibile eseguire contemporaneamente due o più istruzioni LWTP. L'istruzione LWTP non può nemmeno accedere ad una stazione locale contemporaneamente ad una istruzione LRDP.



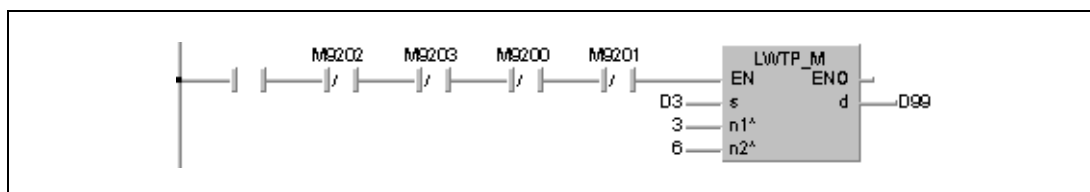
NOTA

I relé speciali M9200, M9201, M9202 e M9203 dovrebbero essere utilizzati come interblocchi e condizioni di ingresso per le istruzioni LRDP e LWTP per assicurare che non ci sia contemporaneità di esecuzione con altre istruzioni LRDP o LWTP.

Il risultato dell'esecuzione dell'istruzione LWTP viene restituito tramite il valore contenuto in D9201 (vedi tabella seguente):

Valore	Significato
LWTP D9201	
0	Completamento istruzione senza errori.
2	Errore di esecuzione per indirizzo non valido: Gli indirizzi in s e d superano i campi operando ammessi. Il valore contenuto in n1 supera il campo da 1 a 64. Il valore contenuto in n2 supera il campo da 1 a 32.
3	La stazione locale indirizzata è offline e non accessibile
4	Nessuna stazione locale corrisponde al numero stazione specificato (errore di elaborazione).

La figura seguente mostra un interblocco dell'istruzione LWTP:



Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

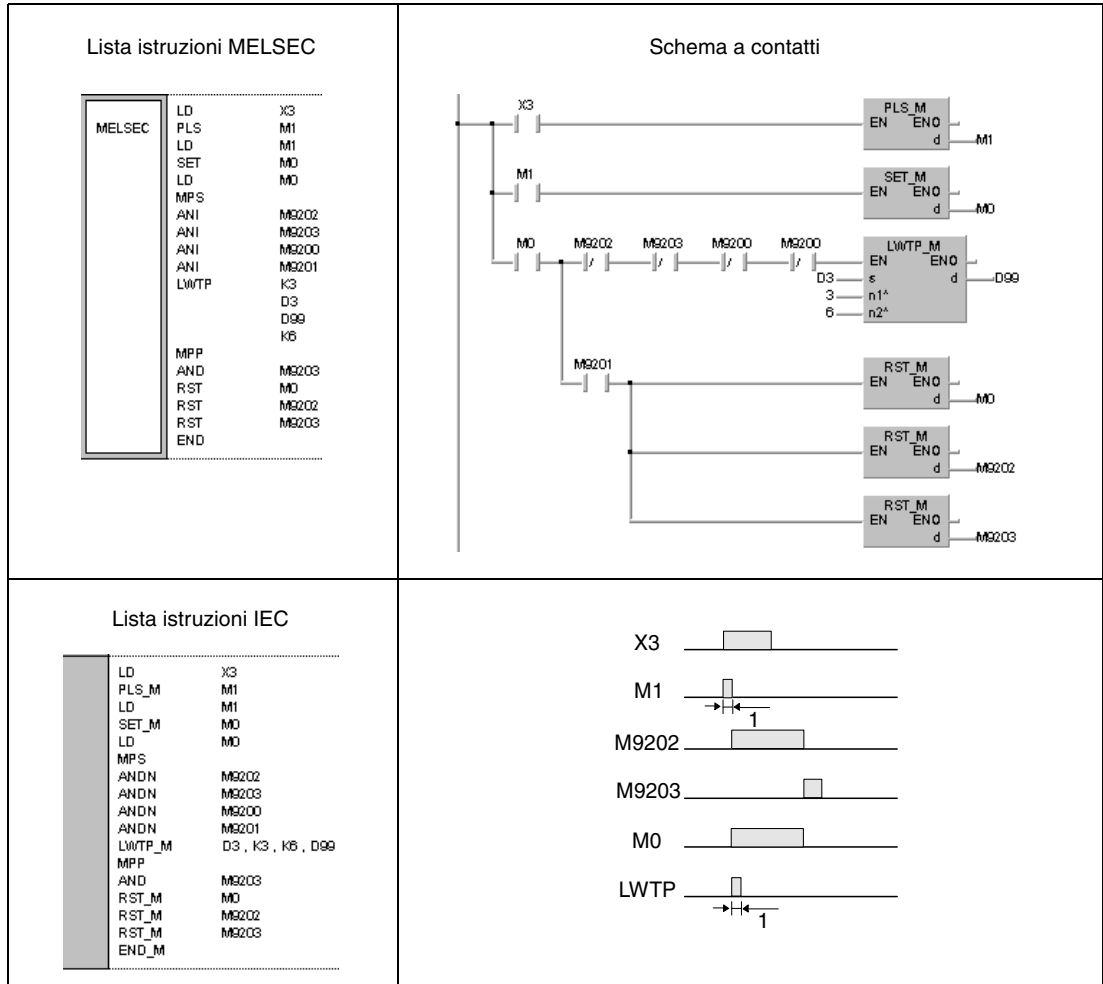
- Non esiste nessuna stazione locale con il numero specificato da n1, oppure il numero supera il campo da 1 a 64.
- Gli indirizzi in s e d superano i campi operando ammessi.
- La quantità di dati specificata da n2 supera il campo da 1 a 32.
- L'istruzione LWTP viene eseguita nel programma di una stazione locale.

NOTA

Se la CPU non supporta istruzioni di comunicazione dati, l'istruzione LWTP non viene eseguita. Nessun errore di funzionamento. M9202 è comunque attivo.

Esempio di programma LWTP

Il programma seguente scrive i dati da D99 a D104 della stazione master nei registri da D3 a D8 nella stazione locale 3. Dopo l'attivazione di X3, si attiva M0 e viene eseguita l'istruzione LWTP. All'inizio del trasferimento dati viene attivato M9202. Al termine del trasferimento, si attiva M9203. L'istruzione LWTP non viene eseguita se un'altra istruzione LWTP o LRDP è già in esecuzione. Al termine del trasferimento M0, M9202 e M9203 vengono azzerati nella prosecuzione del programma.



¹ Esecuzione singola

NOTA

Il contatto corrispondente a M1 deve essere convertito in impulso. Altrimenti l'istruzione LWTP non viene eseguita completamente.

Il contatto corrispondente a M0 deve essere attivato con l'istruzione SET. Se viene programmata una istruzione OUT o PLS invece dell'istruzione SET, si possono verificare errori nell'esecuzione dell'istruzione LWTP.

Per evitare l'esecuzione simultanea di due istruzioni LWTP, si deve realizzare un interblocco per mezzo dei relé speciali M9202 e M9203.

Se all'interno dello stesso programma si accede ad una stazione locale con l'istruzione LRDP, è necessario programmare nell'interblocco anche i relé M9200 e M9201.

8.7.5 RFRP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	

Operandi MELSEC A

	Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Riporto	Errore				
	Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello									
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z				V	K	H (16#)	P	I	N
n1																●	●							
n2																●	●							
d										●											11 ¹	●		●
n3																●	●							


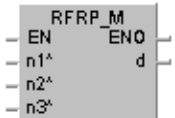

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

Operandi MELSEC Q


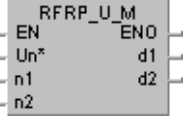

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
n1	●	●	●	—	—	—	—	●	—	SM0	9
d1	—	● ¹	—	—	—	—	—	—			
n2	●	●	●	—	—	—	—	●			
d2	●	●	●	—	—	—	—	—			

¹ Solo registri di comunicazione

serie A
GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
 RFRP n1 n2 d n3		 RFRP_M n1 , n2 , n3 , d

Serie QnA
GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
 G.RFRP Un n1 d1 n2 d2		 RFRP_U_M Un , n1 , n2 , d1 , d2

Variabili

Dati impostati		Significato	Tipo dati
Serie A	Serie QnA		
n1	Un	Indirizzo intestazione I/O del modulo funzione speciale nella stazione I/O remoto ● ¹	BIN 16-bit
n2	n1	Primo indirizzo del buffer di memoria del modulo funzione speciale che contiene i dati da leggere.	
d	d1	Indirizzo del primo registro di comunicazione della stazione host che contiene il dato letto.	Operando numero
n3	n2	Lunghezza dati da leggere	BIN 16-bit
	d2	Operando attivato per una scansione al completamento dell'istruzione.	Bit

NOTA

- ¹ L'indirizzo di intestazione di I/O dell'unità di rete per la stazione host deve essere nel campo da 0 a FE_H. Notare, che il compilatore considera Un come valore esadecimale. Un numero decimale viene convertito automaticamente in valore esadecimale.

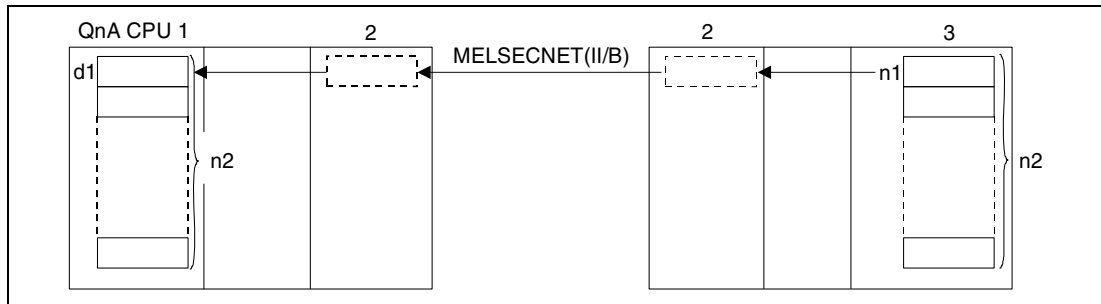
Funzioni **Letture dati da una stazione remota**

RFRP Istruzione di lettura

L'istruzione RFRP legge dati del buffer di memoria di un modulo funzione speciale all'interno di una stazione remota collegata a MELSECNET.

Il numero di word dati da leggere è specificato da n2 (serie A = n3). L'indirizzo all'interno del buffer di memoria è specificato a partire da n1 (serie A = n2). L'indirizzo di I/O del modulo funzione speciale collegato è specificato da Un (serie A = n1). I dati letti vengono memorizzati nei registri di comunicazione specificati a partire da d1 (serie A = d) della stazione master.

Al completamento dell'operazione di lettura dalla stazione di I/O remoto, l'operando specificato da d2 viene attivato (solo serie Q).



¹ Stazione host (stazione master)

² Modulo di comunicazione

³ Modulo funzione speciale (stazione target / stazione I/O remoto)

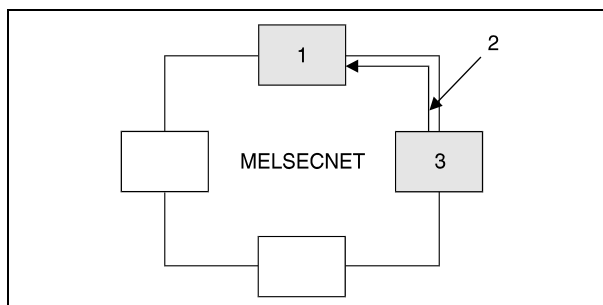
NOTA

Anche se viene seguita solo un'operazione di lettura, l'indirizzo dell'area nei registri di comunicazione in d1 (serie A = d) deve essere compreso nel campo della parametrizzazione MELSECNET della stazione master e remota.

Di seguito vengono descritti gli indirizzi di I/O dei moduli funzione speciali della serie QnA. Queste indicazioni sono valide anche per la serie A, tranne il valore n che deve essere sostituito da n1 (ad es. serie QnA = Y(n+E) ⇒ serie A = Y(n1+E)).

Durante l'esecuzione dell'istruzione RFRP viene attivata l'uscita Y(n+E). X(n+1E) viene attivato al completamento dell'istruzione. Y(n+E) rimane attivo anche dopo il termine dell'esecuzione e deve essere azzerato dal programma di sequenza. L'indirizzamento viene definito automaticamente e non deve essere modificato.

Le operazioni di lettura dalle stazioni di I/O remoto possono essere eseguite da una stazione master collegata a MELSECNET.



¹ Stazione che esegue l'istruzione RFRP (stazione master)

² Operazione di lettura dati dal modulo funzione speciale

³ Stazione I/O remoti

Se l'istruzione RFRP non può essere eseguita a causa di un errore all'interno del modulo funzione speciale indirizzato, viene attivato X(n+1D). In questo caso occorre controllare il modulo interessato. X(n+1D) viene azzerato quando Y(n+D) si attiva.

L'indirizzo dell'instestazione di I/O dei moduli funzione speciali specificati da Un, viene memorizzato con le tre cifre più pesanti in un formato a 4 cifre. Ad esempio, gli indirizzi X/Y0200 sono rappresentati con il valore 20 (solo serie QnA).

NOTA

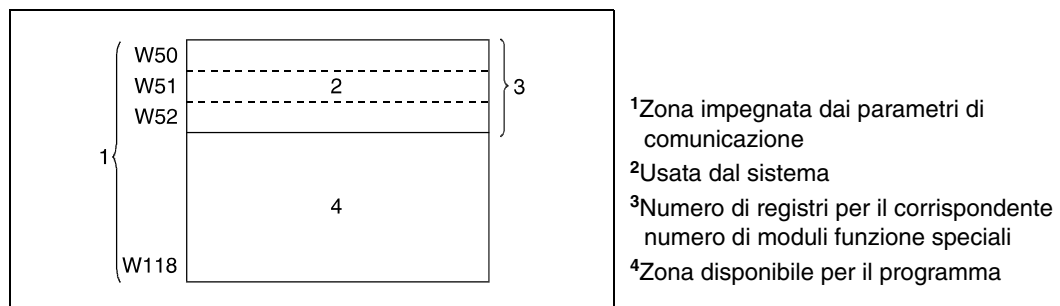
Fare riferimento al manuale del modulo funzione speciale corrispondente per ulteriori informazioni sui campi di indirizzamento validi del buffer di memoria per i moduli funzione speciali specificati da n1 (serie A = n2).

La lunghezza dei dati da leggere (numero word dati) specificata da n2 (serie A = n3) può variare fra 1 e 16.

L'area di indirizzamento del registro di comunicazione in d1 (serie A = d) deve essere compresa nel campo dei parametri di comunicazione della stazione master e della stazione remota.

Il campo del registro di comunicazione Wxxx deve essere differenziato correttamente nelle stazioni master e remota. Il numero di registri di comunicazione usati dal sistema operativo corrisponde al numero di moduli funzione speciali contenuti nelle stazioni remote di una rete. L'area disponibile per la memorizzazione dati è il campo parametrizzato meno il numero di registri di comunicazione usati dal sistema operativo.

L'esempio che segue mostra le diverse aree di un registro di comunicazione. L'area fra stazione master e stazione remota è specificata da W050 a W118 (serie A = W09F) nei parametri. In questa area vengono allocati due moduli funzione speciali in modo che i primi due registri di comunicazione W50 e W51 (2 indirizzi) sono impegnati dal sistema operativo della CPU. L'area disponibile per lo scambio dati è quindi compresa fra W52 e W118 (serie A = W09F).



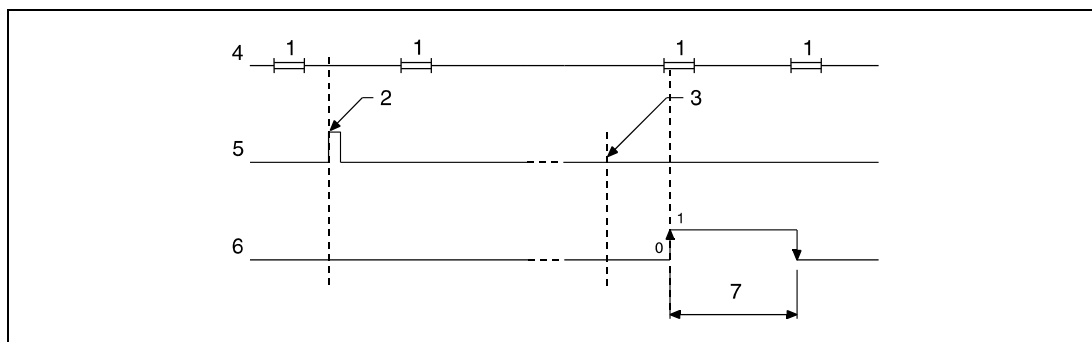
Le istruzioni RFRP e RTOP non possono essere contemporaneamente da più locazioni dallo stesso modulo funzione speciale. L'esecuzione simultanea da due o più locazioni viene evitata da un meccanismo di interblocco fra le due stazioni attive.

Gli ingressi e uscite X(n+1E) e Y(n+E) devono essere programmate come interblocco per assicurare che non possano essere eseguite simultaneamente altre istruzioni RFRP o RTOP.

L'operando di completamento operazione della stazione host (d2) viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione di lettura viene completata. L'operando viene azzerato con la successiva istruzione END (solo serie QnA).

La serie MELSEC A contiene numerosi registri speciali per il trasferimento dati su MELSECNET per registrare diversi stati relativi alla comunicazione. Ad esempio, lo stato delle stazioni di I/O remoto viene registrato tramite i registri speciali da D9228 a D9231. L'accesso ai parametri avviene tramite i relé speciali da M9224 a M9227 (solo serie A).

La figura seguente mostra le operazioni della stazione host durante l'esecuzione di una istruzione RFRP:



- 1 Elaborazione END
- 2 Esecuzione istruzione RFRP
- 3 Completamento operazione
- 4 Programma della stazione host
- 5 Istruzione RFRP
- 6 Operando di completamento della stazione host (d2) attivato al termine dell'operazione (solo serie QnA)
- 7 Una scansione

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- L'indirizzo di I/O specificato da Un (serie A = n1) non corrisponde a una stazione di I/O remoto (serie QnA = codice di errore 4102).
- Il numero di I/O specificato da n1 (serie A = n2) non è l'indirizzo dell'instestazione di I/O di un modulo funzione speciale (serie QnA = codice di errore 4102).
- Il numero di indirizzi specificato da n2 (serie A = n3) supera il campo di indirizzi specificato a partire da d1 (serie A = d, da W0 a W3FF) (serie QnA = codice di errore 4101).
- La rete specificata da Un (serie A = n1) non esiste (codice di errore 2413).
- Il valore specificato per n2 (serie A = n3) supera il campo da 1 a 16 (codice di errore 4100).

Esempio di programma

RFRP (serie A)

Il programma seguente legge i dati da 10 indirizzi in sequenza a partire dall'indirizzo 10, da un modulo funzione speciale (ad es. A68AD). Il modulo è inserito nella seconda stazione remota. Gli indirizzi vanno da 140 a 15F. I dati letti vengono memorizzati nei registri di comunicazione da W52 a W61 della stazione master.

Dopo l'attivazione di X3, M0 si attiva e viene eseguita l'istruzione RFRP. All'inizio del trasferimento dati viene attivato $Y(n+1) = Y14E$. Al completamento del trasferimento viene attivato $X(n+1) = X15E$. L'istruzione RFRP non viene eseguita se è in corso l'esecuzione di un'altra istruzione RFRP o RTOP.

<p>Lista istruzioni MELSEC</p> <pre> MELSEC LD X003 PLS M1 LD M1 SET M0 LD M0 MPS ANI Y14E ANI X15E ANI Y14F ANI X15F RFRP H0140 K10 W052 K10 MPP AND X15E RST M0 RST Y14E END </pre>	<p>Schema a contatti</p>
<p>Lista istruzioni IEC</p> <pre> LD X3 PLS_M M1 LD M1 SET_M M0 LD M0 MPS ANDN Y14E ANDN X15E ANDN Y14F ANDN X15F RFRP_M H0140, K10, W052, K10 MPP AND X15E RST_M M0 RST_M Y14E END_M </pre>	

¹ Esecuzione singola

NOTA

Il contatto corrispondente a M1 deve essere convertito in impulso. Altrimenti l'istruzione RFRP non viene eseguita completamente.

Il contatto corrispondente a M0 deve essere attivato con l'istruzione SET. Se viene programmata una istruzione OUT o PLS invece dell'istruzione SET, si possono verificare errori nell'esecuzione dell'istruzione RFRP.

Per evitare l'esecuzione simultanea di due istruzioni RFRP, si deve realizzare un interblocco per mezzo dell'uscita Y41E e dell'ingresso X15E.

Se la stazione viene indirizzata da una istruzione RTOP contenuta nello stesso programma, anche l'uscita Y14F e l'ingresso X15F devono essere programmati come interblocco.

8.7.6 RTOP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●	●	

Operandi MELSEC A

	Operandi utilizzabili															Definizione cifra	Numero di passi	Indice	Riporto	Errore			
	Operandi a bit					Operandi a word (16-bit)					Costanti		Puntatore		Livello								
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z				V	K	H (16#)	P	I
n1																●	●						
n2																●	●						
s										●													●
n3																●	●						●


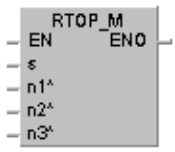

¹ Consultare il capitolo "Programmazione con CPU AnA, AnAS e AnU" di questo manuale per il corrispondente numero di passi.

Operandi MELSEC Q


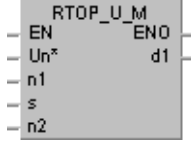

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
n1	●	●	●	—	—	—	—	●	—	SM0	9
s	—	● ¹	—	—	—	—	—	—	—		
n2	●	●	●	—	—	—	—	—	—		
d	●	●	●	—	—	—	—	●	—		

¹ Solo registri di comunicazione

serie A
GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
 RTOP n1 n2 s n3		 RTOP_M s , n1 , n2 , n3

Serie QnA
GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
 G.RTOP Un n1 s n2 d1		 RTOP_U_M Un , n1 , s , n2 , d1

Variabili

Dati impostati		Significato	Tipo dati
Serie A	Serie QnA		
n1	Un	Indirizzo intestazione I/O del modulo funzione speciale nella stazione I/O remoto ● ¹	BIN 16-bit
n2	n1	Primo indirizzo del buffer di memoria del modulo funzione speciale che contiene i dati scritti.	
s	s	Indirizzo del primo registro di comunicazione della stazione host che contiene i dati da scrivere.	Operando numero
n3	n2	Lunghezza dati inviati.	BIN 16-bit
	d	Operando attivato per una scansione al completamento dell'istruzione.	Bit

NOTA

- ¹ L'indirizzo di intestazione di I/O dell'unità di rete per la stazione host deve essere nel campo da 0 a FE_H. Notare, che il compilatore considera Un come valore esadecimale. Un numero decimale viene convertito automaticamente in valore esadecimale.

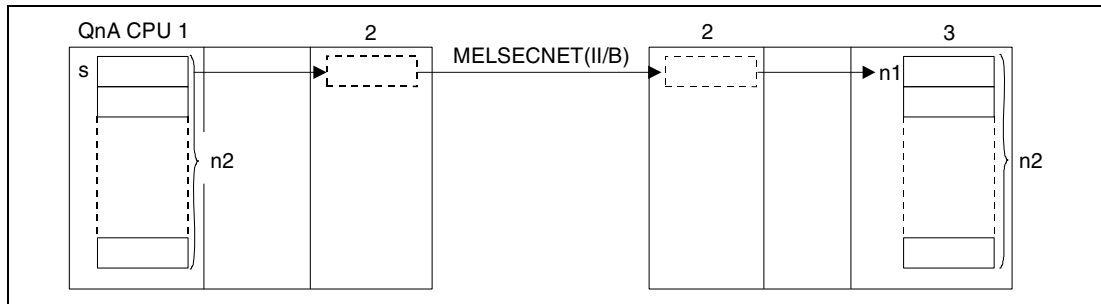
Funzioni Scrittura dati su una stazione remota

RTOP Istruzione di scrittura

L'istruzione RTOP scrive dati nel buffer di memoria di un modulo funzione speciale all'interno di una stazione remota collegata a MELSECNET.

Il numero di word dati da scrivere è specificato da n2 (serie A = n3). L'indirizzo all'interno del buffer di memoria è specificato a partire da n1 (serie A = n2). L'indirizzo di I/O del modulo funzione speciale collegato è specificato da Un (serie A = n1). I dati da scrivere sono contenuti nella stazione master all'interno dei registri di comunicazione specificati da s.

Al completamento dell'operazione di scrittura nella stazione di I/O remoto, l'operando specificato da d viene attivato (solo serie QnA).



¹ Stazione host (stazione master)

² Modulo di comunicazione

³ Modulo funzione speciale (stazione target / stazione I/O remoto)

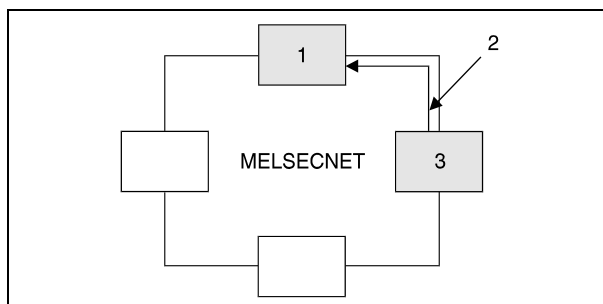
NOTA

Anche se viene seguita solo un'operazione di scrittura, l'indirizzo dell'area nei registri di comunicazione in s deve essere compreso nel campo della parametrizzazione MELSECNET della stazione master e remota.

Di seguito vengono descritti gli indirizzi di I/O dei moduli funzione speciali della serie QnA. Queste indicazioni sono valide anche per la serie A, tranne il valore n che deve essere sostituito da n1 (ad es. serie QnA = Y(n+E) ⇒ serie A = Y(n1+F)).

Durante l'esecuzione dell'istruzione RTOP viene attivata l'uscita Y(n+F). X(n+1F) viene attivato al completamento dell'istruzione.. Y(n+F) rimane attivo anche dopo il termine dell'esecuzione e deve essere azzerato dal programma di sequenza. L'indirizzamento viene definito automaticamente e non deve essere modificato.

Le operazioni di scrittura sulle stazioni di I/O remoto possono essere eseguite da una stazione master collegata a MELSECNET.



¹Stazione che esegue l'istruzione RTOP (stazione master)

² Operazione di scrittura dati nel modulo funzione speciale

³ Stazione I/O remoti

Se l'istruzione RTOP non può essere eseguita a causa di un errore all'interno del modulo funzione speciale indirizzato, viene attivato X(n+1D). In questo caso occorre controllare il modulo interessato. X(n+1D) viene azzerato quando Y(n+D) si attiva.

L'indirizzo dell'instestazione di I/O dei moduli funzione speciali specificati da Un, viene memorizzato con le tre cifre più pesanti in un formato a 4 cifre. Ad esempio, gli indirizzi X/Y0200 sono rappresentati con il valore 20 (solo serie QnA).

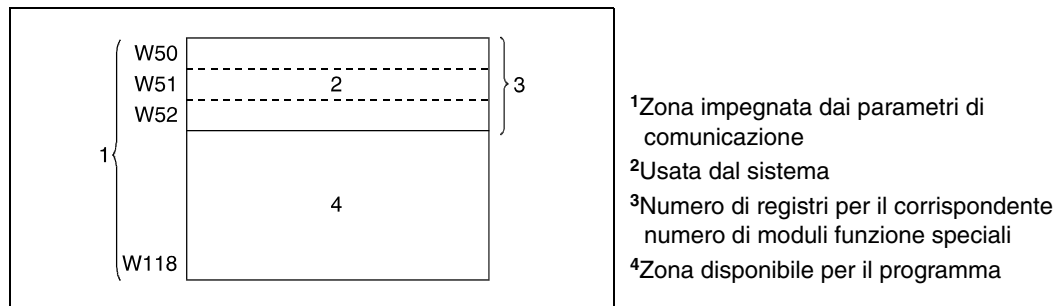
NOTA

Fare riferimento al manuale del modulo funzione speciale corrispondente per ulteriori informazioni sui campi di indirizzamento validi del buffer di memoria per i moduli funzione speciali specificati da n1 (serie A = n2).

La lunghezza dei dati da scrivere (numero word dati) specificata da n2 (serie A = n3) può variare fra 1 e 16.

Il campo del registro di comunicazione Wxxx deve essere differenziato correttamente nelle stazioni master e remota. Il numero di registri di comunicazione usati dal sistema operativo corrisponde al numero di moduli funzione speciali contenuti nelle stazioni remote di una rete. L'area disponibile per la memorizzazione dati è il campo parametrizzato meno il numero di registri di comunicazione usati dal sistema operativo.

L'esempio che segue mostra le diverse aree di un registro di comunicazione. L'area fra stazione master e stazione remota è specificata da W050 a W118 (serie A = W09F) nei parametri. In questa area vengono allocati due moduli funzione speciali in modo che i primi due registri di comunicazione W50 e W51 (2 indirizzi) sono impegnati dal sistema operativo della CPU. L'area disponibile per lo scambio dati è quindi compresa fra W52 e W118 (serie A = W09F).



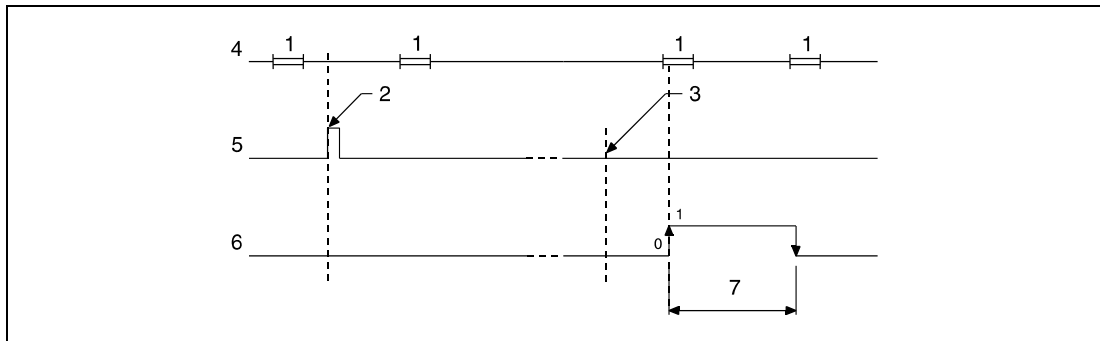
Le istruzioni RTOP e RFRP non possono essere contemporaneamente da più locazioni dallo stesso modulo funzione speciale. L'esecuzione simultanea da due o più locazioni viene evitata da un meccanismo di interblocco fra le due stazioni attive.

Gli ingressi e uscite X(n+1F) e Y(n+F) devono essere programmate come interblocco per assicurare che non possano essere eseguite simultaneamente altre istruzioni RTOP o RFRP.

L'operando di completamento operazione della stazione host (d) viene attivato con l'esecuzione dell'istruzione END nella scansione del programma in cui l'operazione di scrittura viene completata. L'operando viene azzerato con la successiva istruzione END (solo serie QnA).

La serie MELSEC A contiene numerosi registri speciali per il trasferimento dati su MELSECNET per registrare diversi stati relativi alla comunicazione. Ad esempio, lo stato delle stazioni di I/O remoto viene registrato tramite i registri speciali da D9228 a D9231. L'accesso ai parametri avviene tramite i relé speciali da M9224 a M9227 (solo serie A).

La figura seguente mostra le operazioni della stazione host durante l'esecuzione di una istruzione RTOP:



¹ Elaborazione END

² Esecuzione istruzione RTOP

³ Completamento operazione

⁴ Programma della stazione host

⁵ Istruzione RTOP

⁶ Operando di completamento della stazione host (d) attivato al termine dell'operazione (solo serie QnA)

⁷ Una scansione

Errori di esecuzione

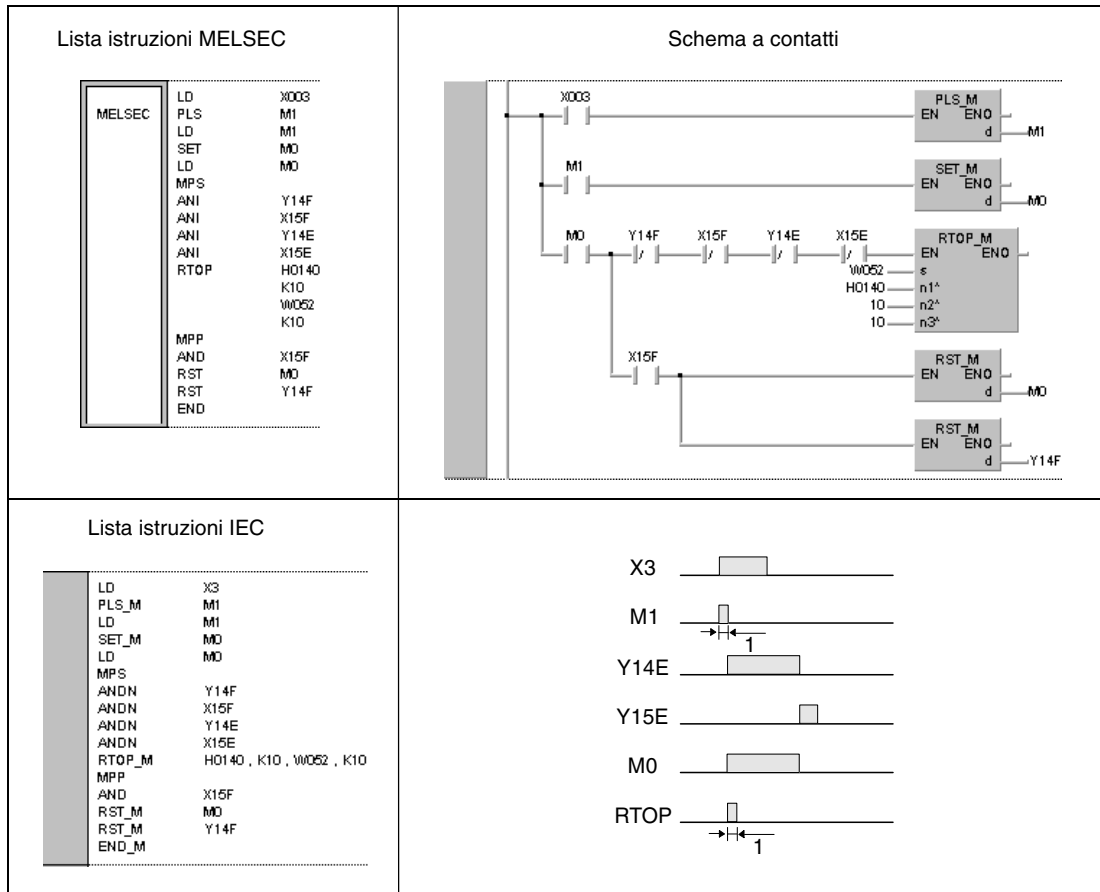
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- L'indirizzo di I/O specificato da Un (serie A = n1) non corrisponde a una stazione di I/O remoto (serie QnA = codice di errore 4102).
- Il numero di I/O specificato da n1 (serie A = n2) non è l'indirizzo dell'instestazione di I/O di un modulo funzione speciale (codice di errore 4102).
- Il numero di indirizzi specificato da n2 (serie A = n3) supera il campo di indirizzi specificato a partire da d1 (serie A = d, da W0 a W3FF) (serie QnA = codice di errore 4101).
- La rete specificata da Un (serie A = n1) non esiste (codice di errore 2413).
- Il valore specificato per n2 (serie A = n3) supera il campo da 1 a 16 (codice di errore 4100).

Esempio di programma RTOP (serie A)

Il programma seguente scrive dati contenuti nei registri di comunicazione da W52 a W61 della stazione master, in 10 indirizzi in sequenza di un modulo funzione speciale (ad es. A68AD). Il modulo è inserito nella seconda stazione remota. Gli indirizzi vanno da 140 a 15F. I dati scritti vengono memorizzati nell'area a partire dall'indirizzo 10.

Dopo l'attivazione di X3, M0 si attiva e viene eseguita l'istruzione RTOP. All'inizio del trasferimento dati viene attivato $Y(n1+F) = Y14F$. Al completamento del trasferimento viene attivato $X(n1+1F) = X15F$. L'istruzione RTOP non viene eseguita se è in corso l'esecuzione di un'altra istruzione RTOP o RFRP. Al termine del trasferimento M0 e Y14F vengono azzerati nella prosecuzione del programma.



¹ Esecuzione singola

NOTA

Il contatto corrispondente a M1 deve essere convertito in impulso. Altrimenti l'istruzione RTOP non viene eseguita completamente.

Il contatto corrispondente a M0 deve essere attivato con l'istruzione SET. Se viene programmata una istruzione OUT o PLS invece dell'istruzione SET, si possono verificare errori nell'esecuzione dell'istruzione RTOP.

Per evitare l'esecuzione simultanea di due istruzioni RTOP, si deve realizzare un interblocco per mezzo dell'uscita Y41F e dell'ingresso X15F.

Se la stazione viene indirizzata da una istruzione RFRP contenuta nello stesso programma, anche l'uscita Y14E e l'ingresso X15E devono essere programmati come interblocco.

8.8 Lettura e scrittura di informazioni di instradamento

Queste istruzioni leggono e scrivono le informazioni di instradamento (routing). I parametri di instradamento comprendono il numero di rete e di stazione della stazione ripetitrice ed il numero di stazione della stazione di routing.

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Lettura dati di instradamento	Z.RTREAD	RTREAD_M
	ZP.RTREAD	RTREADP_M
Scrittura dati di instradamento	Z.RTWRITE	RTWRITE_M
	ZP.RTWRITE	RTWRITEP_M

8.8.1 RTREAD

CPU

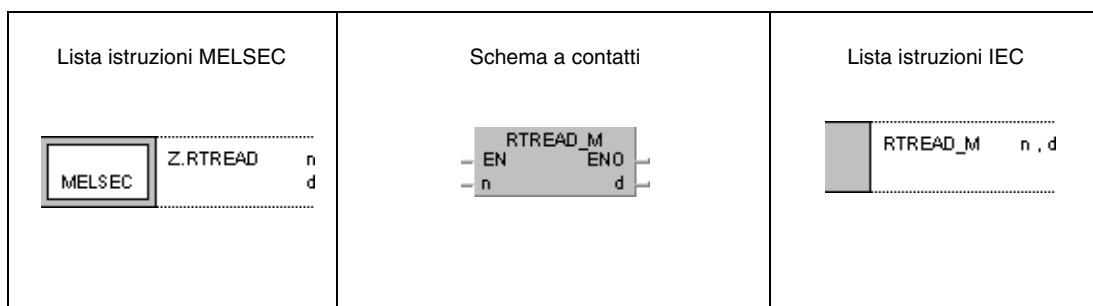
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

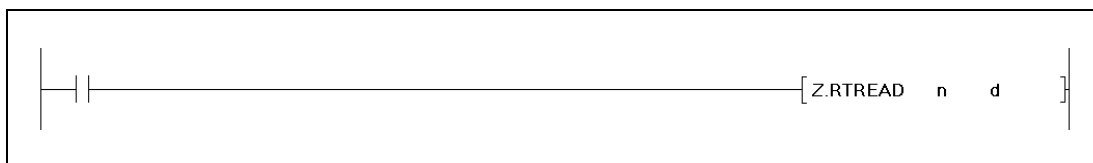
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
n	—	●	●	—	—	—	—	●	—	SM0	7
d	—	●	●	—	—	—	—	—	—		

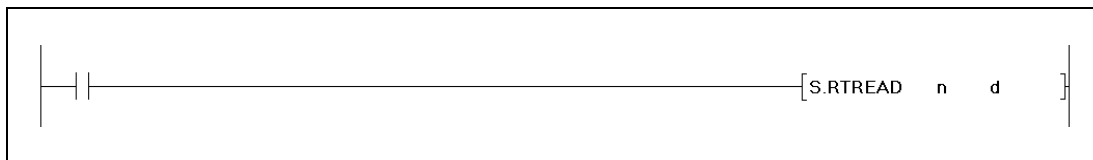
GX IEC Developer (CPU QnA)



GX Developer (CPU QnA)



GX Developer (CPU System Q)



Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
n	Rete di destinazione della trasmissione (da 1 a 239).	BIN 16-bit	ANY16
d	Indirizzo del primo operando che memorizza i dati di instradamento.	Indirizzo operando	Array [1.0.3] di ANY16

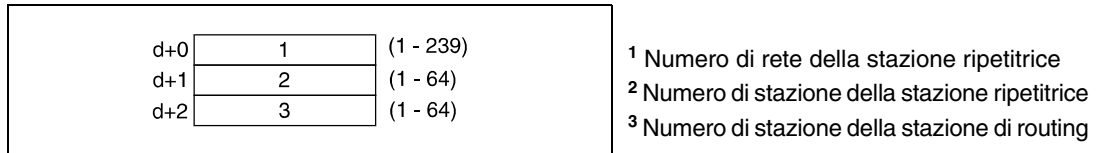
Funzioni Lettura dati di instradamento

RTREAD Istruzione di lettura

L'istruzione RTREAD legge i dati di instradamento dalla rete di destinazione indicata da n. I dati di instradamento sono memorizzati nei parametri di instradamento. Le informazioni di instradamento lette sono memorizzate a partire da d+0 (Array_d[1]).

Se nessun dato viene specificato per la trasmissione, il valore 0 viene scritto negli operandi a partire da d (da Array_d[1] a Array_d[3]).

La figura seguente mostra il contenuto specificato a partire da d+0 (Array_d[1]):



Errori di esecuzione

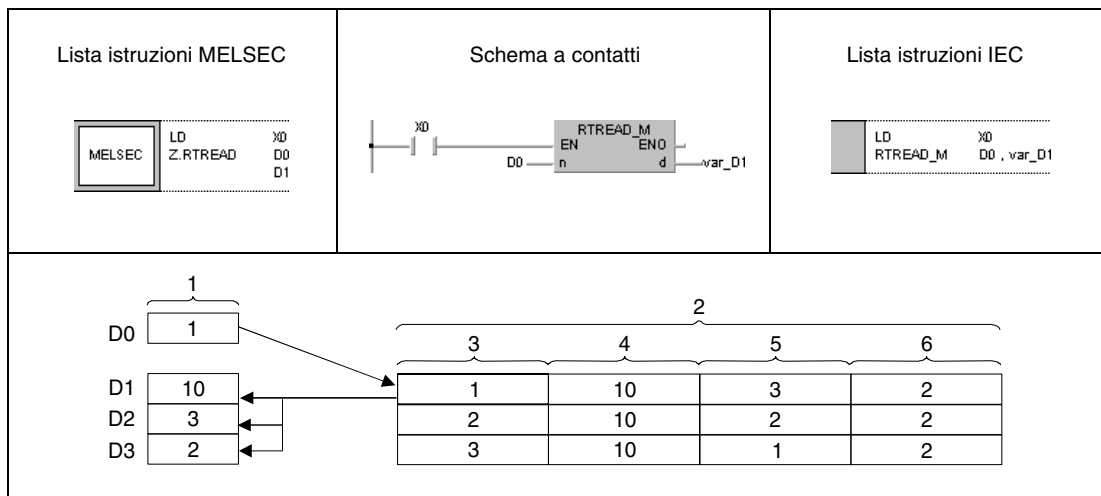
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- La lunghezza dei dati specificata da n non è nel campo da 1 a 239 (codice di errore 4100).

Esempio di programma

Z.RTREAD

Se X0 è attivo, il programma che segue legge le informazioni di instradamento dalla rete (11) specificata da D0 e memorizza i dati da D1 a D3 (da var_D1[1] a var_D1[3]).



- ¹ Funzionamento
- ² Contenuto dei parametri di instradamento
- ³ Numero di rete della rete di destinazione della trasmissione
- ⁴ Numero di rete della stazione ripetitrice
- ⁵ Numero di stazione della stazione ripetitrice
- ⁶ Numero di stazione della stazione di routing

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

8.8.2 RTWRITE

CPU

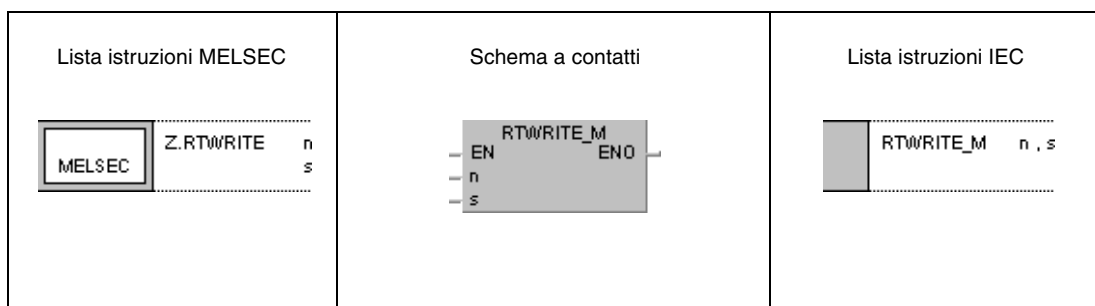
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

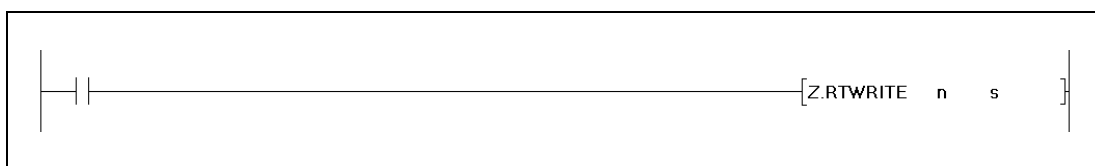
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
n	—	●	●	—	—	—	—	●	—	SM0	8
s	—	●	●	—	—	—	—	—	—		

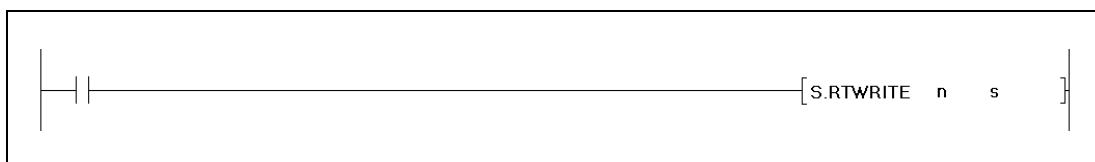
GX IEC Developer (CPU QnA)



GX Developer (CPU QnA)



GX Developer (CPU System Q)



Variabili

Dati impostati	Significato	Tipo dati	
		MELSEC	IEC
n	Rete di destinazione della trasmissione (da 1 a 239).	BIN 16-bit	ANY16
s	Indirizzo del primo operando che contiene i dati di instradamento da scrivere.	Indirizzo operando	Array [1.0.3] di ANY16

Funzioni Scrittura dati di instradamento

RTWRITE Istruzione di scrittura

L'istruzione RTWRITE legge i dati di instradamento dalla rete di destinazione indicata da n. I dati di instradamento sono memorizzati nei parametri di instradamento. Le informazioni di instradamento lette sono memorizzate a partire da s+0 (Array_s[1]).

Se il dato della rete di destinazione è impostato nei parametri di routing, esso viene usato per rinfrescare i dati contenuti a partire da s+0 (Array_s[1]).

La figura seguente mostra il contenuto specificato a partire da s+0 (Array_s[1]):



Errori di esecuzione

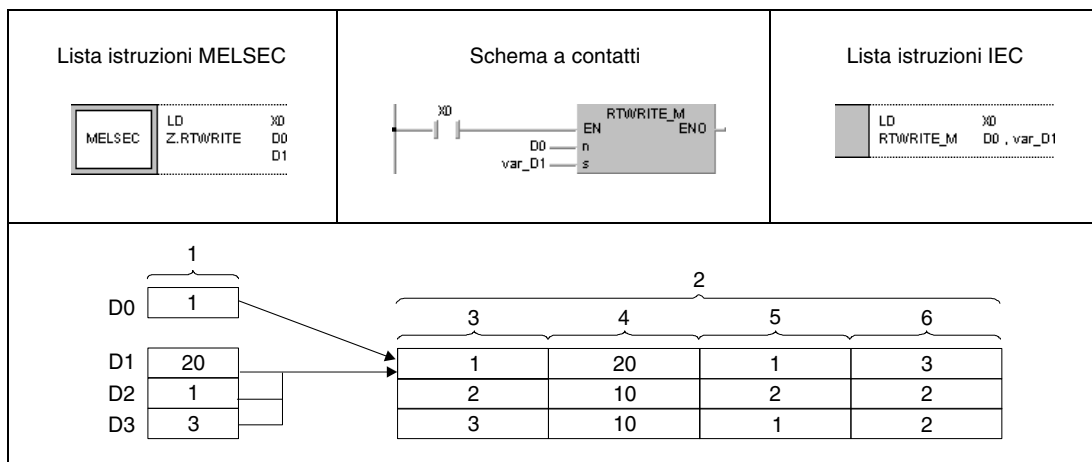
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- La lunghezza dei dati specificata da n non è nel campo da 1 a 239 (codice di errore 4100).
- Il dato specificato da s supera i campi ammessi (codice di errore 4100).

Esempio di programma

Z.RTWRITE

Se X0 è attivo, il programma seguente scrive le informazioni di instradamento contenute da D1 a D3 (da var_D1[1] a var_D1[3]) come parametri di instradamento per la rete (1) specificata da D0.



- ¹ Funzionamento
- ² Contenuto dei parametri di instradamento
- ³ Numero di rete della rete di destinazione della trasmissione
- ⁴ Numero di rete della stazione ripetitrice
- ⁵ Numero di stazione della stazione ripetitrice
- ⁶ Numero di stazione della stazione di routing

NOTA

Questi programmi di esempio non funzionano senza una definizione delle variabili nello header dell'unità organizzativa di programma (POU). Provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

9 Istruzioni per CPU System Q

Le istruzioni seguenti sono disponibili per le CPU del System Q.

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Lettura informazioni modulo	UNIRD	UNIRD_M
	UNIRDP	UNIRDP_M
Istruzioni di debug e diagnosi guasti	TRACE	TRACE_M
	TRACER	TRACER_M
Scrittura e lettura da file	FWRITE	FWRITE_M
	FREAD	FREAD_M
Istruzioni di programma	PLOADP	PLOADP_M
	PUNLOADP	PUNLOADP_M
	PSWAPP	PSWAPP_M
Istruzioni per trasferimento dati	RBMV	RBMV_M
	RBMV_P	RBMV_P_M

Le seguenti istruzioni, utilizzate nei sistemi multi CPU, sono state aggiunte alle CPU System Q dalla versione funzionale B (Q02CPU, Q02HCPU, Q06HCPU, Q12HCPU e Q25HCPU):

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Scrive nella memoria condivisa della CPU della stazione host	S.TO	TO_S_M
	S.TOP	TO_SP_M
Lettura dalla memoria condivisa di una diversa stazione	FROM	FROM_M
	FROM_P	FROM_P_M

9.1 Lettura informazioni modulo

9.1.1 UNIRD, UNIRDP

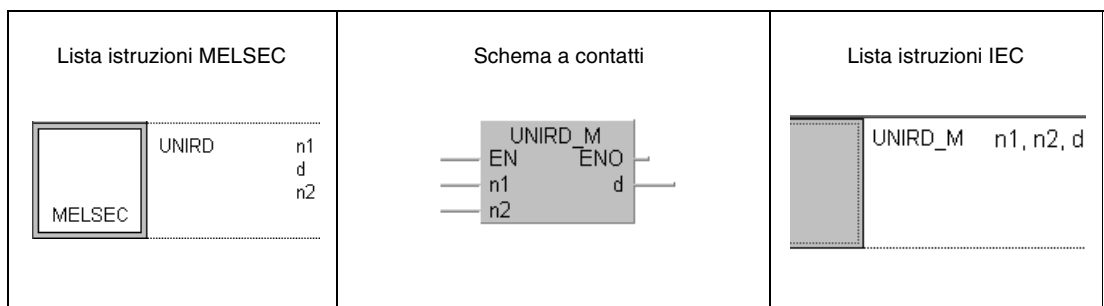
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					●

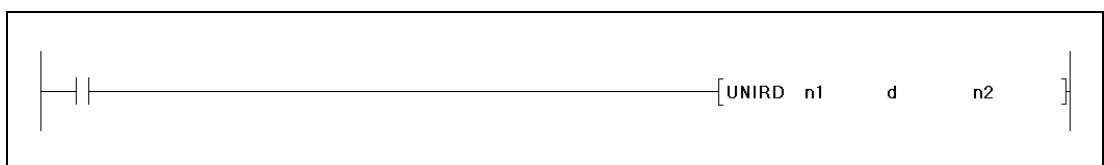
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo Funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
n1	●	●	●	—	—	—	—	●	—	SM0	4
d	—	●	●	—	—	—	—	—	—		
n2	●	●	●	—	—	—	—	●	—		

GX IEC Developer



GX Developer



Variabili

Operando	Significato	Tipo dati
n1	Valore ottenuto dividendo per 16 l'indirizzo di I/O del modulo da cui viene letta l'informazione (da 0 a FF _H).	BIN 16-bit
d	Indirizzo iniziale dell'operando che contiene le informazioni del modulo.	Nome operando
n2	Numero delle informazioni modulo lette (da 0 a 256).	BIN 16-bit

Funzioni Lettura informazioni modulo

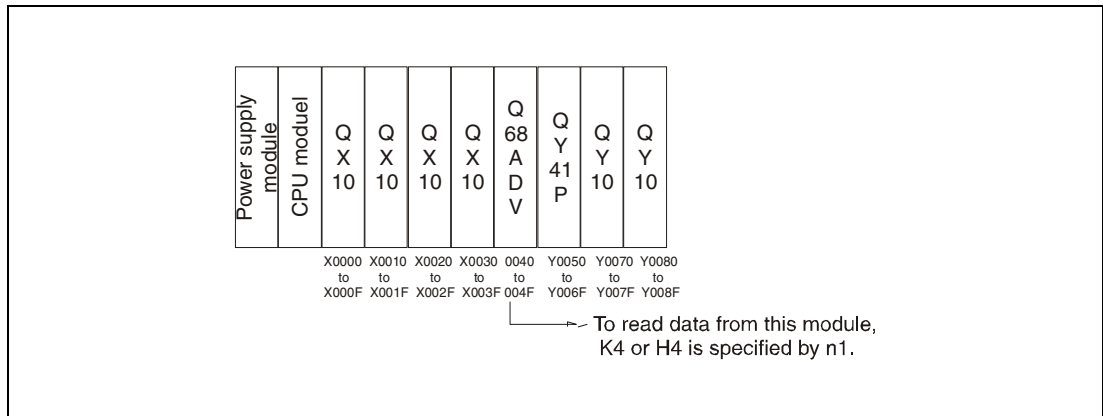
UNIRD Istruzione di lettura

L'istruzione UNIRD legge le informazioni del modulo a partire dall'indirizzo di I/O iniziale specificato da n1, e memorizza i dati a partire dall'indirizzo specificato da d. Il numero di dati è specificato da n2. Il valore di n1 viene calcolato dividendo per 16 il numero di I/O iniziale del modulo.

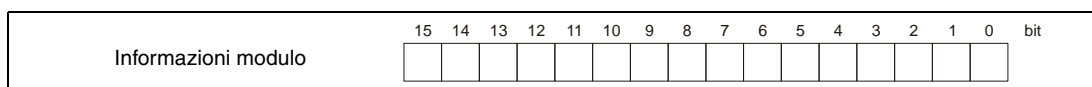
Con l'istruzione UNIRD è possibile leggere gli stati dei moduli effettivamente installati, invece del tipo di moduli designati dall'assegnazione di I/O.

NOTA

Il valore n1 è composto dalle tre cifre più pesanti del numero di I/O iniziale dello slot da cui si leggono le informazioni del modulo. L'indirizzo iniziale di I/O viene espresso come notazione esadecimale a 4 cifre.



I dettagli delle informazioni del modulo sono descritti di seguito:



Bit	Caratteristica	Significato
0	Numero di punti di I/O	000: 16
1		001: 32
2		010: 48
3	Tipo di modulo	011: 64
4		100: 128
5		101: 256
6	Requisiti di alimentazione esterna (per espansione futura)	110: 512
7	Stato fusibile	111: 1024
8	Non usato	000: Modulo ingressi 001: Modulo di uscita 010: modulo I/O misto 011: Modulo funzione intelligente
9	Stato errore leggero/medio	ON: Alimentatore esterno collegato OFF: Alimentatore esterno non collegato
10	Stato errore modulo	ON: Fusibile interrotto OFF: Normale, fusibile non interrotto
11		ON: Si è verificato un errore leggero/medio OFF: Normale
12	Stato standby modulo	00: Nessun errore 01: Errore leggero 10: Errore medio 11: Errore grave
13	Non usato	ON: Normale OFF: Errore modulo attivo
14	Modulo A-/Q	ON: Moduli installati OFF: Nessun modulo installato
15	Stato installazione modulo	ON: il modulo è un modulo serie A OFF: Il modulo è un modulo serie Q

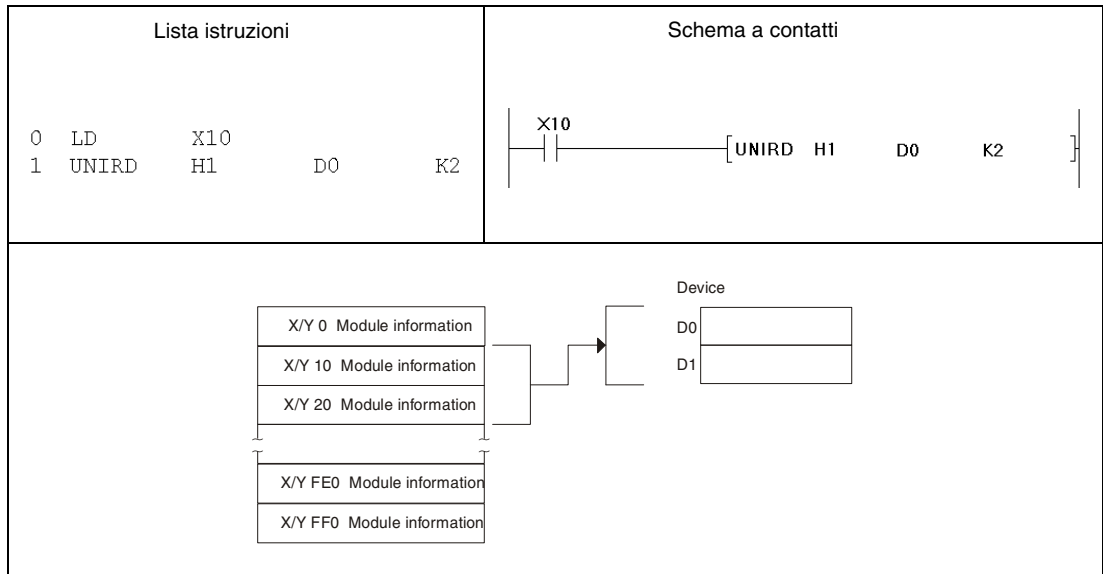
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- È stato specificato un valore fuori del campo ammesso (da 0 a FF_H) in n1 (codice di errore 4100).
- È stato specificato un valore fuori del campo ammesso (da 0 a FF_H) in n2 (codice di errore 4100).
- La somma di n1 e n2 è maggiore di 256 (codice di errore 4100).

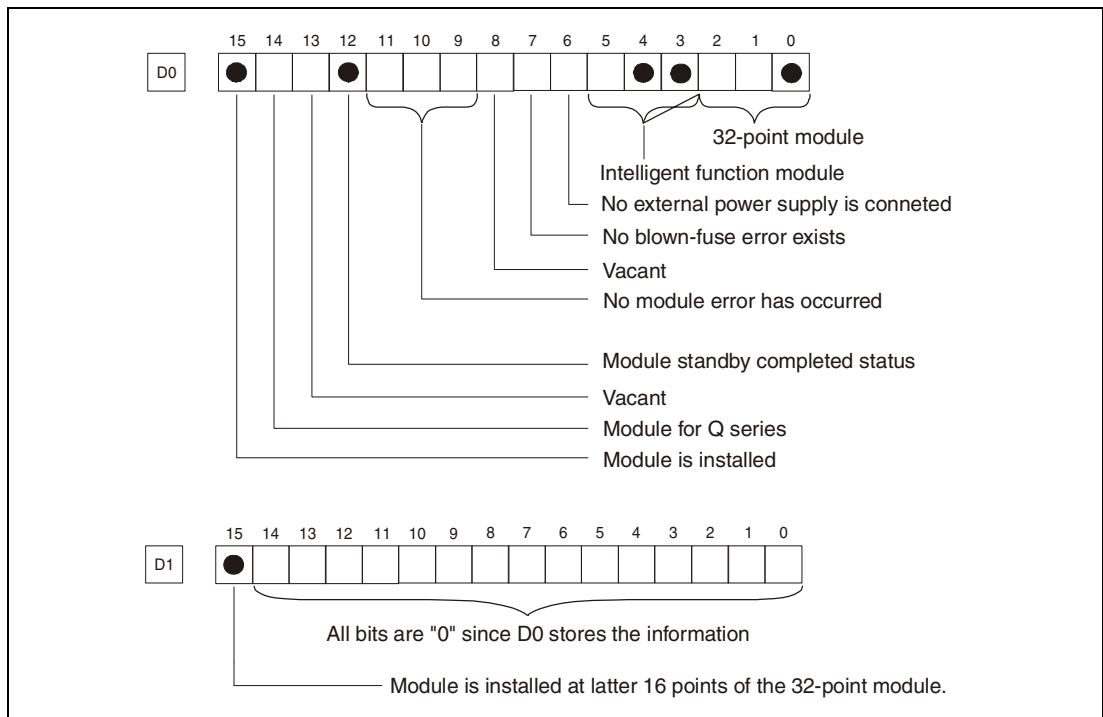
Programma Esempio UNIRD

Il programma seguente memorizza le informazioni dei moduli i cui indirizzi di I/O sono compresi fra 10_H e 2F_H nei registri D0 e D1, se X10 viene attivato (ON).

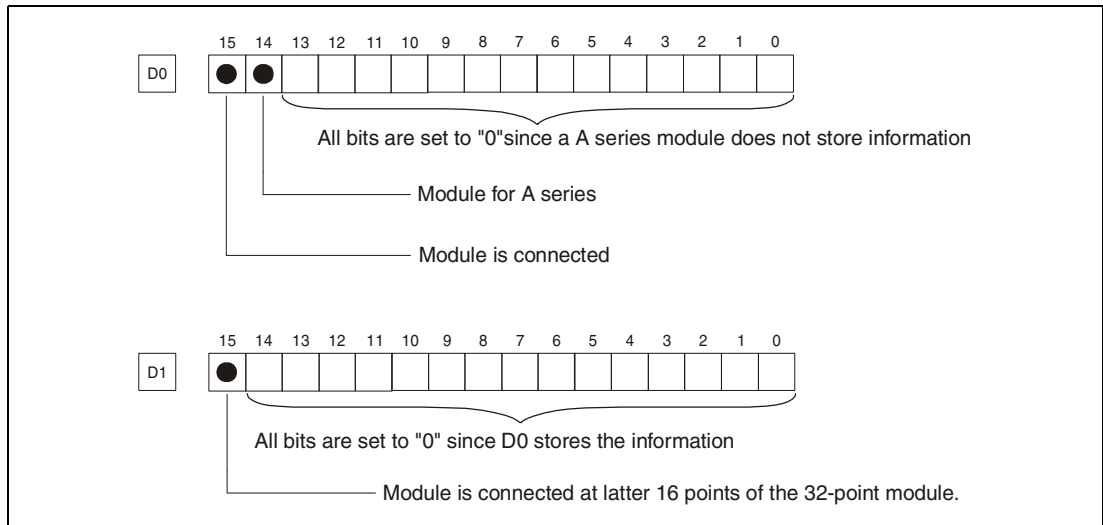


In questo programma di esempio le informazioni del modulo sono memorizzate in D0 e D1. I risultati indicati valgono

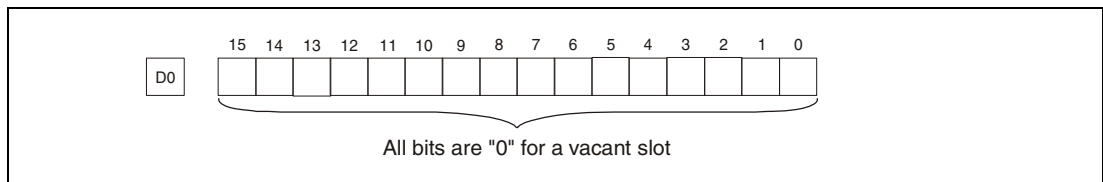
- per un modulo funzione intelligente da 32 punti del System Q. Nei moduli da 48 o 64 punti lo stesso contenuto memorizzato in D1 viene inserito anche in D2 o in D2 e D3 rispettivamente.



- Per un modulo funzione intelligente da 32 punti del System Q. Nei moduli da 48 o 64 punti lo stesso contenuto memorizzato in D1 viene inserito anche in D2 o in D2 e D3 rispettivamente.



- Informazioni modulo per slot vuoto:



9.2 Istruzioni per debug e diagnosi guasti

9.2.1 TRACE, TRACER

CPU

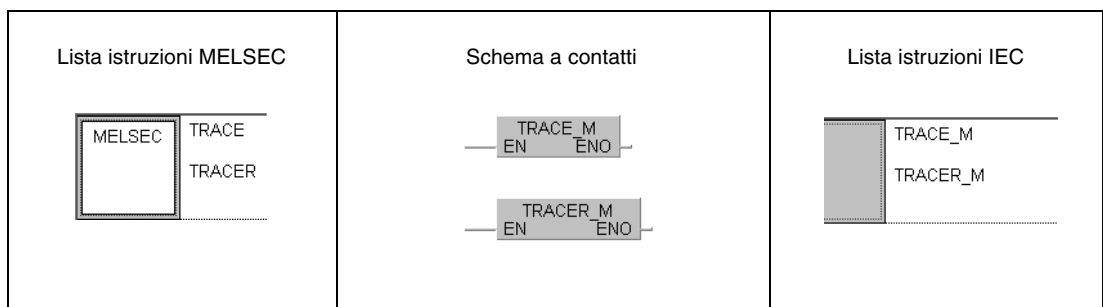
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

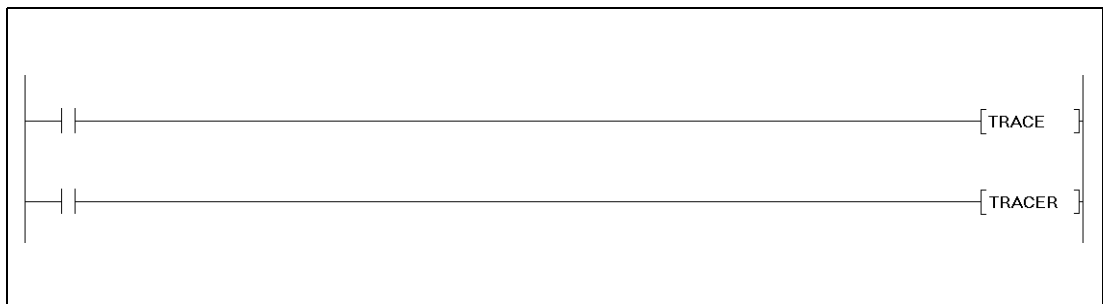
Operandi MELSEC Q

Operandi utilizzabili										Errore	Numero di passi
Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo Funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro			
Bit	Word		Bit	Word							
—	—	—	—	—	—	—	—	—	—	—	1

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
—	—	—

Funzioni **Set / reset traccia****TRACE** **Attiva traccia**

L'istruzione TRACE memorizza i dati di traccia designati da un dispositivo periferico, nel file di traccia di numero designato della memory card, se SM800, SM801 e SM802 sono attivi. Durante l'esecuzione dell'istruzione TRACE, SM803 viene attivato. Il campionamento viene ripetuto per il numero specificato di campionamenti dopo l'istruzione TRACE, poi i dati vengono memorizzati e la traccia viene arrestata.

Il campionamento viene arrestato se SM801 viene disattivato durante l'esecuzione della traccia.

Al termine dell'esecuzione dell'istruzione TRACE, con traccia completata, SM805 viene attivato.

Durante l'esecuzione dell'istruzione TRACE non sono possibili altre istruzioni TRACE, che vengono ignorate. Al termine dell'esecuzione dell'istruzione TRACE, l'istruzione è nuovamente abilitata.

TRACER **Reset traccia**

L'istruzione TRACER disattiva l'istruzione TRACE ed azzerava i flag da SM803 a SM805. Al termine dell'esecuzione dell'istruzione TRACER, l'istruzione TRACE è nuovamente abilitata.

NOTA

Fare riferimento al Manuale Utente delle CPU System Q (modo Q) (Funzioni/fondamenti di programmazione) per ulteriori informazioni sulla funzionalità di traccia.

Fare riferimento ai manuali di istruzione di GX Developer e GX IEC Developer per l'esecuzione di tracce con dispositivi periferici.

Esempio di programma**TRACE, TRACER**

Il programma seguente esegue l'istruzione TRACE quando X0 è attivo. All'attivazione di X1, l'istruzione TRACE viene disattivata dall'istruzione TRACER.

Lista istruzioni	Schema a contatti
0 LD X0	
1 TRACE	
2 LD X1	
3 TRACER	

9.3 Scrittura e lettura da file

9.3.1 FWRITE

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					● ¹

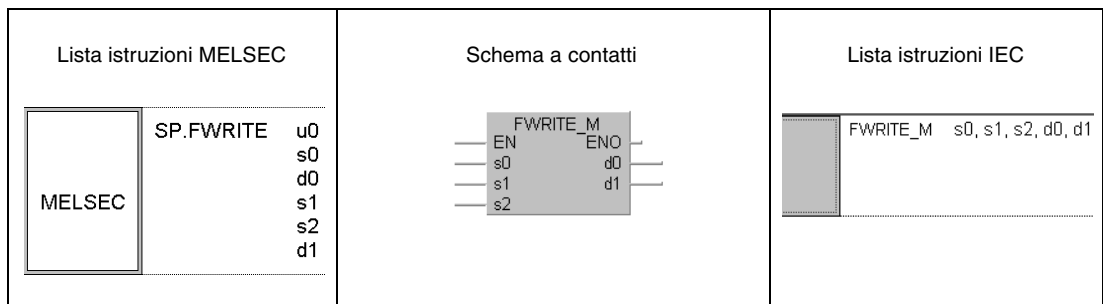
¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

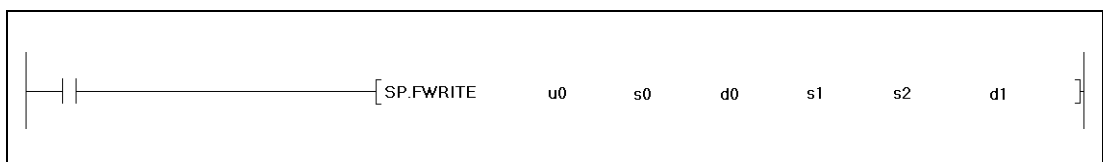
	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s0	●	●	●	—	—	—	—	●	—	SM0	11
d0	—	●	●	—	—	—	—	—	—		
s1	—	●	●	—	—	—	—	—	—		
s2	—	●	●	—	—	—	—	—	●		
d1	●*	●*	●*	—	—	—	—	—	—		

* Non possono essere usati operandi locali o operandi riservati per il programma.

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato		Impostazione Campo	Impostato da	Tipo dati
u0	Dummy		—	—	
s0	Indicazione drive. Può essere specificato solo il drive della scheda ATA (drive 2). Non si possono specificare memory card (ROM) o RAM/ROM standard.		2	Utente	BIN 16-bit
d0	Indirizzo del primo operando che memorizza i dati di controllo. Sono necessari i dati di controllo seguenti.				
	Dati impostati	Caratteristica	Significato/dato impostato	Impostazione Campo	Impostato da
	(d0)	Tipo esecuzione	Specifica il tipo di esecuzione: 0000 _H : Scrittura dati binari 0100 _H : Scrittura dopo conversione in formato CSV	0000 _H 0100 _H	Utente
	(d0)+1	(Riservato)	Usato dal sistema	—	Sistema
	(d0)+2	Esito scrittura (numero dati scritti)	Contiene il numero dei dati effettivamente scritti. L'unità usata per il valore viene definita dalla designazione a word o a byte.	—	Sistema
	(d0)+3	Non utilizzato	—	—	—
	(d0)+4 (d0)+5	Posizione nel file	Imposta la posizione di inizio scrittura all'interno del file quando sono stati selezionati dati binari (d0 = 0000 _H). 00000000 _H : Dall'inizio del file da 00000001 _H a FFFFFFFF _H : Dall'indirizzo specificato. L'unità usata per il valore viene definita dalla designazione a word o a byte. FFFFFFF _H : Sommato alla fine del file. Se è stata selezionata la scrittura dopo la conversione in formato CSV (d0 = 0100 _H): Nelle CPU con numero di serie "01111" o precedente nelle 5 cifre più pesanti, sempre impostato su inizio file (00000000 _H). Nelle CPU con numero di serie "01112" o successivo, imposta la posizione nel file da 00000000 _H a FFFFFFFF _H : Dall'inizio del file. FFFFFFF _H : Sommato alla fine del file.	00000000 _H a FFFFFFF _H	Utente
	(d0)+6	Numero di colonne	Imposta il numero di colonne per la scrittura dati in formato CSV. 0: : Nessuna colonna impostata. I dati sono rappresentati in una sola riga. > 0 : I dati sono rappresentati nel numero specificato di colonne	da 0 a 65535	Utente
	(d0)+7	Indicazione word/byte	0: Word 1: Byte	0, 1	Utente

Variabili

Dati impostati	Significato			Impostazione Campo	Impostato da	Tipo dati
s1	Indirizzo del primo operando che memorizza il nome file.					
	Dati impostati	Caratteristica	Significato/dato impostato	Impostazione Campo	Impostato da	BIN 16-bit
	da (s1)+1 a (s1)+n	Nome file	Il nome file è composto da un massimo di 8 caratteri + punto + estensione (ad esempio: ABD.BIN). L'estensione può essere omessa. In questo caso anche il punto („.“) può essere omesso. Se vengono usati più di 8 caratteri, l'estensione viene ignorata indipendentemente dalla sua presenza. Le estensioni „BIN“ o „CSV“ vengono assegnate automaticamente.	Stringa caratteri	Utente	
s2	Indirizzo del primo operando che memorizza i dati.					
	Dati impostati	Elemento	Significato/dato impostato	Impostazione Campo	Impostato da	BIN 16-bit
	(s2)	Numero di dati da scrivere.	Specifica il numero di dati da scrivere (in unità word). Questo numero deve essere espresso in word anche se è stato selezionato byte in (d0)+7.	da 1 a 480	Utente	
da (s2)+1 a (s2)+n	Dati da scrivere	Dati che devono essere scritti.	0000 _H a FFFF _H			
d1	Operando a bit attivato al termine dell'esecuzione dell'istruzione FWRITE. In caso di errore viene attivato (d1)+1.					
	Dati impostati	Caratteristica	Significato/dato impostato	Impostazione Campo	Impostato da	Bit
	(d1)	Segnale di completamento	Indica lo stato di completamento dell'istruzione FWRITE ON: Completato OFF: Non completato	—	Sistema	
(d1)+1	Segnale completamento con errore	Indica se l'istruzione FWRITE si è conclusa normalmente o con errore. ON: Completamento con errore OFF: Completamento normale	—			

NOTA

I dati scritti in formato CSV sono espressi come valore decimale dal software di programmazione. Ad esempio, il carattere „A“ (41_H) è scritto come 65. Il campo valori ammesso è da -32768 a 32767.

Funzioni Scrittura dati nel file designato

FWRITE Scrittura dati

L'istruzione FWRITE scrive un numero specificato di dati nella scheda ATA. L'utente può scegliere se scrivere i dati direttamente in binario, senza alcuna conversione, oppure se convertirli in formato CSV prima della scrittura.

L'operando a bit del segnale di completamento (d1)+0 si attiva automaticamente quando viene rilevato il completamento dell'istruzione FWRITE e viene eseguita l'istruzione END. L'operando di completamento si disattiva con l'esecuzione dell'istruzione END della scansione successiva.

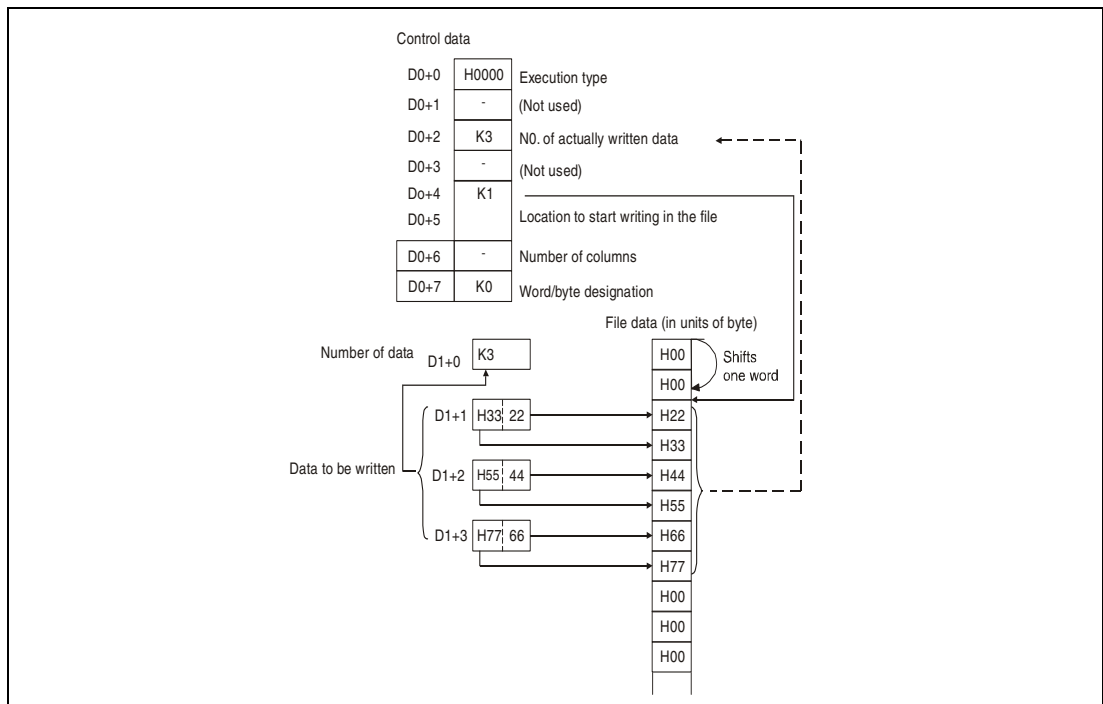
Questo operando può essere usato come flag di completamento dell'istruzione FWRITE.

Se l'istruzione FWRITE termina per errore, l'operando di completamento con errore (d1)+1 si attiva/disattiva in sincronismo con il flag di completamento (d1)+0. Questo operando può essere utilizzato come flag di completamento con errore per l'istruzione FWRITE.

SM721 è attivo (ON) durante l'esecuzione dell'istruzione FWRITE. SM721 viene anche usato da altre istruzioni, come S.FREAD, COMRD e PRC. L'istruzione FWRITE non può essere avviata se SM721 è attivo. In questo caso non viene eseguita nessuna elaborazione.

Se viene rilevato un errore prima dell'esecuzione dell'istruzione (prima dell'attivazione di SM721), l'operando di completamento esecuzione [(d1)+0], l'operando di completamento con errore [(d1)+1] e SM721 non vengono attivati.

L'unità utilizzata per esprimere il numero di dati da scrivere [(s2)+0] è „word“, indipendentemente dall'impostazione in (d0)+7 (specifica word/byte).



Scrittura di dati binari:

Se viene omessa l'estensione nel nome file specificato, viene aggiunta automaticamente l'estensione „.BIN“.

Se il file specificato non esiste, viene creato un nuovo file e i dati vengono aggiunti e salvati a partire dall'inizio del file. Gli attributi di questo nuovo file vengono impostati usando gli attributi di archivio.

Se la dimensione dei dati supera quella dell'area disponibile nel file durante la scrittura, i dati in eccesso vengono aggiunti alla fine del file.

Se la posizione indicata nel file è superiore alla dimensione del file, viene emesso un errore. Una CPU con numero di serie 01111 o precedente (5 cifre più pesanti) emette un codice di errore. Una CPU con numero 01112 o superiore, non scrive i dati e completa l'istruzione senza segnalazione di errore.

Se il supporto di memoria non ha più spazio libero quando vengono aggiunti/salvati i dati, viene emesso un codice di errore. In questo caso i dati che sono stato aggiunti/salvati nel supporto di memoria senza errori, rimangono memorizzati. Il completamento con errore viene segnalato dopo aver aggiunto/salvato la maggior quantità di dati possibile.

Scrittura dati dopo la conversione in formato CSV.

Se viene omessa l'estensione nel nome file specificato, viene aggiunta automaticamente l'estensione „.CSV“.

Se viene specificato un file esistente con una CPU con numero di serie 01111 o precedente (5 cifre più pesanti), tutto il contenuto del file viene cancellato ed i dati specificati vengono salvati dall'inizio del file.

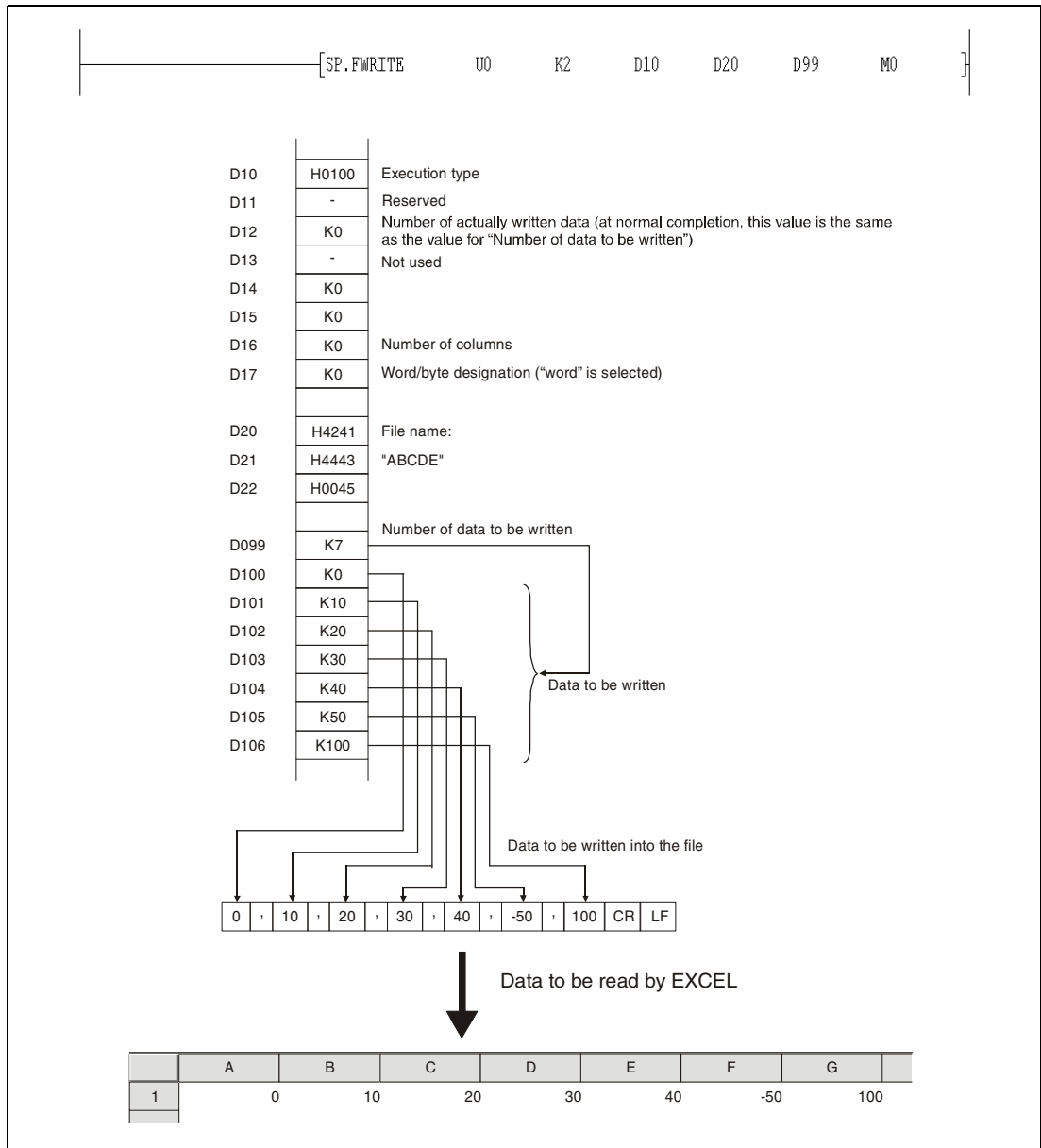
Una CPU successiva (numero di serie 01112 o successivo nelle cinque cifre più pesanti) si comporta in funzione dei valori contenuti in (d0)+4 e (d0)+5, quando viene specificato un file esistente:

Se (d0)+4 e (d0)+5 contengono un valore diverso da FFFFFFFFH , il contenuto del file viene cancellato e i dati vengono memorizzati dall'inizio del file. Se (d0)+4 e (d0)+5 contengono il valore FFFFFFFFH, i dati vengono appesi alla fine del file.

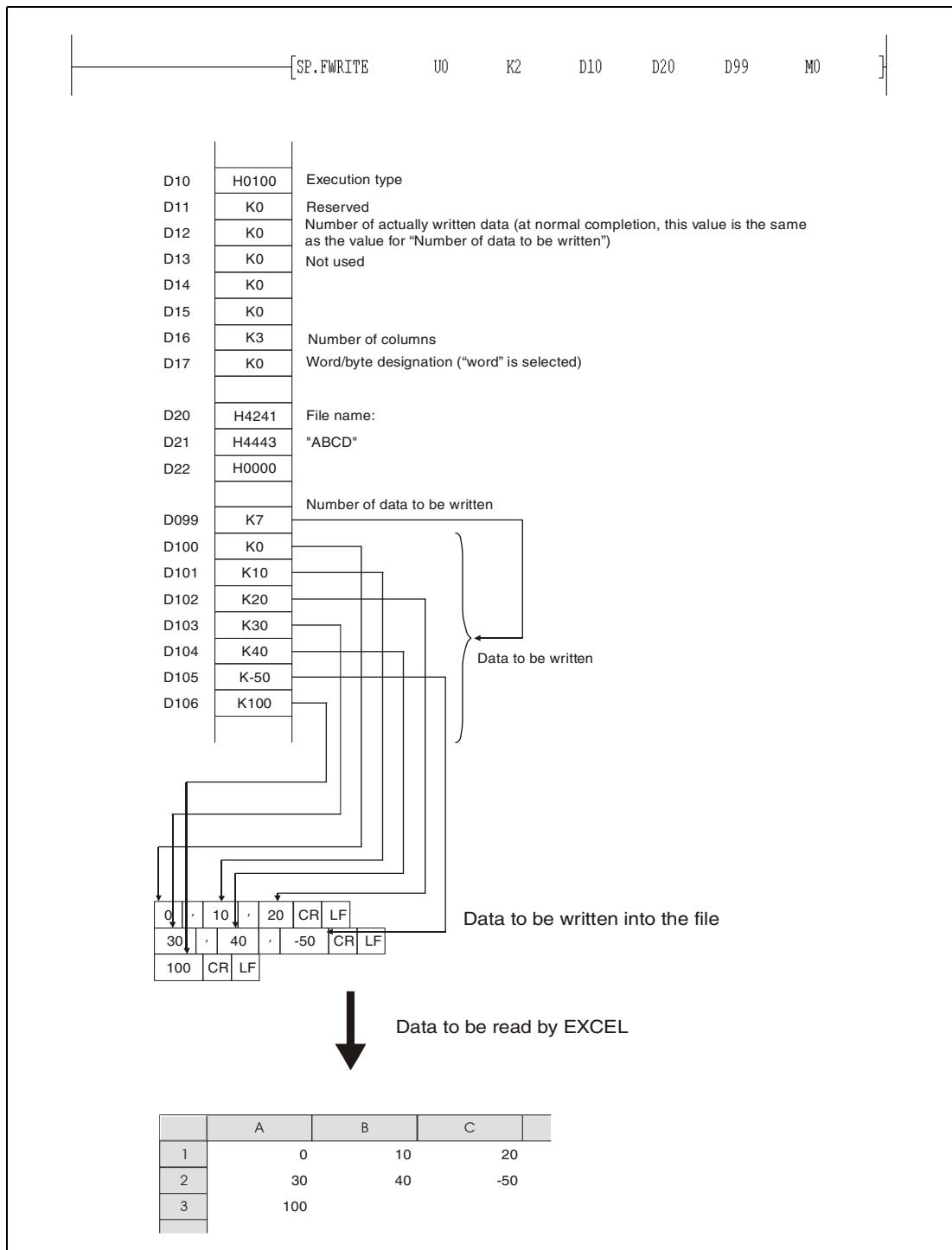
Se il file specificato non esiste, viene creato un nuovo file e i dati vengono aggiunti/salvati a partire dall'inizio del file. Gli attributi di questo nuovo file vengono impostati usando gli attributi di archivio.

Se il supporto di memoria non ha più spazio libero quando vengono aggiunti/salvati i dati, viene emesso un codice di errore. In questo caso i dati che sono stato aggiunti/salvati nel supporto di memoria senza errori, rimangono memorizzati. Il completamento con errore viene segnalato dopo aver aggiunto/salvato la maggior quantità di dati possibile.

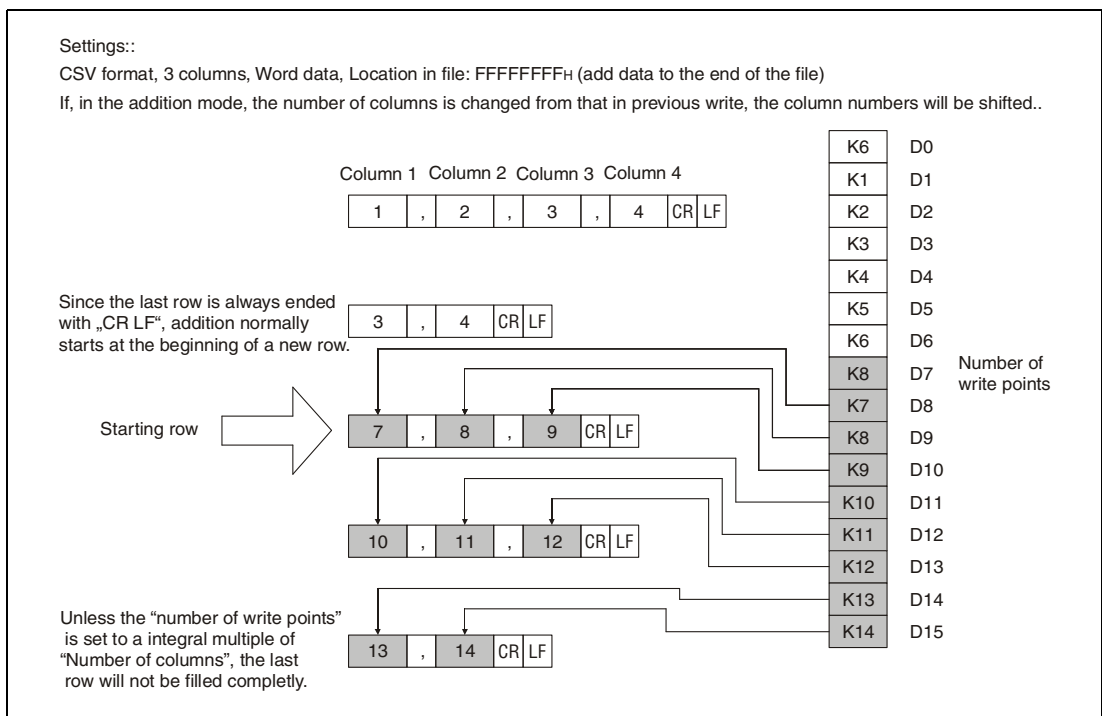
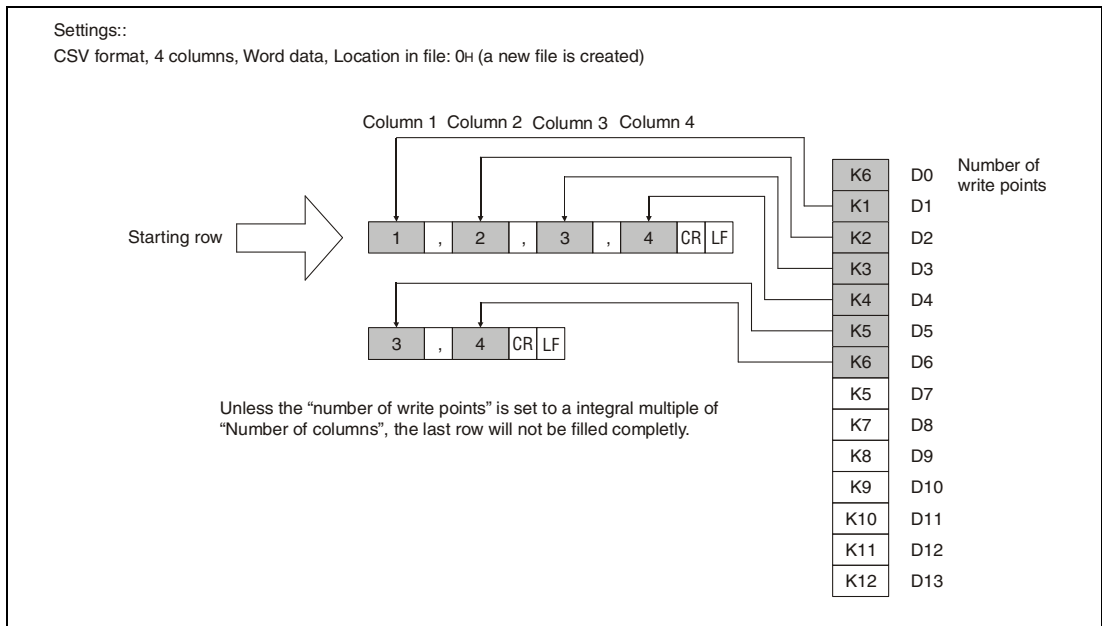
Se il numero di colonne indicate è „0“, i dati vengono memorizzati come dati a riga singola in un file di formato CSV. La figura della pagina seguente mostra questo caso:



Se vengono scritti dati con conversione in formato CSV ed il numero di colonne specificato è diverso da „0“, i dati vengono memorizzati come dati tabellari con il numero di colonne specificato, in un file di formato CSV. La figura seguente mostra un esempio:



Le due figure seguenti mostrano esempi di scrittura dati con una CPU con numero di serie „01112“ (5 cifre pesanti).



NOTA

Non eseguire l'istruzione FWRITE da un programma di interruzione.

Errori di esecuzione

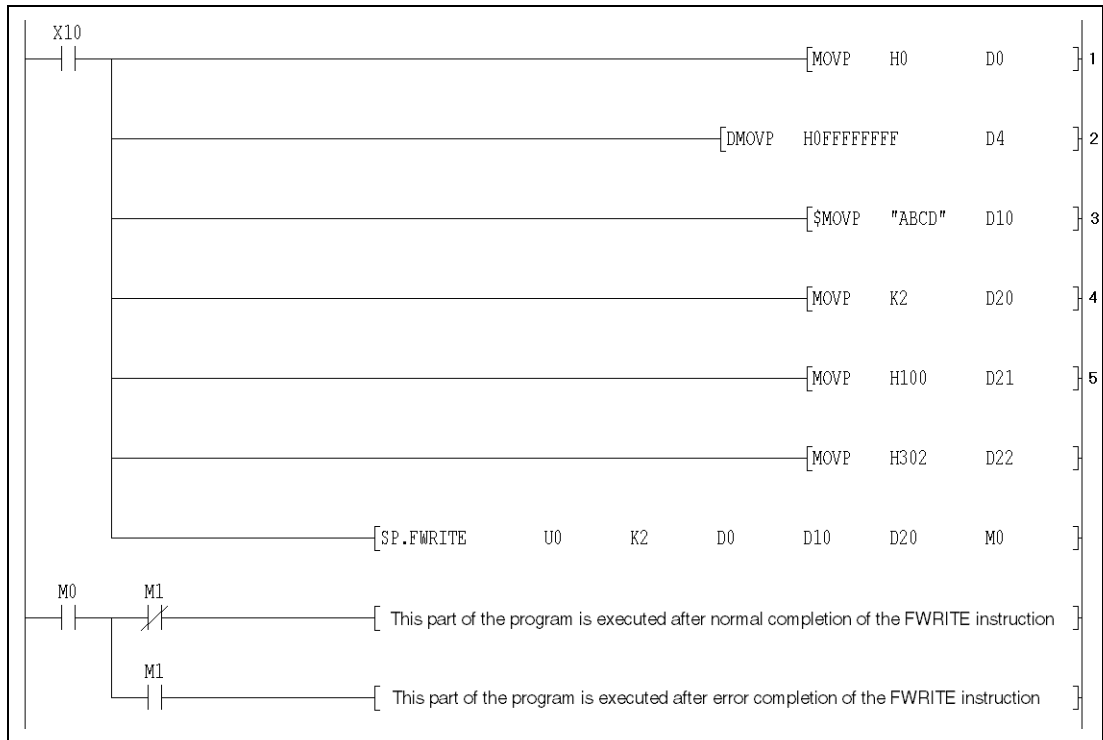
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il drive specificato da s0 contiene un supporto di memoria diverso da una scheda ATA (codice di errore 4100).
- I valori specificati nelle aree dei dati di controllo sono oltre i campi impostati (codice di errore 4100).
- Il valore "numero dati da scrivere" [(s2)+0] è oltre il campo impostato, o superiore alla lunghezza dei dati contenuti nell'area a partire da (s2)+1 (codice di errore 4101).
- Lo spazio libero sul supporto di memoria è insufficiente (codice di errore 4100).
- Nessun nuovo file disponibile durante la creazione di un file (codice di errore 4100).
- Operando specificato non valido (codice di errore 4104).

Programma di esempio 1

FWRITE

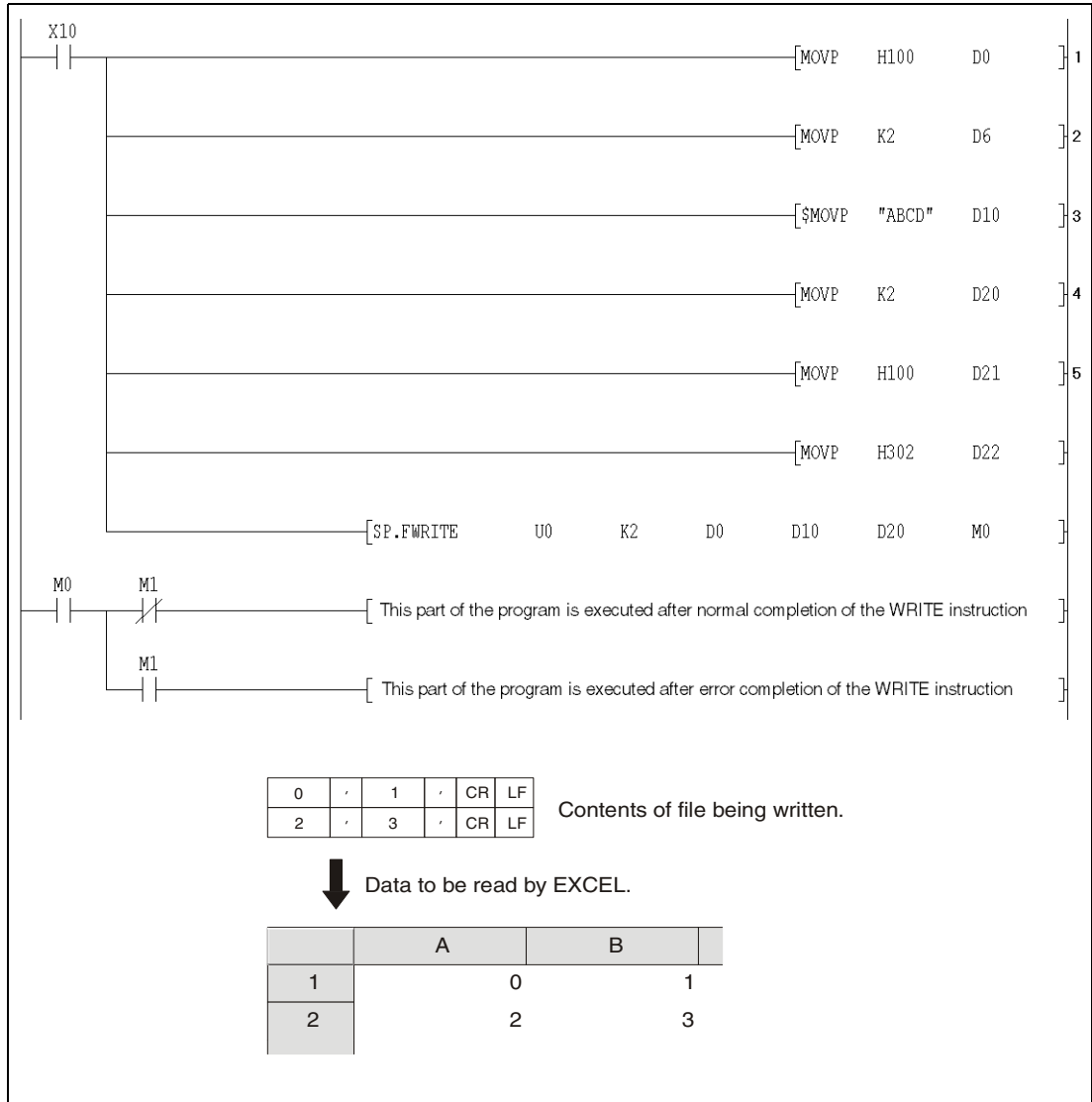
Nel programma di esempio seguente, quattro byte di dati binari (00_H, 01_H, 02_H, e 03_H) vengono aggiunti al file „ABCD.BIN“ quando viene attivato X10. La memory card è inserita nel drive 2. Otto punti, a partire da D0, vengono riservati per i dati di controllo.



- 1 Impostazione del tipo di esecuzione (in questo esempio: dati binari)
- 2 Impostazione della posizione nel file (in questo esempio: aggiunta dati)
- 3 Impostazione del nome file; l'estensione ".BIN" viene aggiunta automaticamente.
- 4 Numero di dati da scrivere.
- 5 I dati (00_H, 01_H, 02_H, e 03_H) vengono inseriti nell'area di controllo.

Programma di esempio 2 FWRITE

Quando X10 viene attivato, il programma seguente crea un file di nome „ABCD.CSV“ nella memory card inserita nel drive 2. Successivamente, quattro byte di dati (00_H, 01_H, 02_H e 03_H) vengono scritti come tabella con due colonne in formato CSV. I dati di controllo vengono scritti a partire da D0 (8 punti).



- ¹ Impostazione del tipo di esecuzione (in questo esempio: formato CSV)
- ² Impostazione del numero di colonne
- ³ Impostazione del nome file; l'estensione ".CSV" viene aggiunta automaticamente.
- ⁴ Numero di dati da scrivere.
- ⁵ I dati (00_H, 01_H, 02_H, e 03_H) vengono inseriti nell'area di controllo.

9.3.2 FREAD

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

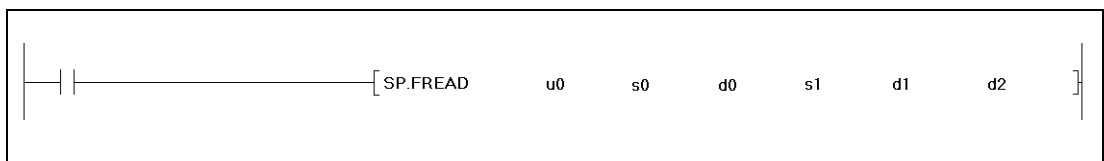
	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s0	●	●	●	—	—	—	—	●	—	SM0	11
d0	—	●	●	—	—	—	—	—	—		
s1	—	●	●	—	—	—	—	—	—		
s2	—	●	●	—	—	—	—	—	●		
d1	●*	●*	●*☆	—	—	—	—	—	—		

* Non possono essere usati operandi locali o operandi riservati per il programma.

GX IEC Developer

<p>Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p>MELSEC</p> </div> <p>SP.FREAD u0 s0 d0 s1 d1 d2</p>	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p>FREAD_M s0, s1, d0, d1, d2</p> </div>
--	--------------------------	---

GX Developer



Variabili

Dati impostati	Significato	Campo impostazione	Impostato da	Tipo dati	
u0	Dummy	—	—		
s0	Indicazione drive. Può essere specificato solo il drive della scheda ATA (drive 2). Non si possono specificare memory card (ROM) o RAM/ROM standard.	2	Utente	BIN 16-bit	
d0	Indirizzo del primo operando che memorizza i dati di controllo. Sono necessari i dati di controllo seguenti.				
	Dati impostati	Caratteristica	Significato/dato impostato	Campo impostazione	Impostato da
	(d0)	Tipo esecuzione-	Specifica il tipo di esecuzione: 0000 _H : Scrittura dati binari 0100 _H : Scrittura dopo conversione in formato CSV	0000 _H 0100 _H	Utente
	(d0)+1	(Riservato)	Usato dal sistema	—	Sistema
	(d0)+2	Numero di dati da leggere.	Specifica il numero di dati da leggere (in unità word). Questo numero deve essere espresso in word anche se è stato selezionato byte in (d0)+7.	da 1 a 480	Utente
	(d0)+3	Non utilizzato	—	—	—
	(d0)+4 (d0)+5	Posizione nel file	Imposta la posizione di inizio lettura all'interno del file quando sono stati selezionati dati binari (d0 = 0000 _H). 00000000 _H : Dall'inizio del file 00000001 _H a FFFFFFFC _H : Dall'indirizzo specificato. L'unità usata per il valore viene definita dalla designazione a word o a byte. FFFFFFFD _H : Impostazione disabilitata Se i dati vengono letti dopo conversione dal formato CSV (d0 = 0100 _H): Nelle CPU con numero di serie "01111" o precedente nelle 5 cifre più pesanti, sempre impostato su inizio file (00000000 _H). Nelle CPU con numero di serie "01112" o successivo, imposta la posizione nel file. 00000000 _H : Dall'inizio del file. 00000001 _H : a FFFFFFFC _H : Dall'indirizzo specificato. FFFFFFFD _H : La lettura continua, iniziando dall'ultima posizione letta	00000000 _H a FFFFFFFC _H FFFFFFFD _H	Utente
	(d0)+6	Numero di colonne	Specifica il numero di colonne dei dati da leggere. 0: : Nessuna colonna impostata. I dati sono considerati inseriti in un singola riga. > 0 : I dati sono considerati come una tabella con il numero di colonne specificato	0, da 1 a 65535	Utente
(d0)+7	Indicazione word/byte	0: Word 1: Byte	0, 1	Utente	

Variabili

Dati impostati	Significato			Campo impostazione	Impostato da	Tipo dati
s1	Indirizzo del primo operando che memorizza il nome file.					
	Dati impostati	Caratteristica	Significato/dato impostato	Campo impostazione	Impostato da	BIN 16-bit
	da (s1) a (s1)+n	Nome file	Il nome file è composto da un massimo di 8 caratteri + punto + estensione (ad esempio: ABD.BIN). L'estensione può essere omessa. In questo caso anche il punto („.“) può essere omesso. Se vengono usati più di 8 caratteri, l'estensione viene ignorata indipendentemente dalla sua presenza. Le estensioni „BIN“ o „CSV“ vengono assegnate automaticamente.	Stringa di caratteri	Utente	
Indirizzo del primo operando che memorizza i dati.						
d1	Dati impostati	Caratteristica	Significato/dato impostato	Campo impostazione	Impostato da	BIN 16-bit
	(d1)	Esito lettura (numero dati letti)	Contiene il numero dei dati effettivamente letti. L'unità usata per il valore viene definita dalla designazione a word o a byte.	da 0 a 480	Sistema	
	da (d1)+1 a (d1)+n	Dati da leggere	Lunghezza dati richiesta	0000 _H a FFFF _H		
d2	Operando a bit attivato al termine dell'esecuzione dell'istruzione FREAD. In caso di errore viene attivato (d1)+1.					
	Dati impostati	Caratteristica	Significato/dato impostato	Campo impostazione	Impostato da	Bit
	(d2)	Segnale di completamento	Indica lo stato di completamento dell'istruzione FREAD ON: Completato OFF: Non completato	—	Sistema	
(d2)+1	Segnale completamento con errore	Indica se l'istruzione FWRITE si è conclusa normalmente o con errore. ON: Completamento con errore OFF: Completamento normale	—			

NOTA

I dati scritti in formato CSV sono espressi come valore decimale dal software di programmazione. Ad esempio, il carattere „A“ (41_H) è scritto come 65. Il campo valori ammesso è da -32768 a 32767.

Funzioni Lettura dati dal file designato

FREAD Lettura dati

L'istruzione FREAD legge il numero di dati specificato, da un file contenuto in una scheda ATA. L'utente può scegliere se leggere i dati direttamente in binario, senza alcuna conversione, oppure se convertirli in formato CSV prima della lettura.

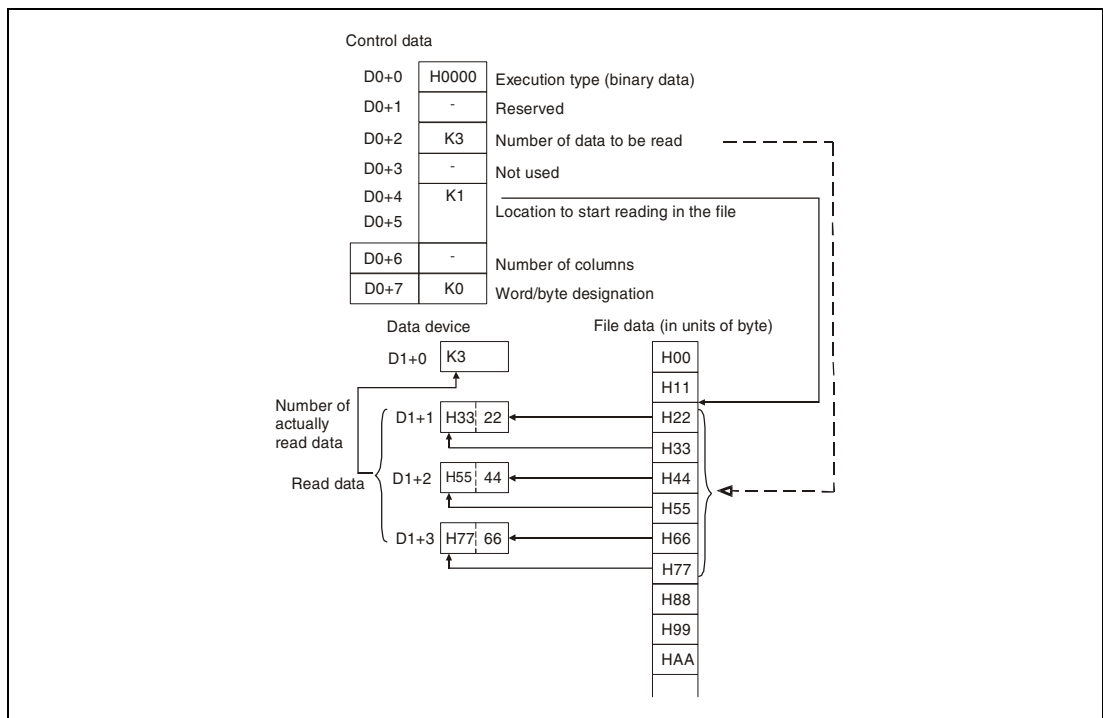
L'operando a bit del segnale di completamento (d1)+0 si attiva automaticamente quando viene rilevato il completamento dell'istruzione FREAD e viene eseguita l'istruzione END. L'operando di completamento si disattiva con l'esecuzione dell'istruzione END della scansione successiva. Questo operando può essere usato come flag di completamento dell'istruzione FREAD.

Se l'istruzione FREAD termina per errore, l'operando di completamento con errore (d1)+1 si attiva/disattiva in sincronismo con il flag di completamento (d2)+0. Questo operando può essere utilizzato come flag di completamento con errore per l'istruzione FREAD.

SM721 è attivo (ON) durante l'esecuzione dell'istruzione FREAD. SM721 viene anche usato da altre istruzioni, come S.FREAD, COMRD e PRC. L'istruzione FREAD non può essere avviata se SM721 è attivo. In questo caso non viene eseguita nessuna elaborazione.

Se viene rilevato un errore prima dell'esecuzione dell'istruzione (prima dell'attivazione di SM721), l'operando di completamento esecuzione [(d2)+0], l'operando di completamento con errore [(d2)+1] e SM721 non vengono attivati.

L'unità utilizzata per esprimere il numero di dati da scrivere [(d0)+0] è „word“, indipendentemente dall'impostazione in (d0)+7 (specifica word/byte). La figura seguente mostra la lettura di dati binari:



Letture di dati binari:

Se viene omessa l'estensione nel nome file specificato, viene aggiunta automaticamente l'estensione „.BIN“.

Se il file specificato non esiste, viene emesso un errore.

Se la posizione indicata nel file è superiore alla dimensione del file, viene emesso un errore. Una CPU con numero di serie 01111 o precedente (5 cifre più pesanti) emette un codice di errore. Una CPU con numero 01112 o superiore, non legge i dati e completa l'istruzione senza segnalazione di errore.

Letture dati dopo la conversione in formato CSV.

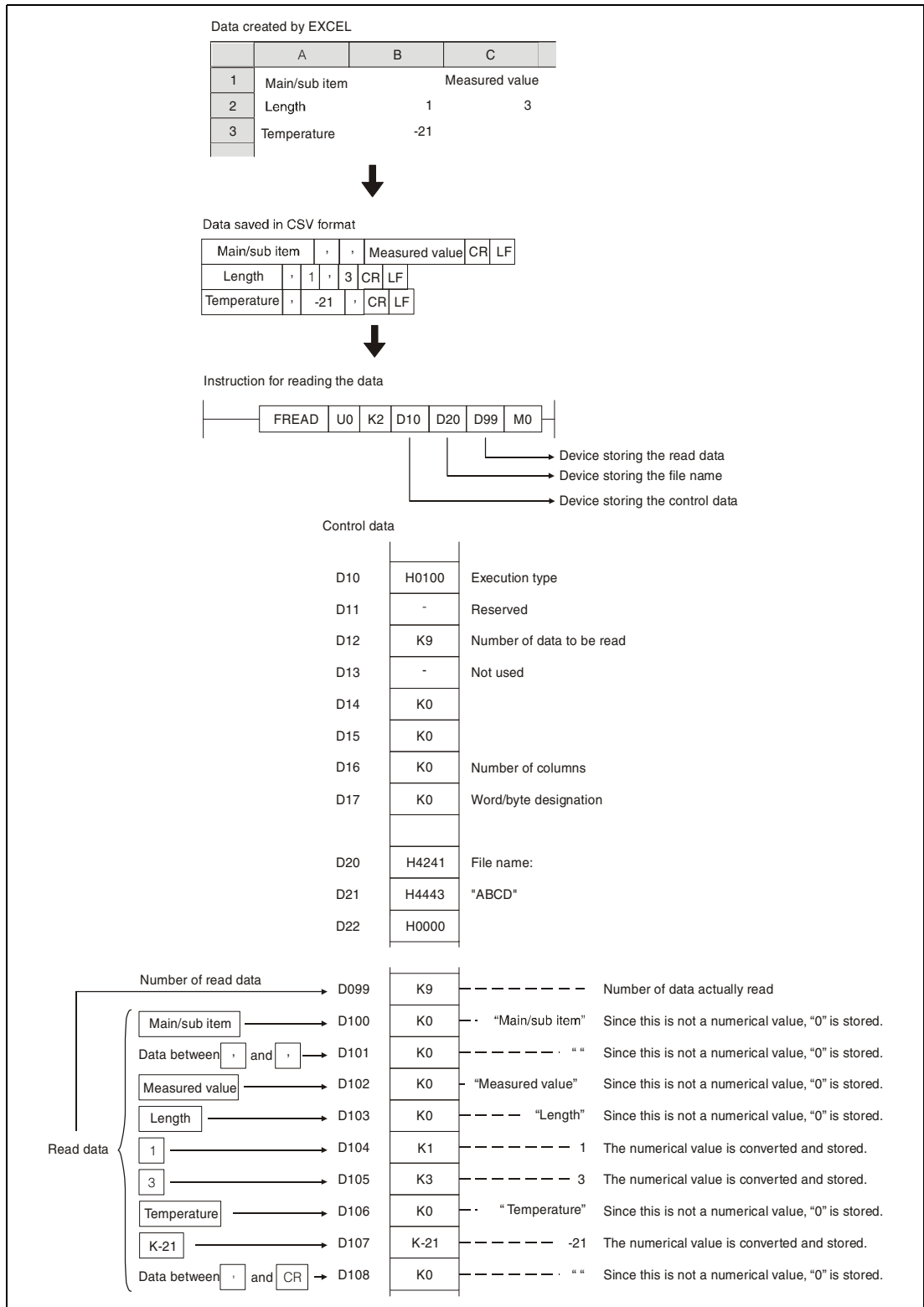
Gli elementi di un file in formato CSV (celle di EXCEL) vengono letti riga per riga. I valori numerici e le stringhe di caratteri vengono convertiti in dati binari e memorizzati nel supporto di memoria.

Se viene omessa l'estensione nel nome file specificato, viene aggiunta automaticamente l'estensione „.CSV“.

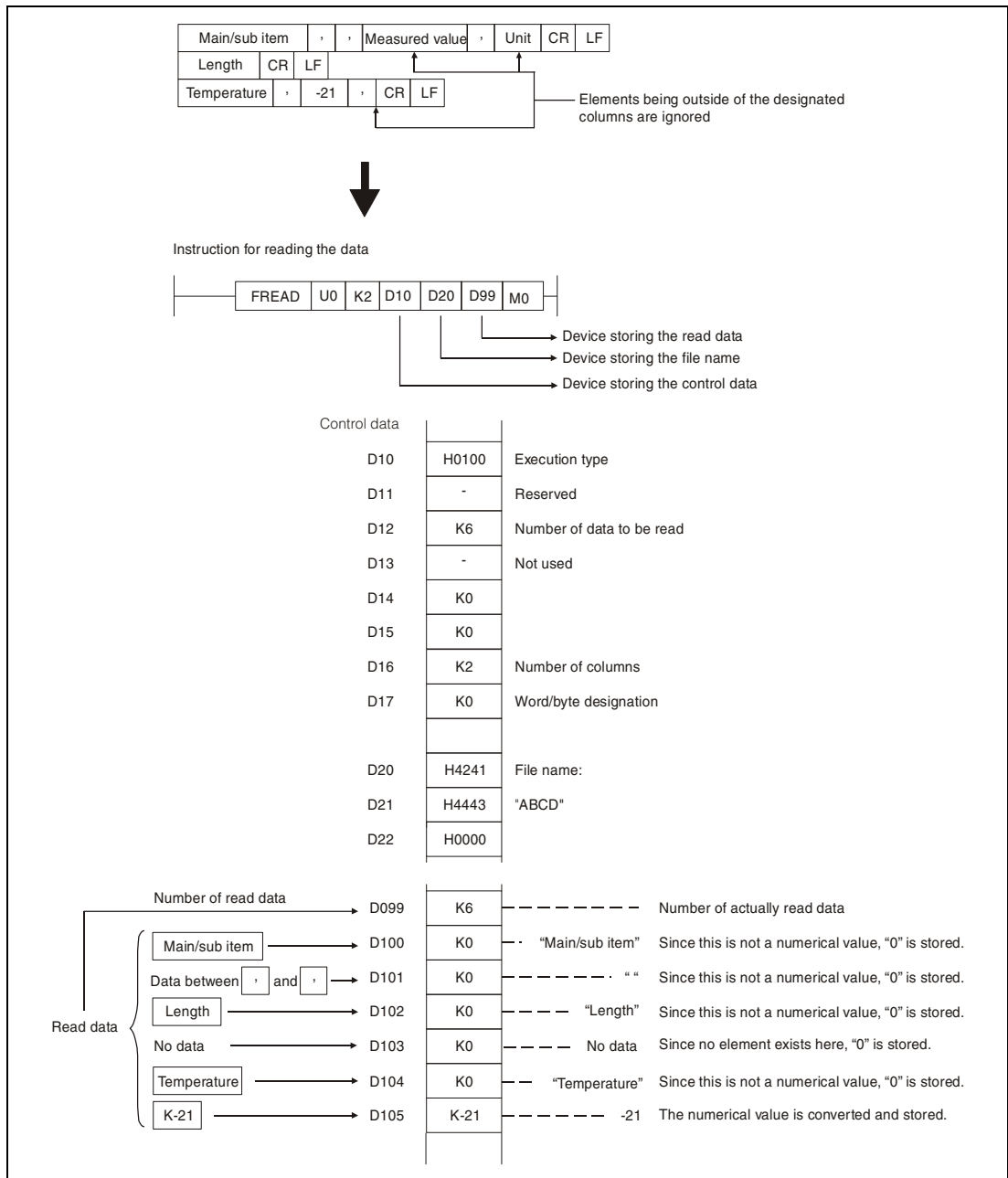
Se il file specificato non esiste, viene emesso un errore.

La lettura inizia dalla posizione specificata nel file. Il numero di elementi da leggere viene impostato con il dato di controllo (d0)+2. Se l'ultimo dato contenuto nel file viene letto prima di aver raggiunto il numero di caratteri specificato, una CPU con numero di serie 01111 o precedente (cinque cifre superiori) emette un codice di errore. Una CPU con numero di serie 01112 o successiva, legge semplicemente i dati disponibili.

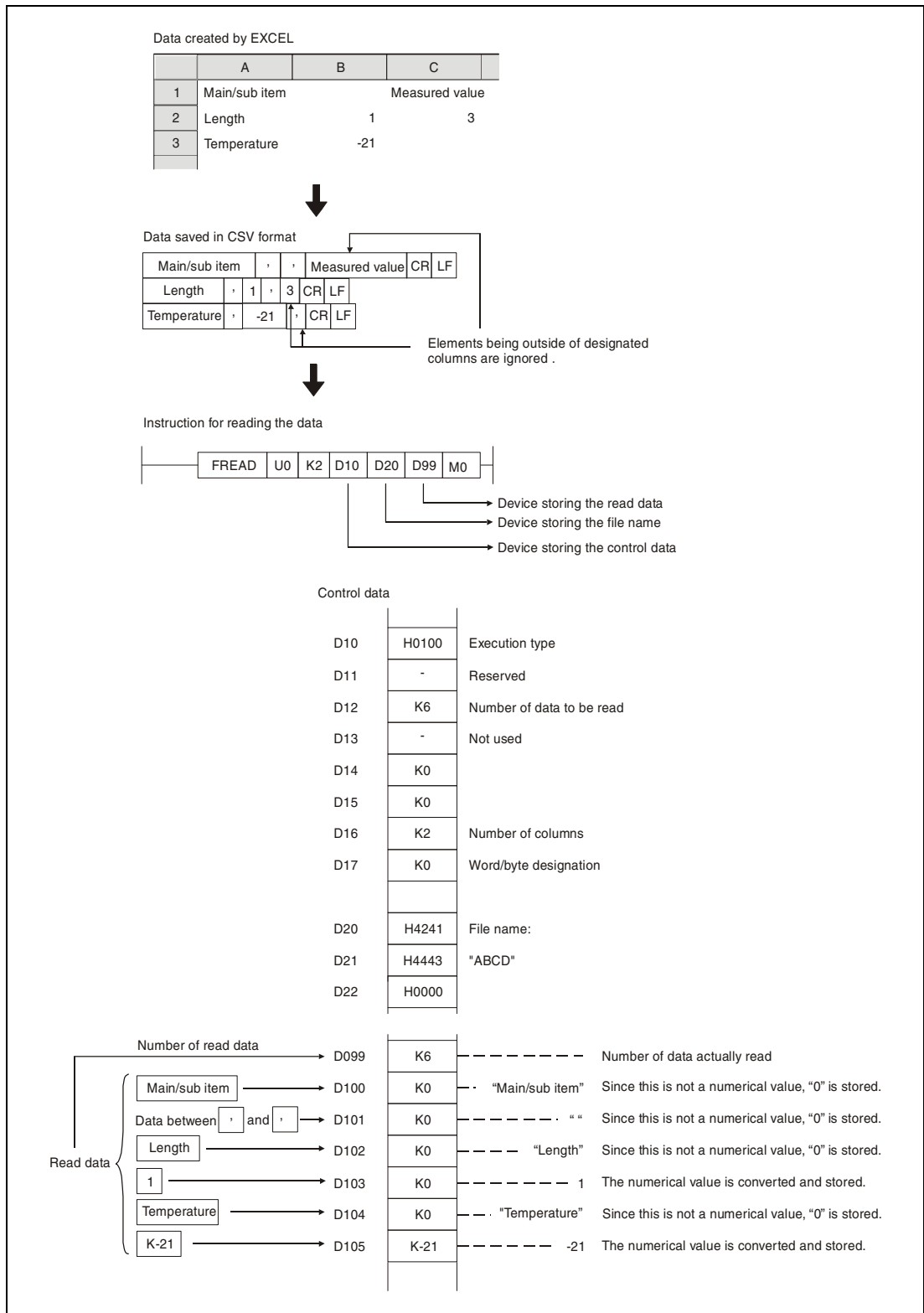
Se il numero di colonne indicate è „0“, i dati vengono letti ignorando le righe di un file di formato CSV. La figura a pagina seguente mostra la gestione dati in questo caso.



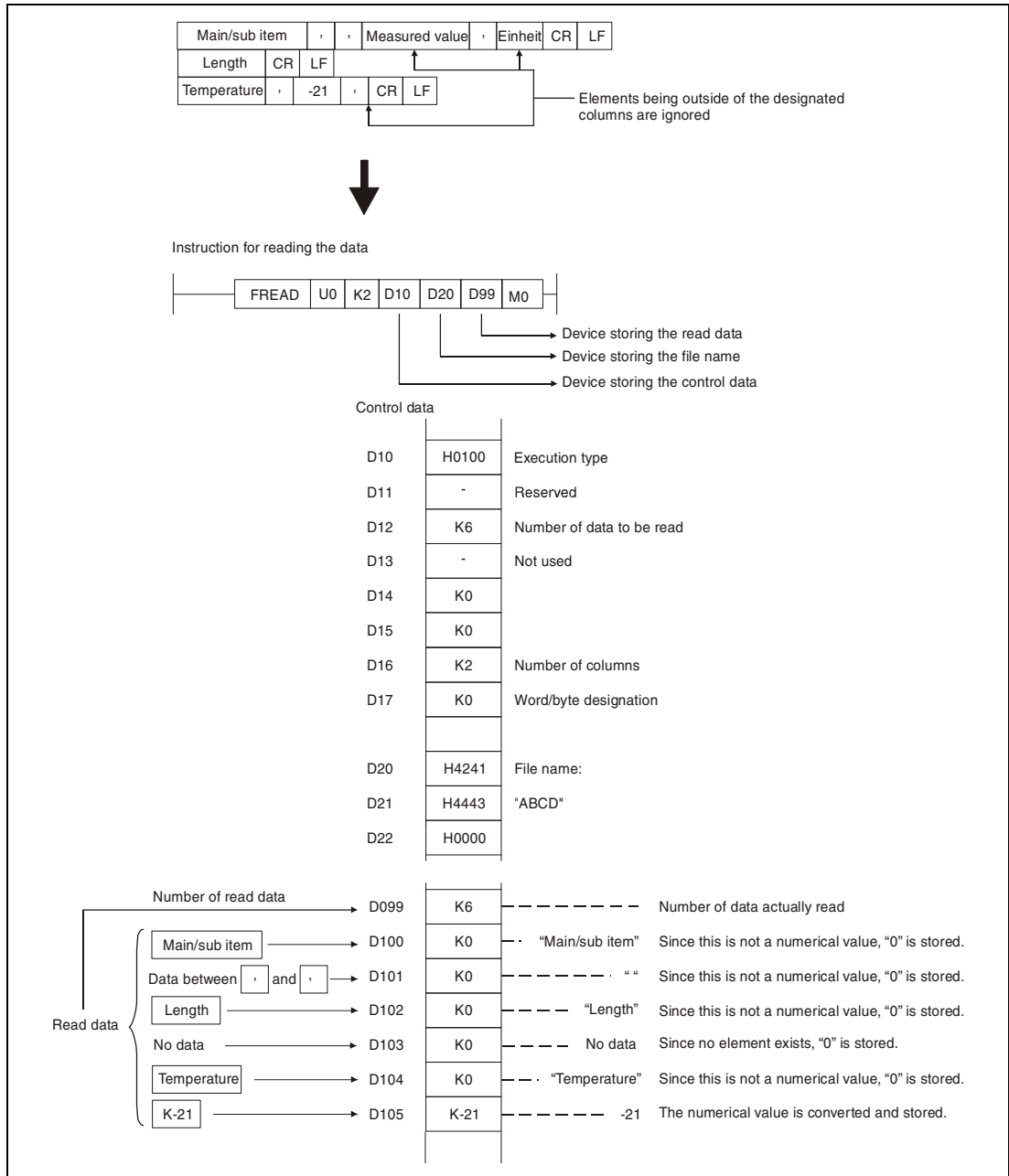
Anche se il numero di colonne varia in ogni riga, i dati vengono letti ignorando le righe. (EXCEL non crea questo tipo di file. Questo accade modificando un file CSV)



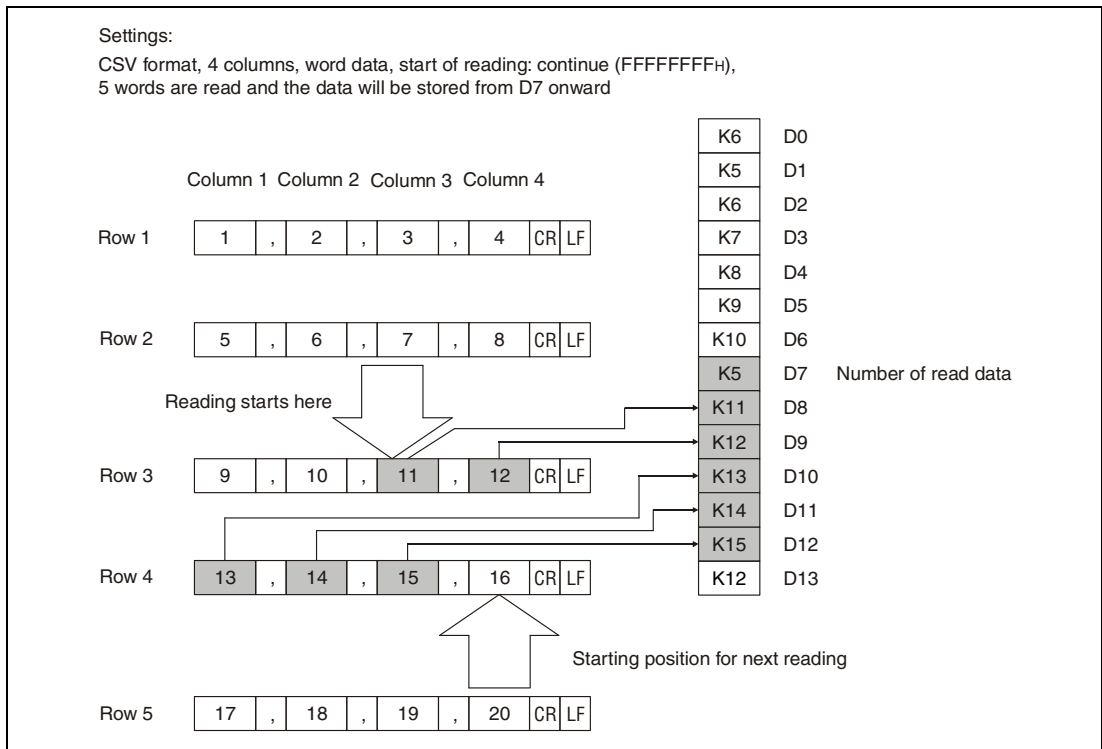
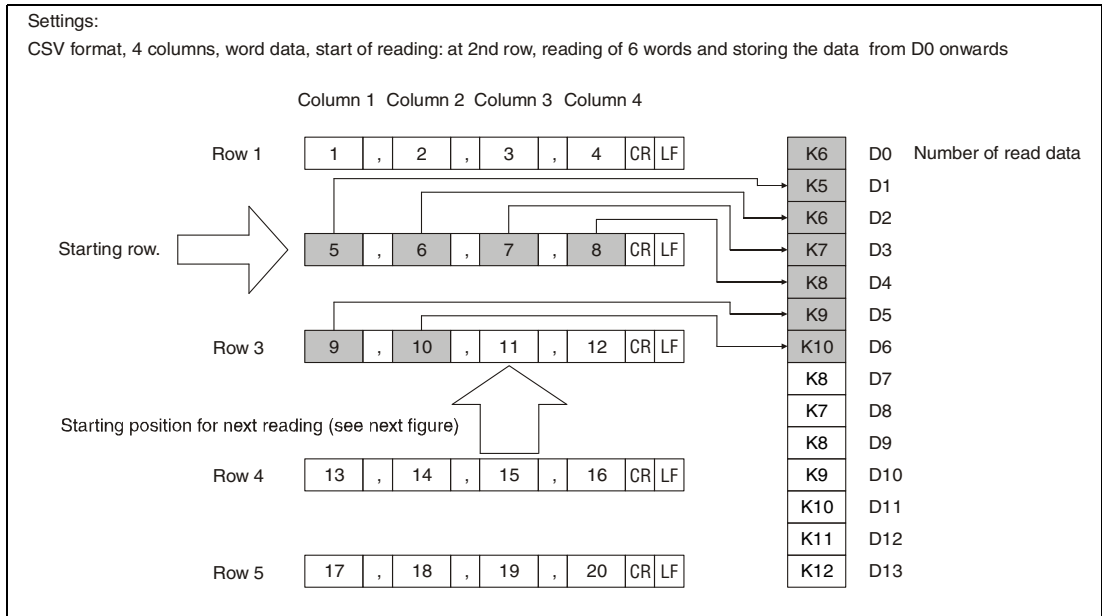
Se vengono letti dati con conversione da formato CSV ed il numero di colonne specificato è diverso da „0“, i dati vengono letti come dati tabellari con il numero di colonne specificato. Gli elementi oltre il numero di colonne specificato vengono ignorati. La figura seguente mostra questo caso:



Se il numero di colonne varia in ogni riga, gli elementi oltre il numero di colonne specificato vengono ignorati, mentre nelle posizioni che non corrispondono a elementi letti viene inserito „0“.
 Se il numero di righe nel file è inferiore a quello specificato da (d0)+2 (numero dati da leggere), nelle posizioni corrispondenti alle righe inesistenti viene inserito „0“.



Le figure seguenti mostrano il caso in cui i dati vengono letti separatamente per diverse volte dallo stesso file (modo continuazione)m usando una CPU con numero di serie "01112" o successivo nelle cinque cifre più pesanti.



Se si esegue una lettura in modo continuo, le impostazioni del formato dati, numero di colonne e lettura a word/byte non devono cambiare rispetto alla lettura precedente.

Durante la lettura in modo continuo, si deve disabilitare l'esecuzione di altre istruzioni FREAD o FWRITE.

Quando si leggono dati con conversione dal formato CSV, i valori numerici vengono letti e convertiti come segue:

Valori numerici in formato CSV	Operando a word	
	senza segno	con segno
-32768	32768	-32768
-1	65535	-1
0	0	0
1	1	1
32767	32767	32767
32768	32768	-32768
65535	65535	-1

Valori numerici fuori campo ed elementi diversi da valori numerici nel file CSV specificato, vengono convertiti in „0“.

NOTA

Non eseguire l'istruzione FREAD da un programma di interruzione.

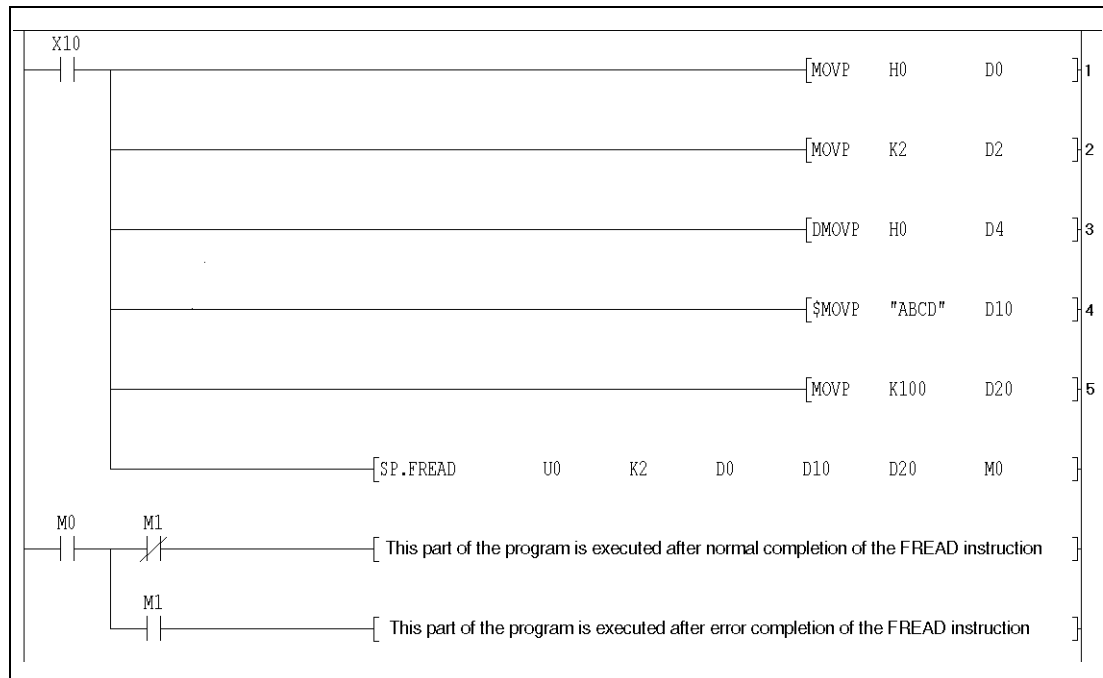
Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il drive specificato da s0 contiene un supporto di memoria diverso da una scheda ATA (codice di errore 4100).
- I valori specificati nelle aree dei dati di controllo sono oltre i campi impostati (codice di errore 4100).
- Il valore "numero dati da leggere" [(d0)+0] è oltre il campo impostato (codice di errore 4101).
- Operando specificato non valido (codice di errore 4004).
- Il nome file specificato da s1 non esiste nel drive specificato (codice di errore 2410).
- La lunghezza dei dati da leggere supera la dimensione del dispositivo di lettura (codice di errore 4101).
- Nella lettura di dati binari, il numero di dati contenuti nel file è inferiore alla lunghezza specificata dei dati da leggere [(d0)+2] (codice di errore 4100).

**Programma
di esempio 1** FREAD

Se X10 è attivo, quattro byte di dati binari vengono letti a partire dall'inizio del file „ABCD.BIN“. Il file „ABCD.BIN“ è contenuto in una memory card inserita nel drive 2. Otto punti sono riservati per i dati di controllo, a partire da D0. Per contenere i dati letti, vengono riservati 100 byte a partire da D20.



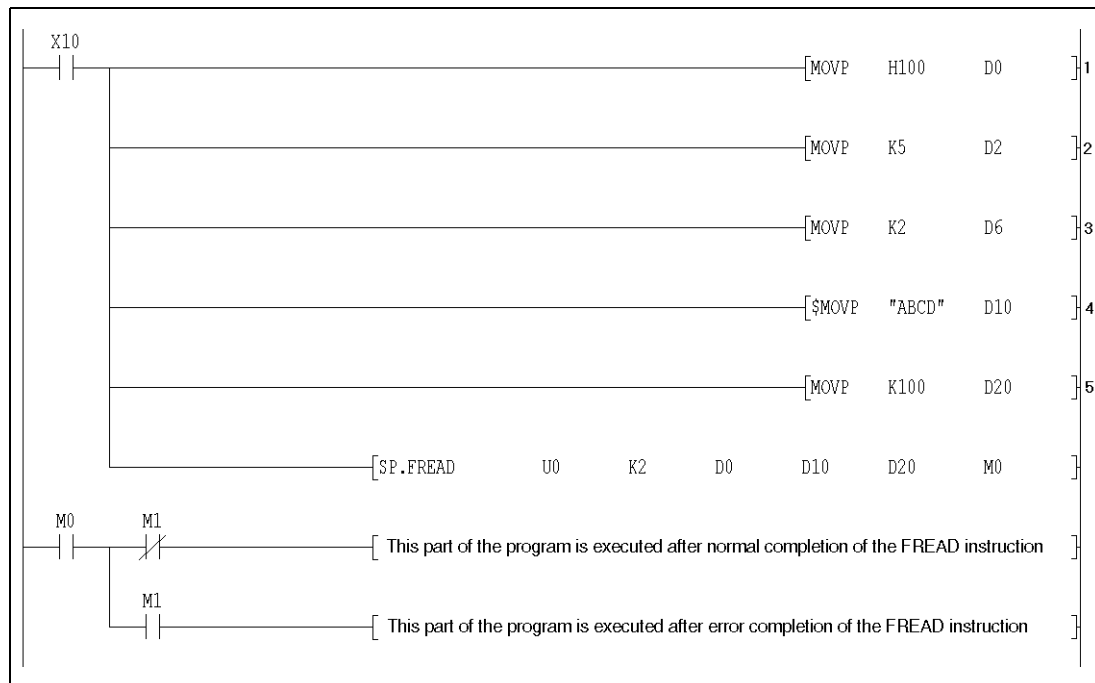
- 1 Impostazione tipo di esecuzione
- 2 Impostazione numero dati da leggere
- 3 Indirizzo iniziale nel file (lettura a partire da inizio file)
- 4 Trasferimento del nome file nei dati di controllo
- 5 Impostazione della dimensione del dispositivo di lettura

Programma di esempio 2 FREAD

Il programma seguente legge i dati dal file „ABCD.CSV“, che è contenuto nella scheda di memoria nel drive 2, se X10 è attivo. Il contenuto del file è una tabella a due colonne in formato CSV. Il file contiene solo valori numerici.

Otto punti sono riservati per i dati di controllo, a partire da D0.

Per contenere i dati letti, vengono riservati 100 byte a partire da D20.



- 1 Impostazione del tipo di esecuzione (in questo esempio: formato CSV)
- 2 Impostazione numero dati da leggere
- 3 Impostazione del numero di colonne
- 4 Trasferimento del nome file nei dati di controllo
- 5 Impostazione della dimensione del dispositivo di lettura

9.4 Istruzioni di programma

9.4.1 PLOADP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					● ¹

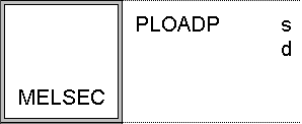
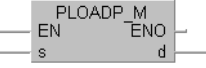
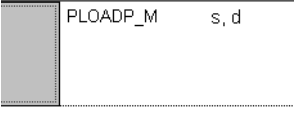
¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

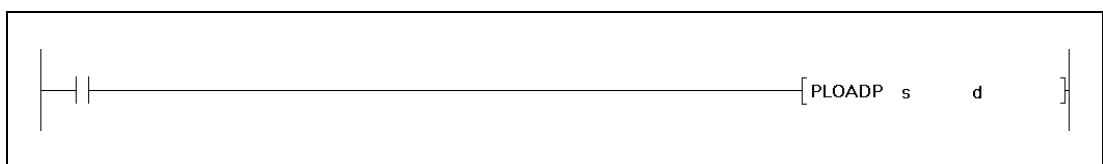
	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	—	●	SM0	3
d	●*	—	—	—	—	—	—	—	—		

* non si possono usare operando locali

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
		

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Numero del drive che contiene il programma da caricare, stringa di caratteri del nome file, o indirizzo iniziale dell'operando che contiene la stringa di caratteri	BIN 16-bit
d	Operando attivato per una scansione al completamento dell'istruzione	Bit

NOTA

Il file system non è supportato da GX IEC Developer.

Funzioni **Caricamento di un programma da una memory card****PLOADP** **Caricamento programma**

L'istruzione PLOADP carica nella memoria interna (drive 0) un programma contenuto in una memory card o una memoria standard, e pone il programma in modo standby. La memory card può essere inserita nei drive 1, 2 o 4. Il drive 0 deve possedere sufficiente spazio libero continuo.

Non è necessario specificare l'estensione „QPG“, nel nome file.

L'operando a bit indicato da d viene attivato durante l'elaborazione dell'istruzione END della scansione in cui viene completata l'istruzione PLOADP. L'operando viene disattivato con l'istruzione END della scansione successiva.

Le istruzioni PLOADP, PUNLOADP e PSWAPP non possono essere usate contemporaneamente. Se vengono richiamate due o più istruzioni, l'ultima istruzione richiamata non viene eseguita.

Per evitare questi casi, occorre stabilire degli interblocchi.

Come numero del programma che viene aggiunto, viene utilizzato il più basso numero di programma inutilizzato nella CPU. I numeri di programma possono essere controllati con GX Developer, leggendo la lista programmi. È possibile specificare un numero per il programma da aggiungere, inserendone il numero in SD 720.

L'istruzione PLOADP non può essere eseguita da un programma in interruzione.

Per eseguire il programma trasferito nella memoria programmi con l'istruzione PLOADP, deve essere eseguita l'istruzione PSCAN.

Le impostazioni del file PLC relativo al programma caricato sono le seguenti:

Per ciascun programma:

le impostazioni relative a file registri, valore iniziale operandi, commenti e operandi locali del programma caricato sono ricavate da "Use PLC file setting" (usa impostazioni da file PLC).

Tuttavia, se nel file PLC è specificato "Use local device" e viene caricato un programma, si verifica un errore ogni volta che il numero dei programmi eseguiti supera il numero dei programmi impostato con i parametri. Per usare operandi locali in un programma caricato, registrare un file dummy nei parametri, cancellare il file dummy con l'istruzione PUNLOADP, poi caricare il programma effettivo con l'istruzione PLOADP.

Impostazione rinfresco I/O:

L'impostazione del rinfresco I/O per il programma caricato è "Disabled" (disabilitato) per ingressi e uscite.

Non si possono apportare modifiche ad un programma in RUN durante l'esecuzione dell'istruzione PLOADP. Le modifiche vengono trasferite al termine dell'istruzione PLOADP. Simmetricamente, l'istruzione PUNLOADP non viene eseguita durante una modifica con programma in RUN, ma la sua esecuzione viene avviata al termine del trasferimento della modifica.

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il file indicato non esiste sul drive specificato da s (codice di errore 2410).
- Il numero drive specificato da s non esiste (codice di errore 4100).
- Spazio insufficiente per il caricamento del programma specificato nel drive 0 (codice di errore 2413).
- Il numero di programma mostrato di seguito è già registrato nella memoria programmi (codice di errore 4101).
- Il numero di programma contenuto in SD720 è già usato, o superiore al massimo numero di programma mostrato di seguito (codice di errore 4101).

Tipo di CPU	Memoria programmi (numero di file)	Massimo numero programma
Q02(H)	28	28
Q06H	60	60
Q12H	124	124
Q25H	124	124

- Un file programma con lo stesso nome del file che si vuole caricare è già esistente (codice di errore 2410).
- Impossibile allocare la dimensione file per gli operandi locali (codice di errore 2401).

Esempio di programma

PLOADP

Se M0 è attivo nel programma seguente, il programma „ABCD.QPG“ viene trasferito dal drive 4 al drive 0 e posto in modo standby.

Lista istruzioni	Schema a contatti
<pre> 0 LD M0 1 PLOADP "4:ABCD" M10 7 END </pre>	

9.4.2 PUNLOADP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					● ¹

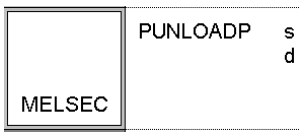


¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

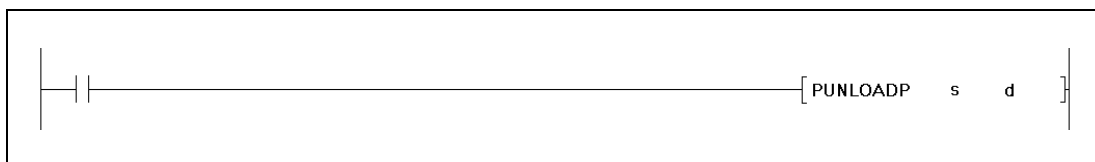
	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	—	●	SM0	3
d	●*	—	—	—	—	—	—	—	—		

* non si possono usare operando locali

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
---	--	--

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Stringa di caratteri che indica il nome del file da scaricare, o indirizzo del primo operando che memorizza questi dati	BIN 16-bit
d	Operando attivato per una scansione al completamento dell'istruzione	Bit

NOTA

Il file system non è supportato da GX IEC Developer.

Funzioni **Scaricamento di un programma dalla memoria programmi****PUNLOADP Scaricamento (cancellazione) programma**

L'istruzione PUNLOADP viene utilizzata per cancellare un programma in standby dalla memoria programmi (drive 0). Il programma in esecuzione attivato dall'istruzione PSCAN non può essere cancellato.

Non è necessario specificare l'estensione „QPG“, nel nome file.

L'operando a bit indicato da d viene attivato durante l'elaborazione dell'istruzione END della scansione in cui l'istruzione PUNLOADP viene completata. L'operando viene disattivato con l'istruzione END della scansione successiva.

Le istruzioni PLOADP, PUNLOADP e PSWAPP non possono essere usate contemporaneamente. Se vengono richiamate due o più istruzioni, l'ultima istruzione richiamata non viene eseguita.

Per evitare questi casi, occorre stabilire degli interblocchi.

Se l'alimentatore della CPU viene spento e poi riacceso, oppure se il modulo CPU viene resettato dopo la cancellazione di un programma, viene emesso „FILE SET ERROR (codice di errore 2400)“. Per risolvere questo problema, cancellare il nome del programma cancellato dai dati di configurazione di programma.

L'istruzione PUNLOADP non può essere eseguita da un programma in interruzione.

Il programma che si vuole cancellare dalla memoria programmi con l'istruzione PUNLOADP, deve prima essere posto in modo standby con l'istruzione PSTOP.

Non si possono apportare modifiche ad un programma in RUN durante l'esecuzione dell'istruzione PUNLOADP. Le modifiche vengono trasferite al termine dell'istruzione PUNLOADP. Simmetricamente, l'istruzione PUNLOADP non viene eseguita durante una modifica con programma in RUN, ma la sua esecuzione viene avviata al termine del trasferimento della modifica.

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il nome file specificato da s non esiste (codice di errore 2410).
- Il programma specificato da s non è in standby o è in esecuzione (codice di errore 4101).
- Il programma specificato da s è il solo presente nella memoria programmi (codice di errore 4101).

Programma Esempio PUNLOADP

Il programma seguente cancella dalla memoria il programma „ABCD.QPG“ contenuto nel drive 0, quando M0 passa da OFF a ON.

Lista istruzioni	Schema a contatti
<pre> 0 LD M0 1 PUNLOADP "ABCD" M10 6 END </pre>	

9.4.3 PSWAPP

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					● ¹

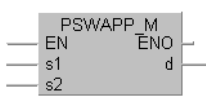
¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

Operandi MELSEC Q

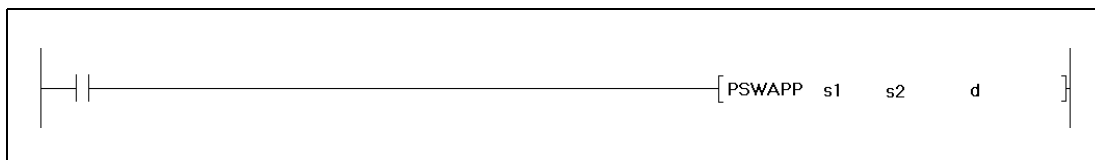
	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	●	SM0	3
s2	—	●	●	—	—	—	—	—	●		
d	●*	—	—	—	—	—	—	—	—		

* non si possono usare operando locali

GX IEC Developer

<p>Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; border: 1px solid black; text-align: center;">MELSEC</td> <td style="border: 1px solid black; padding: 2px;">PSWAPP</td> <td style="border: 1px solid black; padding: 2px;">s1 s2 d</td> </tr> </table> </div>	MELSEC	PSWAPP	s1 s2 d	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; border: 1px solid black; text-align: center;">PSWAPP_M</td> <td style="border: 1px solid black; padding: 2px;">s1, s2, d</td> </tr> </table> </div>	PSWAPP_M	s1, s2, d
MELSEC	PSWAPP	s1 s2 d					
PSWAPP_M	s1, s2, d						

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s1	Stringa di caratteri che indica il nome del file da cancellare, o indirizzo del primo operando che memorizza questi dati	BIN 16-bit
s2	Numero del drive che contiene il programma da caricare, stringa di caratteri del nome file, o indirizzo iniziale dell'operando che contiene la stringa di caratteri	BIN 16-bit
d	Operando attivato per una scansione al completamento dell'istruzione	Bit

NOTA

Il file system non è supportato da GX IEC Developer.

Funzioni **Cancellazione di un programma dalla memoria e caricamento di un programma****PSWAPP Scaricamento e caricamento di un programma**

L'istruzione PSWAPP cancella un programma in standby dalla memoria programmi (drive 0). Il programma da cancellare viene specificato tramite s1. Il programma in esecuzione attivato dall'istruzione PSCAN non può essere cancellato. Dopo la cancellazione, uno dei programmi contenuto nel drive 1, 2 o 4 viene trasferito nella memoria programmi e posto in modo standby. Questo programma è specificato da s2. Il drive 0 della memoria programmi deve avere sufficiente spazio libero continuo prima di caricare il programma.

Non è necessario specificare l'estensione „QPG“, nel nome file.

L'operando a bit indicato da d viene attivato durante l'elaborazione dell'istruzione END della scansione in cui l'istruzione PSWAPP viene completata. L'operando viene disattivato con l'istruzione END della scansione successiva.

Le istruzioni PLOADP, PUNLOADP e PSWAPP non possono essere usate contemporaneamente. Se vengono richiamate due o più istruzioni, l'ultima istruzione richiamata non viene eseguita.

Per evitare questi casi, occorre stabilire degli interblocchi.

Il numero di programma del programma cancellato viene utilizzato per il nuovo programma caricato.

Se l'alimentatore della CPU viene spento e poi riacceso, oppure se il modulo CPU viene resettato dopo la sostituzione di un programma, viene emesso „FILE SET ERROR (codice di errore 2400)“. Per risolvere il problema, inserire nel nome del programma cancellato nelle impostazioni di programma, quello del programma attivato.

L'istruzione PSWAPP non può essere eseguita da un programma in interruzione.

Le impostazioni del file PLC relativo al programma caricato sono le seguenti:

- le impostazioni relative a file registri, valore iniziale operandi, commenti e operandi locali del programma caricato sono ricavate da "Use PLC file setting" (usa impostazioni da file PLC).
- L'impostazione del rinfresco I/O per il programma caricato è "Disabled" (disabilitato) per ingressi e uscite.

Non si possono apportare modifiche ad un programma in RUN durante l'esecuzione dell'istruzione PSWAPP. Le modifiche vengono trasferite al termine dell'istruzione PSWAPP. Simmetricamente, l'istruzione PSWAPP non viene eseguita durante una modifica con programma in RUN, ma la sua esecuzione viene avviata al termine del trasferimento della modifica.

Errori di esecuzione

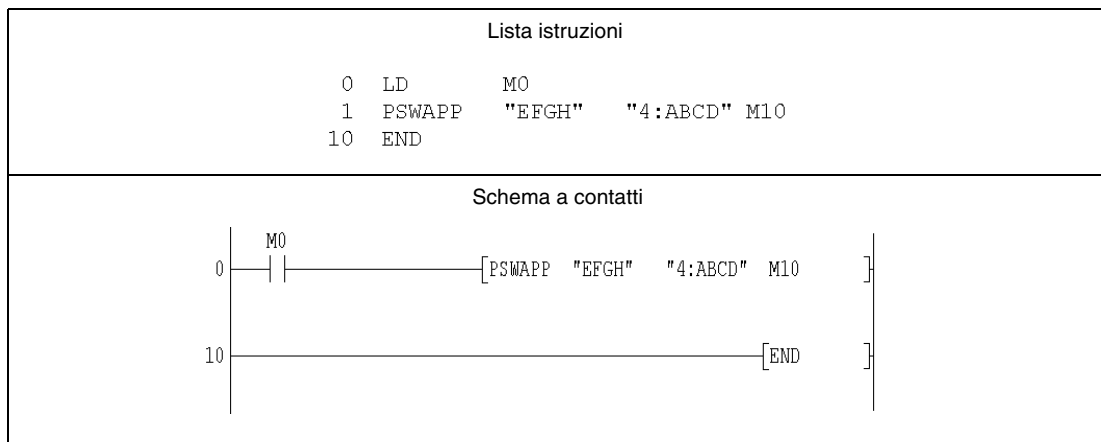
Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

- Il numero drive o il file specificati da s1 o s2 non esistono (codice di errore 2410).
- Il numero di drive specificato da s2 non è valido (codice di errore 4100).
- Spazio insufficiente nella memoria programmi (drive 0) per caricare il programma specificato (codice di errore 2413).
- Il programma specificato da s1 non è in standby o è in esecuzione (codice di errore 4101).

Esempio di programma

PSWAPP

Quando M0 è attivo nel programma di esempio seguente, il programma „EFGH.QPG“ viene cancellato dalla memoria programmi. Successivamente, il programma „ABCD.QPG“ viene caricato nella memoria programmi dal drive 4, e posto nello stato standby.



9.5 Istruzioni per trasferimento dati

9.5.1 RBMOV, RBMOV_P

CPU

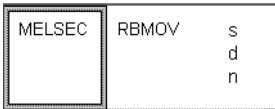
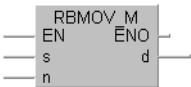
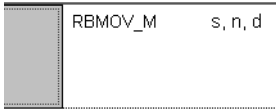
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					● ¹

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

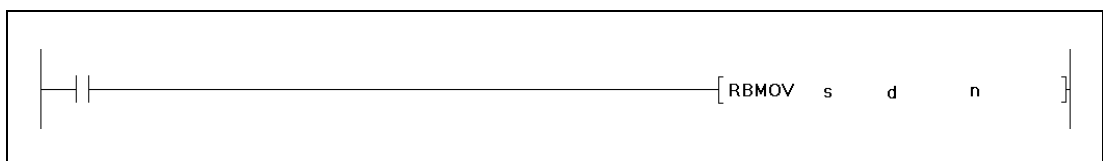
Operandi MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s	●	●	●	●	●	—	—	—	SM0	4	
d	●	●	●	●	●	—	—	—			
n	●	●	●	●	●	●	●	—			

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
		

GX Developer



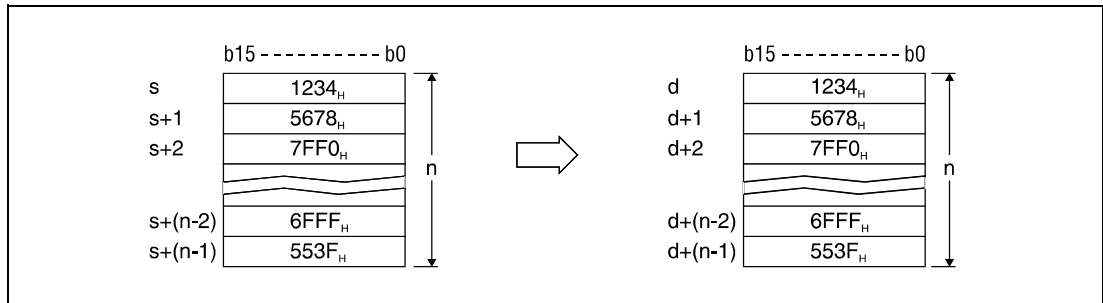
Variabili

Dati impostati	Significato	Tipo dati
s	Indirizzo del primo operando che memorizza i dati da trasferire	BIN 16-bit
d	Indirizzo iniziale degli operandi di destinazione	
n	Numero di dati da trasferire	

Funzioni Trasferimento blocchi di file registri ad alta velocità

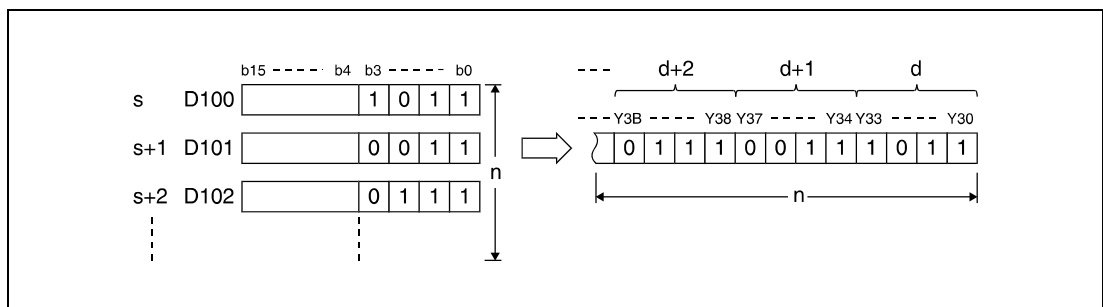
RBMOV/RBMOV P Trasferimento blocchi

L'istruzione RBMOV trasferisce "n" punti di dati a 16 bit a partire dall'operando specificato da s, verso l'area di "n" punti a partire dall'operando specificato da d.



Il trasferimento è possibile anche se gli operandi sorgente e destinazione sono sovrapposti. Il trasferimento verso operandi verso operandi con indirizzi più bassi inizia da s, mentre il trasferimento verso indirizzi più alti inizia da s+(n-1).

Se s è un operando a word e d è un operando a bit, il numero di bit determinati dall'indicazione di cifra dell'operando a bit divengono i bit oggetto dell'operando a word. Ad esempio, se d specifica „K1Y30“, il target sono i quattro bit più leggeri dell'operando a word specificato da s.



Se s e d specificano operandi a bit, il numero di cifre deve essere lo stesso per s e per d.

NOTA

Le istruzioni **RBMOV** e **RBMOVP** sono utili per il trasferimento in blocco di grosse porzioni dei dati del file registri con CPU System Q ad alte prestazioni. Con una Q02CPU, questa istruzione è simile all'istruzione **BMOV**. Il confronto di prestazioni fra le istruzioni **RBMOV** e **BMOV** è il seguente:

Q02HCPU, Q06HCPU, Q12HCPU, Q25HCPU				
Memoria target	RBMOV		BMOV	
	Tempo (μs) di trasferimento		Tempo (μs) di trasferimento	
	100 word	1000 word	100 word	1000 word
SRAM	56,30	367,77	44,37	393,14
RAM interna	44,37	393,14		
ROM Flash	29	308		

Q02CPU				
Memoria target	Istruzione RBMOV		Istruzione BMOV	
	Tempo (μs) di trasferimento		Tempo (μs) di trasferimento	
	100 word	1000 word	100 word	1000 word
SRAM	115,89	579,47	115,89	535,23
RAM interna				
ROM Flash				

Errori di esecuzione

Nei casi seguenti si verifica un errore di esecuzione ed il segnale di errore viene attivato:

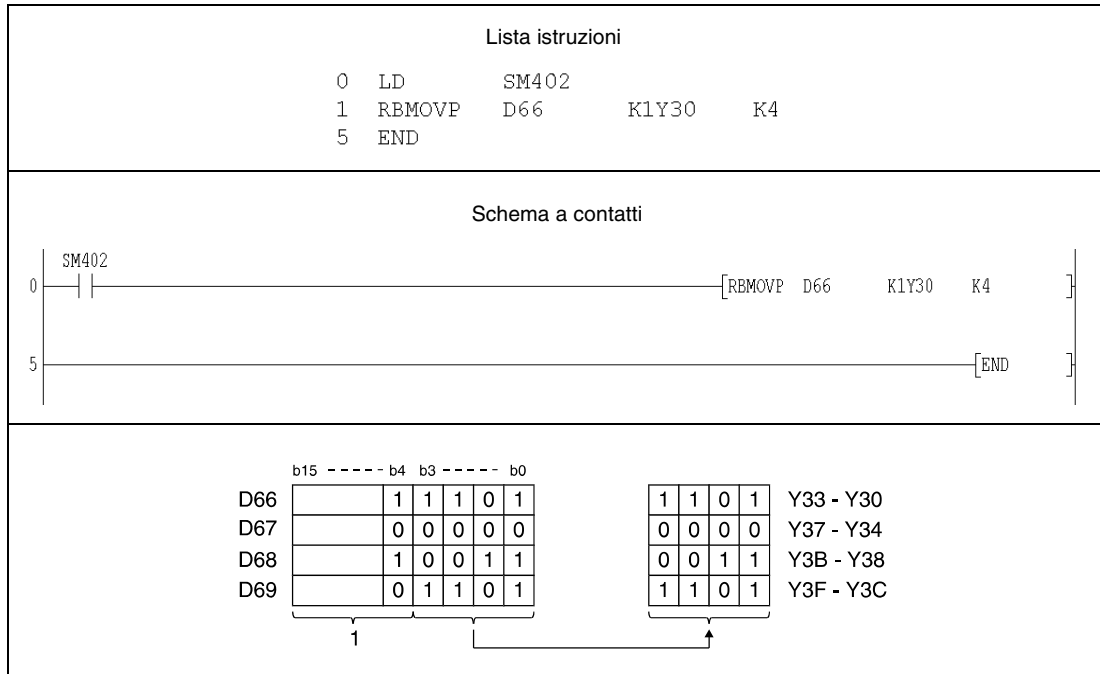
- Il campo di "n" operandi a partire da s supera il campo operandi ammesso (codice di errore 4101).
- Il file registri non è indicato sia da s che da d (codice di errore 4101).

Programma di esempio 1

RBMOV P

Il programma seguente trasferisce i quattro bit meno significativi (da b0 a b3) dei dati da D66 a D69 sulle uscite da Y30 a Y3F, sul fronte di salita di SM402. Il numero di dati (4 blocchi) è indicato in n.

La figura mostra la struttura dei bit prima e dopo il trasferimento.

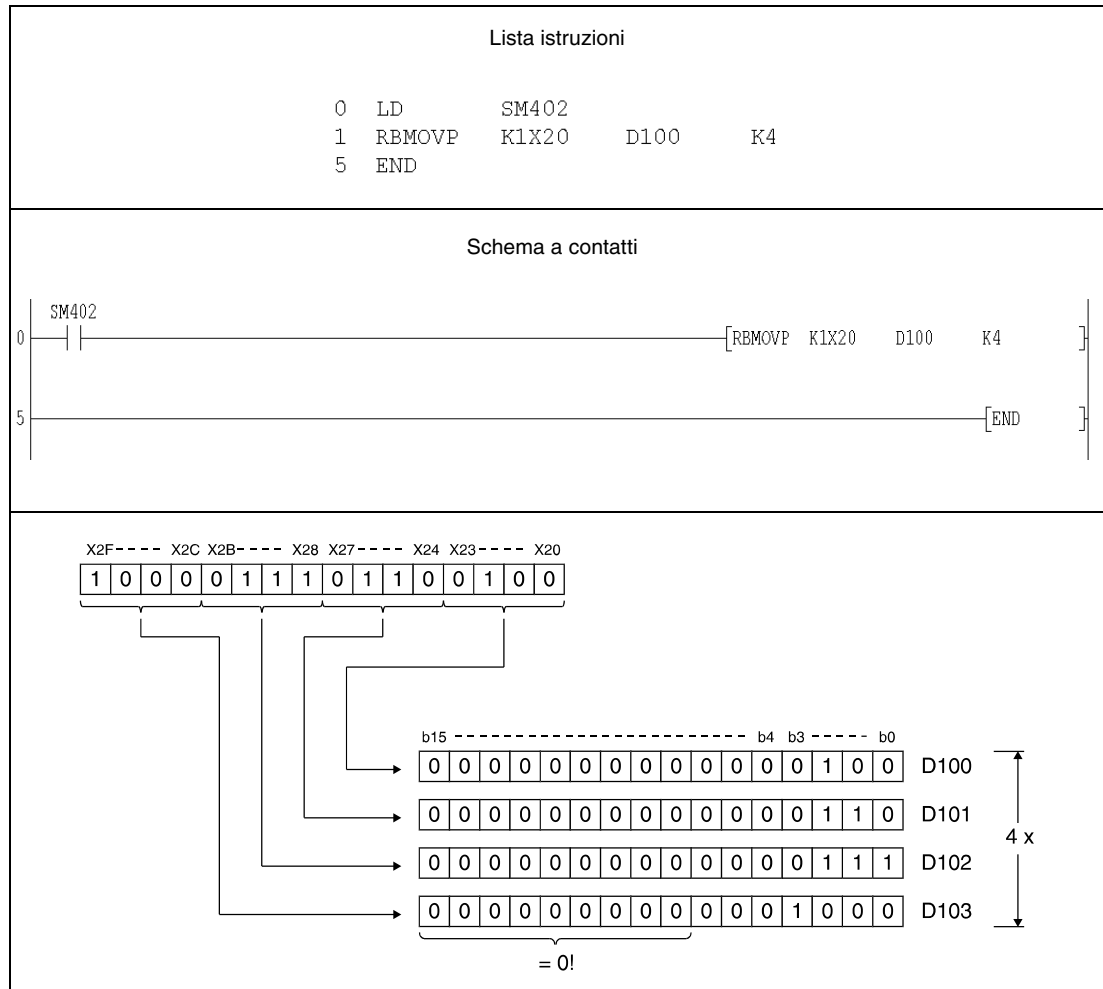


¹ Questi bit sono ignorati.

**Programma
di esempio 2** RBMOVP

Sul fronte di salita di SM402, il programma che segue trasferisce i dati da X20 a X2F in D100 fino a D103. Il numero di blocchi (4) da trasferire è indicato dalla costante K4.

La figura mostra la struttura dei bit prima e dopo il trasferimento.



9.6 Istruzioni per uso in un sistema multi CPU

9.6.1 S.TO, SP.TO

CPU

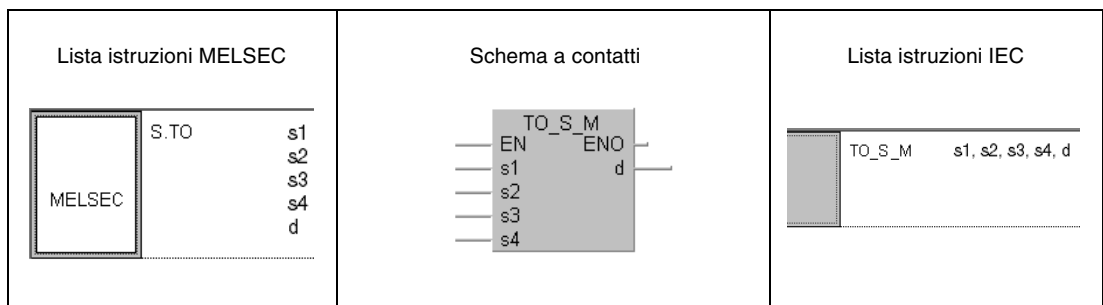
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					● ¹

¹ Solo per Q02CPU, Q02HCPU, Q06HCPU, Q12HCPU e Q25HCPU dalla versione funzionale B.

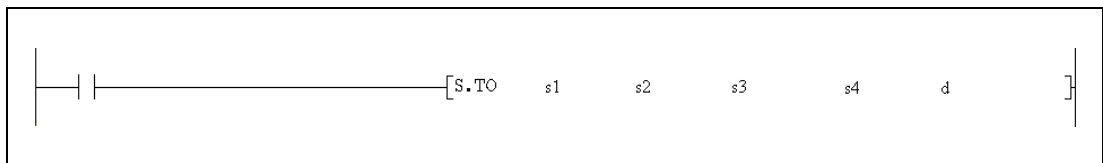
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	●	—	SM0	5
s2	—	●	●	—	—	—	—	●	—		
s3	—	●	●	—	—	—	—	—	—		
s4	—	●	●	—	—	—	—	●	—		
d	●	●	●	—	—	—	—	—	—		

GX IEC Developer



GX Developer



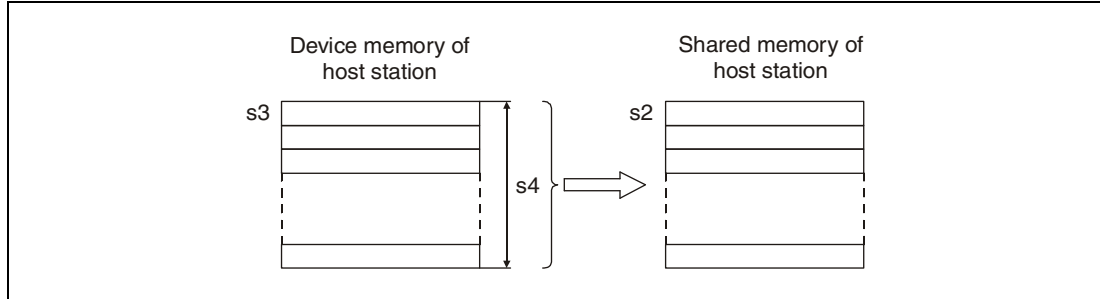
Variabili

Dati impostati	Significato	Tipo dati
s1	Primo indirizzo di I/O della CPU che esegue l'istruzione S.TO	BIN16-bit
s2	Primo indirizzo dell'area di memoria condivisa della CPU in cui scrivere (da 800 _H a 0FFF _H).	
s3	Indirizzo del primo operando dell'area in cui scrivere.	
s4	Numero dati da scrivere (da 1 a 256)	
d	Operando a bit attivato per una scansione al termine dell'esecuzione dell'istruzione.	Bit

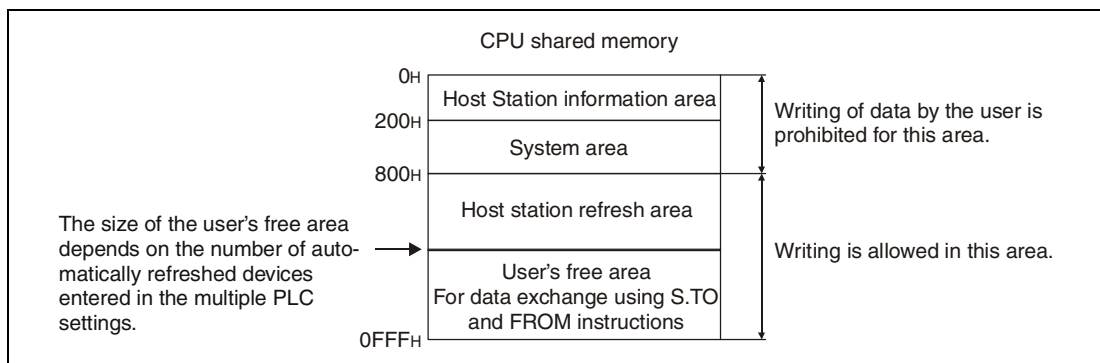
Funzioni Scrittura dati nella memoria condivisa della CPU**S.TO/SP.TO Scrittura dati**

L'istruzione S.TO scrive dati nell'area utente della memoria condivisa della CPU che esegue l'istruzione S.TO (stazione host). L'indirizzo di destinazione della memoria condivisa è specificato da s2. I dati originano da un'area operandi della stessa CPU, iniziando dall'operando specificato con s3. Il numero di word dati è specificato da s4.

L'istruzione S.TO non può essere usata per scrivere dati direttamente in una diversa CPU di un sistema multi CPU.



La memoria condivisa della CPU viene usata per scambiare dati con altre CPU di un sistema multi CPU. Il rinfresco automatico dell'area inizia dall'indirizzo 800H, seguito dall'area utente libera.



L'indirizzo iniziale di I/O della CPU viene determinato dallo slot in cui è inserito il modulo CPU. In s1 vengono inserite solo le prime tre cifre dell'indirizzo iniziale di I/O.

Slot dell'unità base	CPU	0	1	2
Numero di CPU del sistema multi CPU	1	2	3	4
Indirizzo iniziale di I/O	3E00	3E10	3E20	3E30
Contenuto di s1	3E0	3E1	3E2	3E3

Se il numero di punti da scrivere viene inserito come „0“, l'istruzione non viene elaborata e l'operando di completamento, specificato in d, non viene attivato.

NOTA

In un ciclo di scansione di ciascuna CPU, può essere eseguita solo una istruzione S.TO. Tuttavia, lo handshaking automatico assicura che venga elaborata solo l'istruzione richiamata per prima, nel caso che due o più istruzioni S.TO vengano abilitate simultaneamente.

Errori di esecuzione

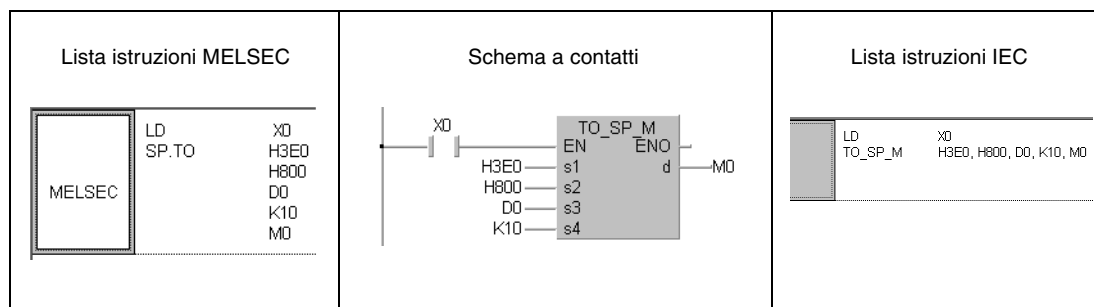
Nei casi seguenti, si verifica un errore di funzionamento, viene attivato il flag di errore, ed il codice di errore corrispondente viene memorizzato in SDO:

- Il numero di punti da scrivere specificato da s4 non è nel campo da 0 a 256 (codice di errore 4101).
- L'indirizzo iniziale della memoria condivisa della CPU specificata in s2 è superiore al campo di indirizzamento della memoria condivisa (codice di errore 4101).
- L'indirizzo iniziale della memoria condivisa della CPU specificata in s2 sommato al numero di punti da scrivere specificato in s4 è superiore al campo di indirizzamento della memoria condivisa (codice di errore 4101).
- L'indirizzo del primo operando (s3) che specifica l'area di memorizzazione, sommato al numero di punti da scrivere specificato in s4, supera il campo operando (codice di errore 4101).
- Il valore contenuto in s1 non è l'indirizzo iniziale di I/O della CPU che esegue l'istruzione S.TO (codice di errore 2107).
- Il numero contenuto in s1 non specifica un indirizzo iniziale di I/O corretto (3E0_H, 3E1_H, 3E2_H o 3E3_H)(codice di errore 4100).
- L'istruzione specificata non è corretta (codice di errore 4002).
- L'indirizzo degli operandi specificati è errata (codice di errore 4003).
- Operando specificato non valido (codice di errore 4002).

Esempio di programma

SP.TO

I dati contenuti in CPU1 nei registri dati da D0 a D9 vengono scritti nella memoria condivisa della stessa CPU, a partire dall'indirizzo 800_H quando X0 viene attivato.



9.6.2 FROM, FROMP

CPU

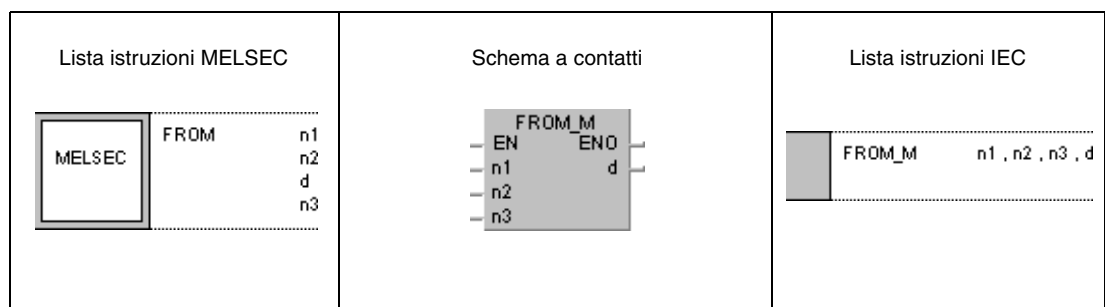
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					● ¹

¹ Solo per Q02CPU, Q02HCPU, Q06HCPU, Q12HCPU e Q25HCPU dalla versione funzionale B.

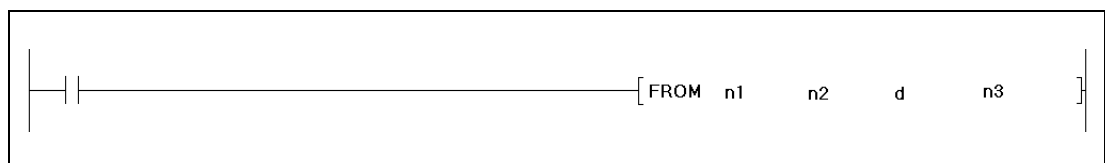
Operandi MELSEC Q

	Operandi utilizzabili									Errore	Passo
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word				U		
n1	—	●	●	●	●	●	●	●	●	SM0	5
n2	—	●	●	●	●	●	●	—	—		
d	—	●	●	—	—	—	—	—	—		
n3	—	●	●	●	●	●	●	—	—		

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
n1	Indirizzo iniziale di I/O della CPU che contiene i dati da leggere	BIN 16-bit
n2	Indirizzo iniziale dei dati da leggere nella memoria condivisa della CPU (da 800 _H a 0FFF _H).	
d	Indirizzo iniziale dell'area di memoria che riceve i dati letti	
n3	Numero word dati da leggere (da 1 a 6144)	

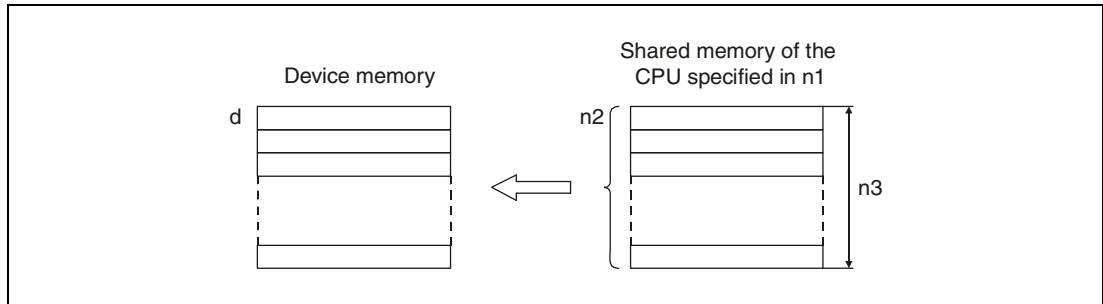
NOTA

Vedi il capitolo 7.8.1 per i dettagli sull'uso dell'istruzione FROM per la lettura di dati dal buffer di memoria dei moduli funzione speciali.

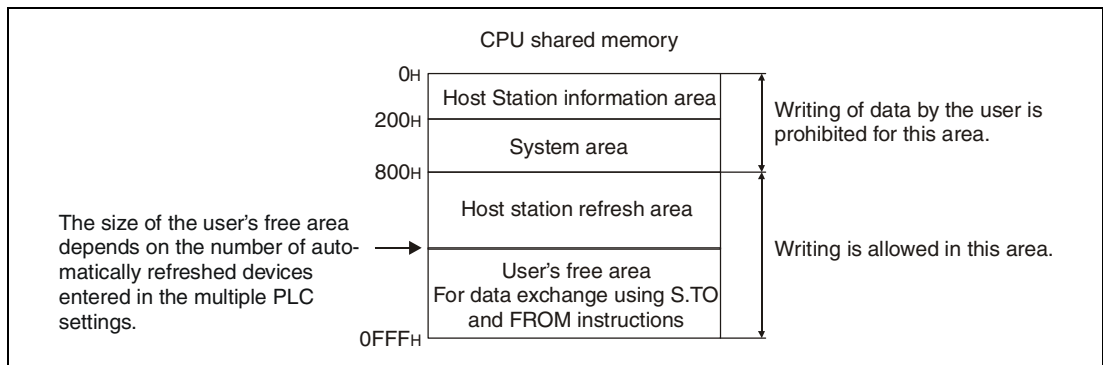
Funzioni **Letture dalla memoria condivisa di una diversa CPU**

FROM/FROMP **Letture word dati**

In un sistema multi CPU l'istruzione FROM viene usata per leggere word dati dall'area utente libera della memoria condivisa di un'altra CPU. L'indirizzo iniziale di questa CPU è specificato in n1. Inserire il numero di word da leggere in n3. L'indirizzo iniziale della memoria condivisa dell'altra CPU è specificato in n2. I dati verranno memorizzati nella CPU che esegue l'istruzione FROM, a partire dall'indirizzo dell'operando specificato da d.



La memoria condivisa della CPU viene usata per scambiare dati con altre CPU di un sistema multi CPU. Il rinfresco automatico dell'area inizia dall'indirizzo 800H, seguito dall'area utente libera.



L'indirizzo iniziale di I/O della CPU viene determinato dallo slot in cui è inserito il modulo CPU. In n1 vengono inserite solo le prime tre cifre dell'indirizzo iniziale di I/O.

Slot dell'unità base	CPU	0	1	2
Numero di CPU del sistema multi CPU	1	2	3	4
Indirizzo iniziale di I/O	3E00	3E10	3E20	3E30
Contenuto di n1	3E0	3E1	3E2	3E3

Il relé speciale SM390 si attiva dopo la lettura dei dati. SM390 si attiva se la CPU specificata in n1 è in stato reset. In questo caso non viene segnalato alcun errore.

L'elaborazione dell'istruzione non viene eseguita se il numero dei dati da leggere in n3 è „0“.

Errori di esecuzione

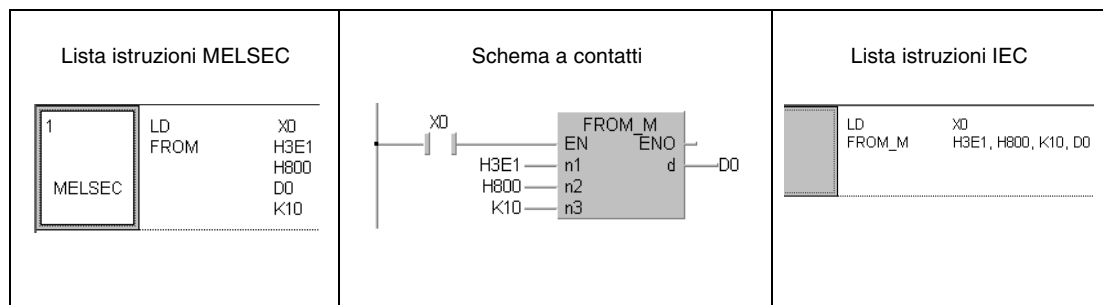
Nei casi seguenti, si verifica un errore di funzionamento, viene attivato il flag di errore, ed il codice di errore corrispondente viene memorizzato in SD0:

- L'indirizzo iniziale della memoria condivisa della CPU (n2) da cui vengono letti i dati è superiore al campo di indirizzamento della memoria condivisa (codice di errore 4101).
- L'indirizzo iniziale della memoria condivisa della CPU specificato in n2 sommato al numero di punti da scrivere (n3) è superiore al campo di indirizzamento della memoria condivisa (codice di errore 4101).
- L'indirizzo dell'operando di memorizzazione dati (d) sommato al numero di punti da leggere (n3) è superiore al campo dell'operando specificato (codice di errore 4101).
- L'indirizzo iniziale di I/O specificato in n1 è l'indirizzo iniziale di I/O della CPU che esegue l'istruzione FROM (codice di errore 2114).
- Non esiste nessun modulo CPU nella posizione specificata dall'indirizzo iniziale di I/O in n1 (codice di errore 2110).

Esempio di programma

FROM

Quando X0 è attivo, vengono lette 10 word dati dalla memoria condivisa della CPU 2, a partire dall'indirizzo 800_H. I dati vengono memorizzati nei registri dati da D0 a D9 della CPU che esegue l'istruzione FROM.



10 Istruzioni per Q4ARCPU

Due CPU Q4ARCPU possono formare un sistema di PLC ridondante in cui una CPU assume il controllo se l'altra CPU va in errore. In questo modo è possibile continuare il controllo senza interruzioni. Applicazioni tipiche di un PLC ridondante sono ad esempio centrali elettriche, industria chimica o il rifornimento idrico di comunità.

Le istruzioni seguenti sono disponibili solo su Q4ARCPU.

Funzione	Istruzione MELSEC in editor MELSEC	Istruzione MELSEC in editor IEC
Impostazione modalità di avviamento della CPU	S.STMODE	STMODE_S_M
Impostazione modo di commutazione delle CPU	S.CGMODE	CGMODE_S_M
Trasferimento dati a sistema in standby	S.TRUCK	TRUCK_S_M
Trasferimento in blocco di dati da e per la memoria buffer dei moduli funzione speciali	S.SPREF	SPRE_S_M

10.1 Istruzioni di impostazione di modo

10.1.1 STMODE

CPU

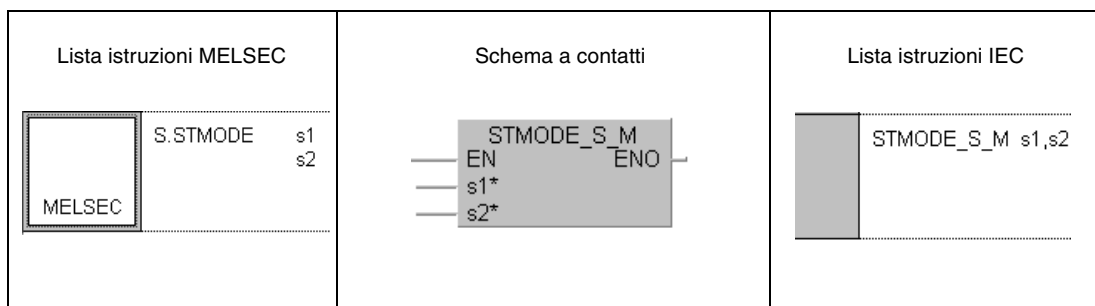
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				● ¹	

¹ Solo per Q4AR

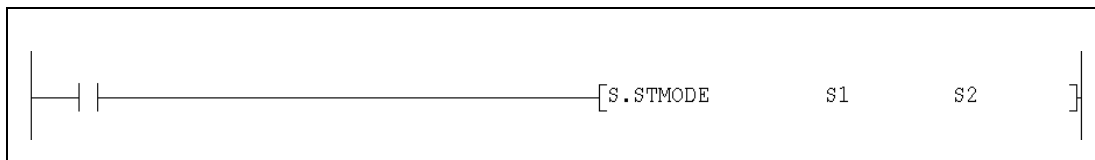
Operandi MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo Funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s1	—	—	—	—	—	—	—	●	—	—	—
s2	—	—	—	—	—	—	—	●	—	—	

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s1	Impostazione modalità di avviamento della CPU (0 = avviamento con inizializzazione, 1 = avviamento a caldo)	BIN 16-bit
s2	Tempo massimo di mancanza rete. Dopo questo tempo viene eseguito un avviamento a caldo.	BIN 16-bit

Funzioni **Impostazione modo operativo durante l'avviamento della CPU****STMODE** **Impostazione modo operativo**

Il contenuto di s1 seleziona se gli operandi della CPU vengono azzerati (modo inizializzazione) o meno (modo avviamento a caldo) quando viene acceso l'alimentatore del PLC.

Quando si specifica il modo avviamento a caldo, può essere eseguita una cancellazione dati e riavviamento se si verifica una mancanza rete per un tempo specificato. In questo caso, specificare il tempo di mancanza rete in s2.

Questa istruzione viene eseguita quando l'alimentatore viene acceso. L'istruzione viene eseguita indipendentemente dalla sua condizione di esecuzione. La condizione di esecuzione è inessenziale. Durante la normale elaborazione del programma, se la condizione di esecuzione diventa attiva, viene eseguita una istruzione NOP.

Ciascun sistema deve contenere una di queste istruzioni. In caso di programmi con file multipli, questa istruzione è necessaria in un solo file. Se esiste più di una istruzione, non può essere garantito un funzionamento corretto.

Il contenuto di s1 può essere 0 o 1:

0: Modo inizializzazione (azzeri gli operandi fuori della zona retentiva)

1: Modo avviamento a caldo (gli operandi non vengono azzerati, eccetto i registri indice ed i risultati operazione, che vengono azzerati). Inoltre viene attivato il relé speciale e precaricato il registro speciale SD.

Il tempo in s2 è specificato in secondi (da 0 a 65535). Specificando 0, il modo inizializzazione non può essere eseguito. Se viene impostato un numero superiore a 32767, impostarlo in notazione esadecimale.

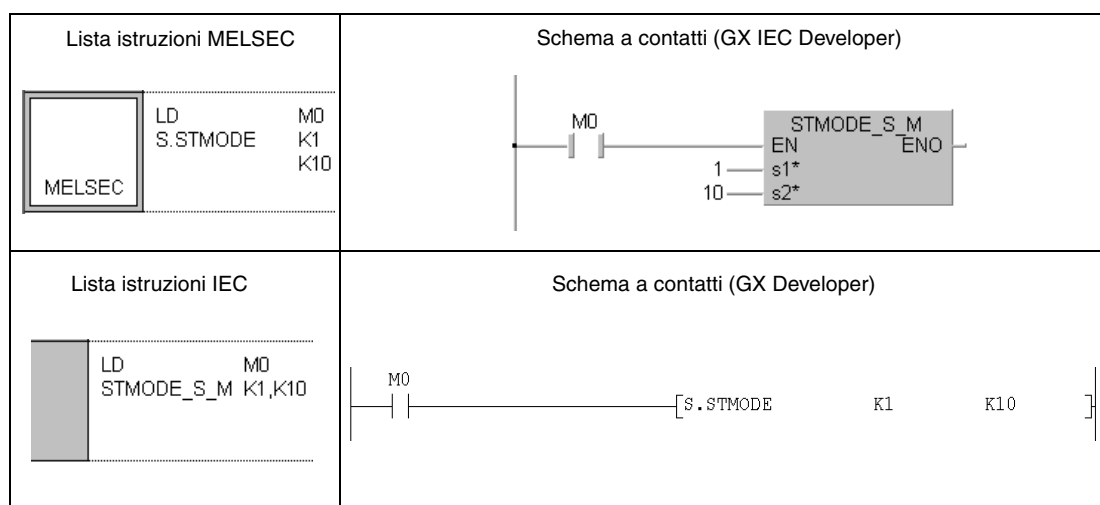
Errori di esecuzione

Nei casi seguenti, si verifica un errore di funzionamento, viene attivato il flag di errore SM0, ed un codice di errore viene memorizzato in SD0:

- Se viene impostato un valore oltre il campo ammesso in s1 o s2 (codice di errore: 4104).

Esempio di programma**STMODE**

Il programma seguente avvia la CPU con avviamento a caldo alla comparsa dell'alimentazione. Se l'alimentazione del PLC è stata interrotta per oltre 10 secondi, viene eseguito un avviamento con inizializzazione.



10.1.2 CGMODE

CPU

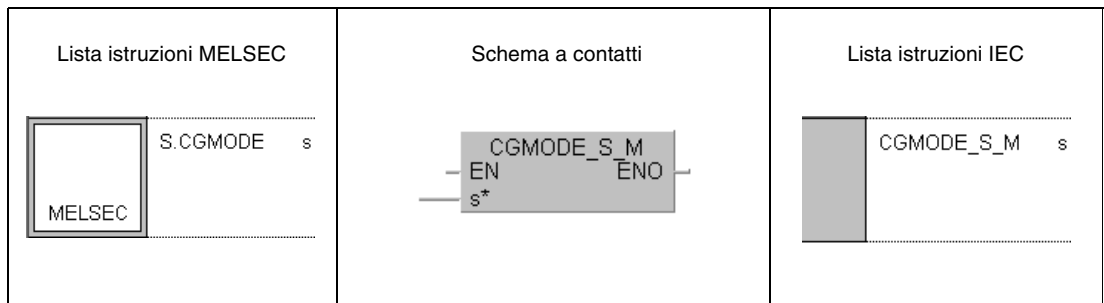
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				● ¹	

¹ Solo per Q4AR

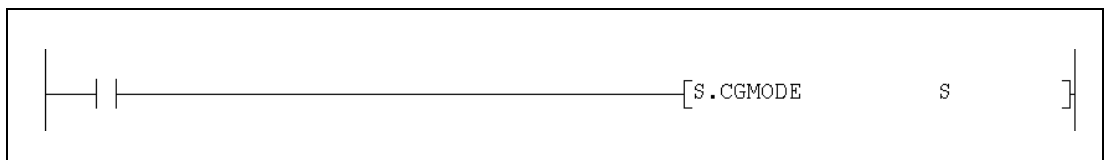
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo Funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s	—	—	—	—	—	—	—	●	—	—	

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Impostazione modo operativo	BIN 16-bit

Funzioni **Modalità durante al commutazione da sistema attivo a sistema in standby****CGMODE Impostazione modo operativo**

Questa istruzione specifica se gli operandi della CPU vengono o meno azzerati quando viene trasferito il controllo dal sistema attivo a quello in standby. La scelta avviene con il parametro s.

L'istruzione viene eseguita nel passaggio da STOP a RUN a seguito di una accensione. L'istruzione viene eseguita indipendentemente dalla sua condizione di esecuzione. La condizione d'ingresso dell'istruzione è irrilevante. Durante la normale elaborazione del programma, se la condizione di esecuzione diventa attiva, viene eseguita una istruzione NOP.

Un sistema può contenere solo una di queste istruzioni. Usare questa istruzione solo in caso di programmi su file multipli. Se esiste più di una istruzione, non può essere garantito un funzionamento corretto.

Il contenuto di s1 può essere 0 o 1:

0: Modo inizializzazione (azzeri gli operandi fuori della zona retentiva)

1: Avviamento a caldo (gli operandi e i risultati operazione non vengono azzerati come nel modo inizializzazione. Il relé speciale SM e il registro speciale SD vengono precaricati).

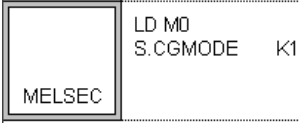
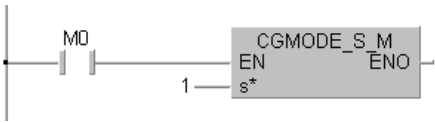
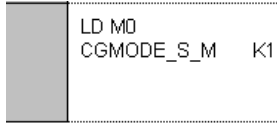

Errori di esecuzione

Nei casi seguenti, si verifica un errore di funzionamento, viene attivato il flag di errore SM0, ed un codice di errore viene memorizzato in SD0:

- Se viene specificato un valore diverso da 0 o 1 per s (codice di errore: 4104).

Esempio di programma**CGMODE**

Questo programma esegue un avviamento a caldo della CPU durante la commutazione da sistema attivo a sistema in standby.

<p>Lista istruzioni MELSEC</p>  <pre>LD M0 S.CGMODE K1</pre>	<p>Schema a contatti (GX IEC Developer)</p> 
<p>Lista istruzioni IEC</p>  <pre>LD M0 CGMODE_S_M K1</pre>	<p>Schema a contatti (GX Developer)</p> 

10.2 Istruzioni per trasferimento dati

10.2.1 TRUCK

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				● ¹	

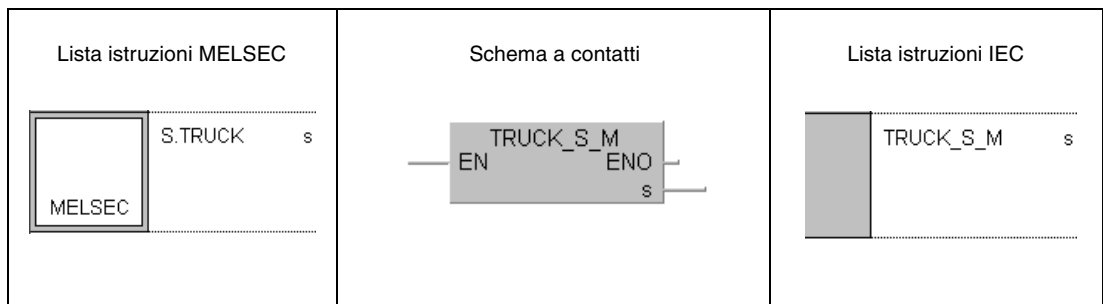
¹ Solo per Q4AR

Operandi MELSEC Q

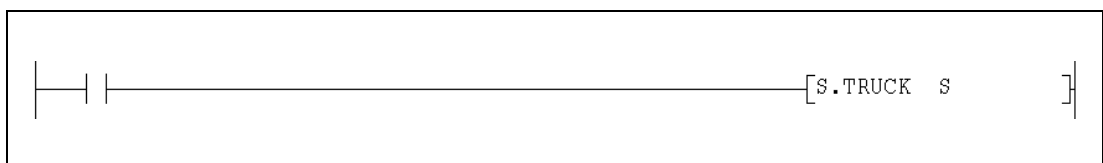
	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo Funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s	—	● ¹	●	—	—	—	—	—	—	—	—

¹ Solo operandi retentivi

GX IEC Developer



GX Developer



Variabili

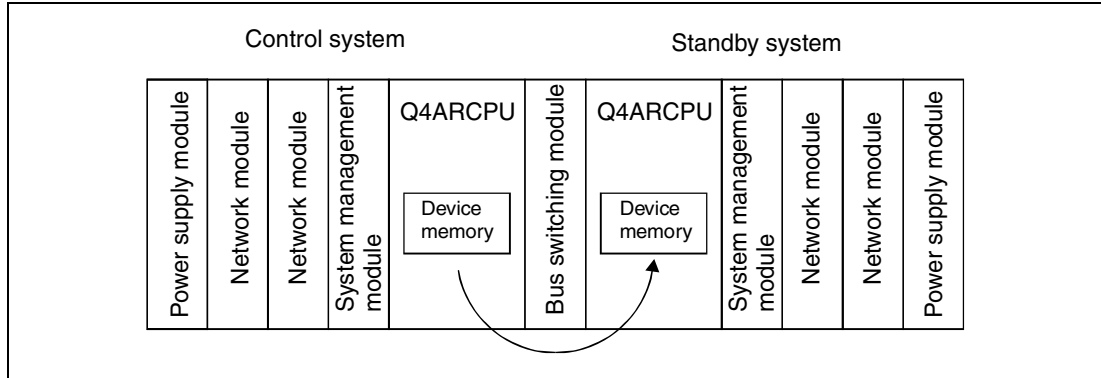
Dati impostati	Significato	Tipo dati
s	Operando intestazione blocco parametri	BIN 16-bit

Funzioni Trasferimento dati verso la CPU in standby di un sistema PLC ridondante

TRUCK Istruzioni di trasporto dati

La funzione TRUCK trasmette dati dal dispositivo di memoria del sistema Q4ARCPU attivo alla memoria del sistema Q4ARCPU in standby.

In un sistema di PLC ridondante, durante l'elaborazione dell'istruzione END, avviene il trasferimento dei dati dalla memoria della Q4ARCPU attiva verso la memoria della CPU (anche in questo caso Q4ARCPU) in standby. Un blocco di parametri contenente numero e tipo dei dati da trasferire è contenuto a partire dall'operando specificato da s.



Questa istruzione deve essere utilizzata una sola volta in un sistema. Usare questa istruzione solo in un file, in caso di programmi su file multipli. Se esiste più di una istruzione, non può essere garantito un funzionamento corretto.

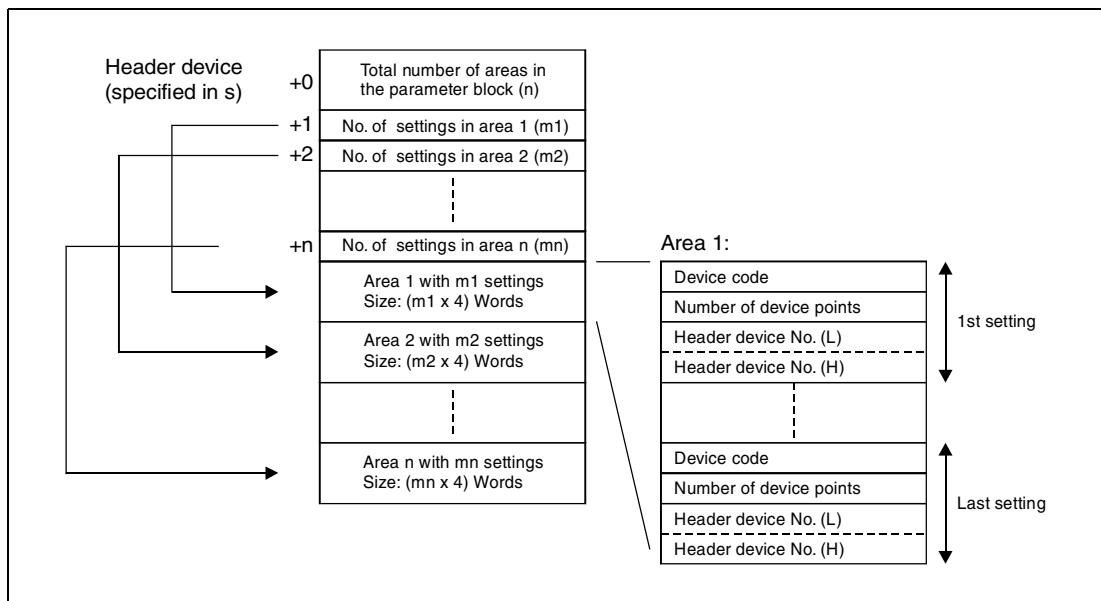
La Q4ARCPU legge il contenuto del blocco di parametri all'accensione o dopo un reset. (Quindi, se il contenuto del blocco di parametri viene modificato, è necessario riavviare il sistema.) Il blocco parametri è composto da diverse zone. Il trasferimento dei dati in ciascuna zona viene controllato tramite uno dei relé speciali da SM1520 a SM1583.

I dati specificati per la prima zona vengono trasferiti nel sistema in standby quando SM1520 è attivo, SM1521 invia i dati della seconda zona, ecc.

NOTA

Con l'istruzione TRUCK si possono usare gli stessi trigger di trasmissione (da SM1520 a SM1583) usati per l'istruzione SPREF.

Il blocco parametri è configurato come segue:



Contenuto del blocco parametri:

- Numero totale delle zone (n)
Il blocco parametri è una collezione di diverse zone di parametrizzazione. Questo insieme indica quante zone sono comprese nel blocco parametri.
- Numero di impostazioni in ciascuna zona (da m1 a mn).
In una zona si possono avere più impostazioni. Ciascuna impostazione è composta dal codice operando, dal numero di operandi e dall'indirizzo del primo operando.
- Zone di impostazione
Ciascuna impostazione occupa 4 word:
word 1: Codice operando (vedi tabella seguente)

Operando	Codice	Operando	Codice	Operando	Codice	Operando	Codice
X	0	B	5	C ¹	10	Z	15
Y	1	F	6	D	11	SB	16
M	2	V	7	W	12	SW	17
L	3	ST	8	R	13	SM	18
S	4	T ¹	9	ZR	14	SD	19

¹ Con i timer (T) e contatori (C) sono compresi contatto, bobina e valore attuale.

NOTA

Gli operandi specificati come locali non vengono trasferiti.

word 2: Numero operandi

Il dato può essere impostato sia in decimale che in esadecimale. Gli operandi a bit sono impostati a multipli di 16.

word 3 e 4: Indirizzo primo operando, leggero (L) e pesante (H)

Il dato viene impostato su due word, sia in decimale che in esadecimale. Gli operandi a bit vengono specificati con 0 o con multipli di 16 (ad es. 0, 16, 32, ...).

NOTE

Nell'impostazione dei blocchi di parametrizzazione, osservare le seguenti limitazioni:

- un blocco parametri può specificare al massimo 64 zone ($n \leq 64$).
- Il numero totale di impostazioni non deve superare 2048 ($m1+m2+...mn \leq 2048$).
- Se il numero di impostazioni (da m1 a mn) è 0, il numero delle zone viene impostato = 0. Impostando a 0 il numero delle zone, consente di non inserire la zona nel trasferimento.
- Il numero massimo di punti che si possono trasferire durante l'elaborazione di END di una scansione è 48k word. Se questo numero viene superato, si attiva un errore e il trasferimento non può essere eseguito.
- Se l'oggetto del trasferimento è un operando a bit, impostare numero di punti e indirizzo iniziale dell'operando con multipli di 16.
- Se l'oggetto del trasferimento è un timer o contatore, usare la formula seguente per calcolare il numero effettivo di operandi da trasportare.

$$\text{Numero di punti dell'operando} = \text{numero di punti specificati} \times (1 + 1/8)$$

Il valore "1" presente in parentesi si riferisce al dato a word del valore attuale, mentre la frazione „1/8“ rappresenta l'informazione a bit del contatto o della bobina.

Modi di trasferimento

Tramite il relé speciale SM1518 si possono selezionare due tipi di trasferimento. La selezione diventa valida dopo l'esecuzione di END al termine della scansione che imposta SM1518 ON o OFF.

a) Trasferimento con attesa (SM1518 = 0)

Se il sistema in standby sta usando la memoria di trasferimento con un trasferimento in corso, il sistema attivo esegue il processo di trasferimento dopo aver atteso che la CPU in standby sia pronta. Questo tempo di attesa aumenta il tempo di scansione del sistema attivo.

b) Trasferimento continuo (SM1518 = 1)

Se il sistema in standby sta utilizzando la memoria di trasferimento durante un trasferimento, il sistema attivo continua con l'elaborazione dell'istruzione END senza trasferire dati. Durante la ripetizione del trasferimento non vengono accettate ulteriori richieste di trasferimento. La CPU del sistema attivo non genera alcun tempo di attesa, per cui il tempo di scansione non viene aumentato.

Flag di fine trasferimento

Quando l'elaborazione del trasferimento di un blocco specificato è terminata, viene attivato un relé speciale (da SM1712 a SM1775) per ciascuna zona per la durata di una scansione. (Zona 1: SM1712, zona 2: SM1713 zona 64: SM1775)

Errori di esecuzione

Nei casi seguenti, si verifica un errore di funzionamento, viene attivato il flag di errore SM0, ed un codice di errore viene memorizzato in SD0:

- Il file del file registri non esiste, anche se il file registri R è stato specificato nel blocco parametri. (Codice di errore: 2402)
- Se viene specificato un valore oltre il campo ammesso. (Codice di errore: 4104)
- Se il numero di punti dell'operando da trasferire supera 38k word. (Codice di errore: 4104)

Esempio di programma TRUCK

Lo stato dei relé da M0 a M95 e da M320 a M639, oltre al contenuto dei registri dati da D0 a D29 e da D600 a D699 viene trasmesso verso il sistema in standby. Il blocco parametri inizia da R100 ed è composto da due zone: nella prima zona vengono specificati i relé interni, mentre la seconda zona viene usata per trasferire i registri dati. L'invio di queste zone viene attivato dai relé speciali SM1520 e SM1521.

Blocco parametri		Significato	Nota	
Operando	Contenuto			
R100	2	Numero zone	—	
R101	2	Impostazioni zona 1	—	
R102	2	Impostazioni zona 2	—	
R103	2	Codice operando (2 = M)	Zona 1	Impostazione 1
R104	96	Numero operandi		
R105	0	Indirizzo iniziale operando (M0)		
R106	0			
R107	2	Codice operando (2 = M)		Impostazione 2
R108	320	Numero operandi		
R109	320	Indirizzo iniziale operando (M320)		
R110	0		Zona 2	Impostazione 1
R111	11	Codice operando (11 = R)		
R112	30	Numero operandi		
R113	0	Indirizzo iniziale operando (D0)		
R114	0			
R115	11	Codice operando (11 = R)		Impostazione 2
R116	100	Numero operandi		
R117	600	Indirizzo iniziale operando (D600)		
R118	0			

<p>Lista istruzioni MELSEC</p> <pre> MELSEC LD M0 OUT SM1520 OUT SM1521 MELSEC LD M10 S.TRUCK R100 </pre>	<p>Schema a contatti (GX IEC Developer)</p>
<p>Lista istruzioni IEC</p> <pre> LD M0 ST SM1520 ST SM1521 LD TRUCK_S_M M10 R100 </pre>	<p>Schema a contatti (GX Developer)</p>

10.2.2 SPREF

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				● ¹	

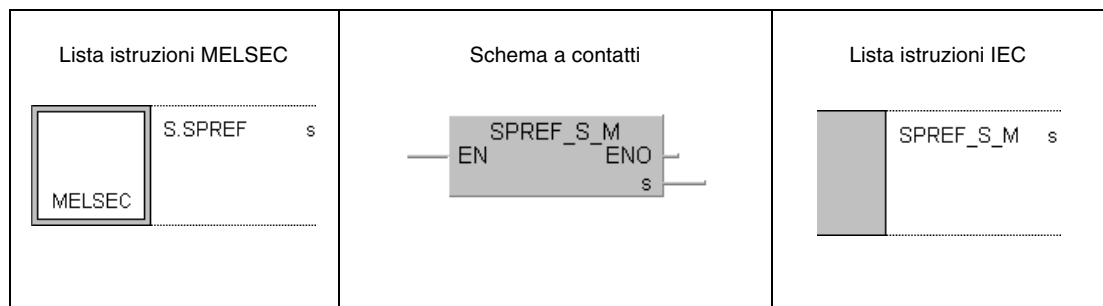
¹ Solo per Q4AR

Operandi MELSEC Q

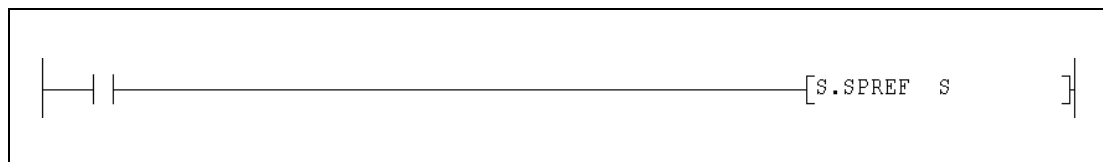
s	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo Funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
—	● ¹	●	—	—	—	—	—	—	—	—	

¹ Solo operandi retentivi.

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
s	Operando intestazione blocco parametri	BIN 16-bit

Funzioni

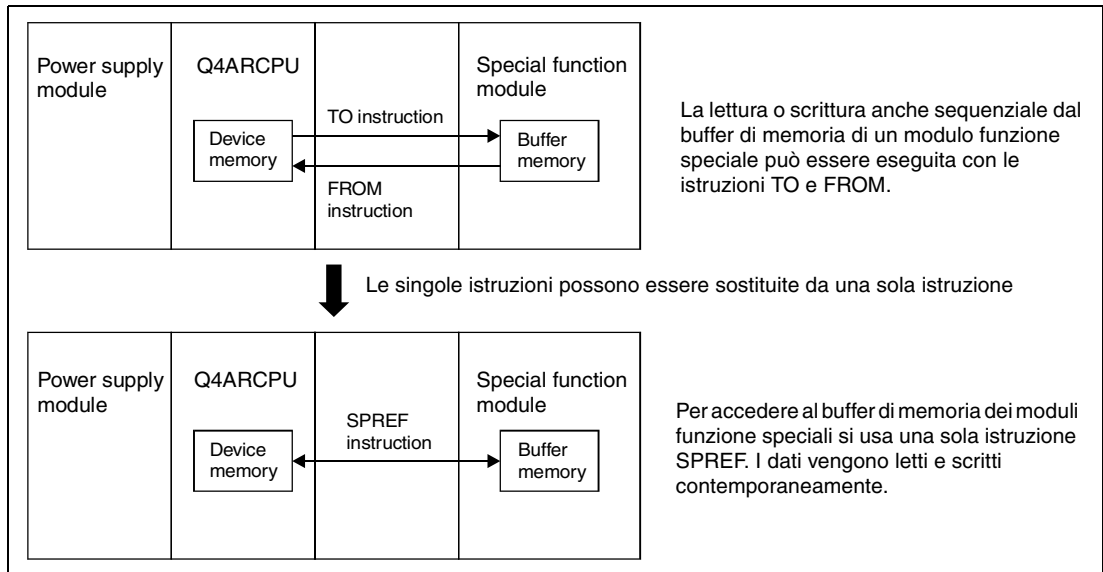
Rinfresco in blocco buffer di memoria

S.SPREF Rinfresco buffer di memoria

Con l'istruzione SPREF il contenuto del buffer di memoria di uno o più moduli funzione speciali viene letto o scritto in blocco.

NOTA

L'istruzione per il rinfresco in blocco del buffer di memoria non può essere eseguito per i moduli funzione speciali o le stazioni di I/O remoto nelle reti MELSECNET (II), /B, /10 o MELSECNET/MINI-S3.



L'indirizzo iniziale del blocco parametri contenente le impostazioni per il trasferimento dati è contenuto in s.

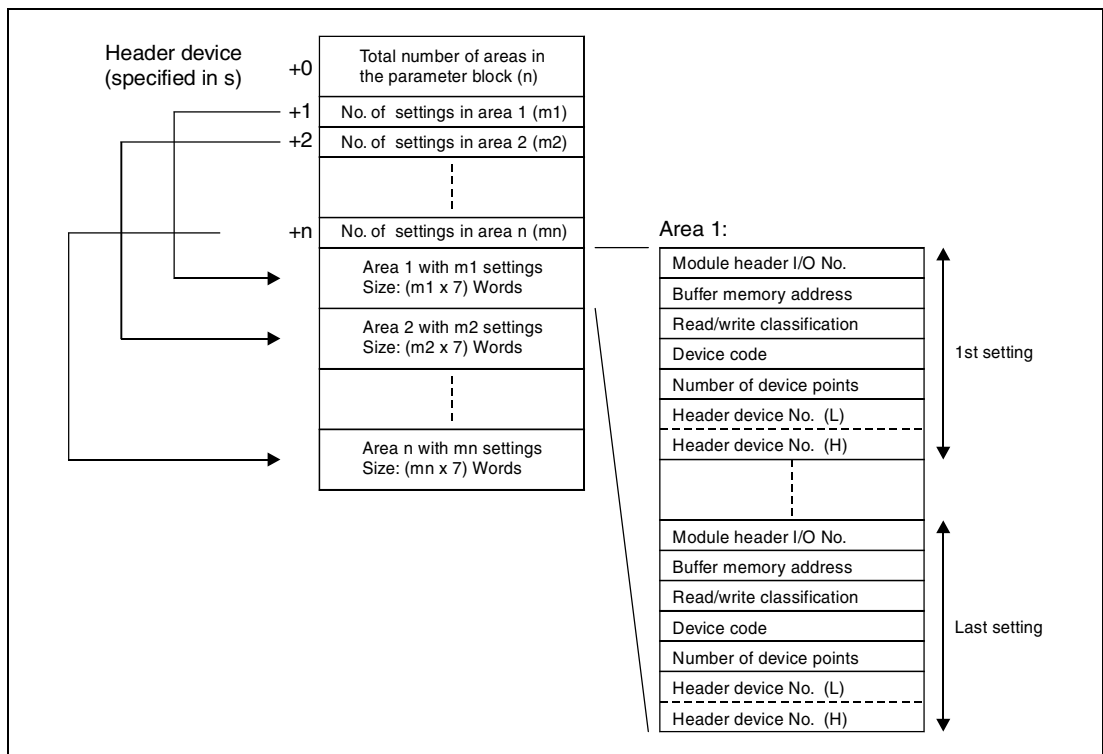
Il contenuto del blocco parametri deve essere definito prima dell'esecuzione dell'istruzione SPREF. Il blocco parametri è composto da diverse zone. Il trasferimento dei dati in ciascuna zona viene controllato tramite uno dei relé speciali da SM1520 a SM1583.

I dati specificati per la prima zona vengono letti o scritti quando SM1520 è attivo, SM1521 trasferisce i dati della seconda zona, ecc. I relé speciali da SM1520 a SM1583 devono essere attivati prima dell'esecuzione dell'istruzione SPREF.

NOTA

Con l'istruzione SPREF si possono usare gli stessi trigger di trasmissione (da SM1520 a SM1583) usati per l'istruzione TRUCK.

Il blocco parametri è configurato come segue:



Contenuto del blocco parametri:

- Numero totale delle zone (n)
Il blocco parametri è una collezione di diverse zone di parametrizzazione. Ciascuna di queste zone contiene informazioni che specificano lettura/scrittura, tipo dispositivo di memoria, numero di punti, indirizzo iniziale, ecc. Questo specifica quante zone sono contenute nel blocco parametri.
- Numero di impostazioni in ciascuna zona (da m1 a mn).
In una zona si possono avere più impostazioni. Ciascuna impostazione è composta dalle voci seguenti.
- Zona di impostazione
Ciascuna impostazione occupa 7 word della zona:

word 1: Indirizzo iniziale di I/O del modulo

Specifica l'indirizzo iniziale di I/O del modulo funzione speciale indicato. L'impostazione avviene con le prime due cifre, mentre il numero viene espresso come numero esadecimale a 3 cifre. (Esempio: L'indirizzo iniziale di I/O per X/Y100 viene inserito come 10H).

word 2: Indirizzo buffer di memoria

Imposta l'indirizzo del buffer di memoria in decimale o esadecimale.

word 3: lettura/scrittura

La specifica di lettura/scrittura indica il tipo di operazione da effettuare sul buffer di memoria. 0 = Lettura (dal buffer di memoria alla CPU), 1 = Scrittura (dalla CPU al buffer di memoria)

word 4: Codice operando (vedi tabella seguente)

Operando	Codice	Operando	Codice	Operando	Codice	Operando	Codice
X	0	B	5	C ¹	10	Z	15
Y	1	F	6	D	11	SB	16
M	2	-	-	W	12	SW	17
L	3	ST	8	R	13	SM	18
-	-	T ¹	9	ZR	14	SD	19

¹ Con i timer (T) e contatori (C) viene compreso il solo valore attuale.

word 5: Numero operandi

Il dato può essere impostato sia in decimale che in esadecimale. Gli operandi a bit sono impostati a multipli di 16.

word 6 e 7: Indirizzo primo operando, leggero (L) e pesante (H)

Il dato viene impostato su due word, sia in decimale che in esadecimale. Gli operandi a bit vengono specificati con 0 o con multipli di 16 (ad es. 0, 16, 32, ...).

NOTE

Nell'impostazione dei blocchi di parametrizzazione, osservare le seguenti limitazioni:

- un blocco parametri può specificare al massimo 64 zone ($n \leq 64$).
- Il numero totale di impostazioni non deve superare 2048 ($m1+m2+...mn \leq 2048$).
- Se il numero di impostazioni (da m1 a mn) è 0, il numero delle zone viene impostato = 0. Impostando a 0 il numero delle zone, consente di non inserire la zona nel trasferimento.

Errori di esecuzione

Nei casi seguenti, si verifica un errore di funzionamento, viene attivato il flag di errore SM0, ed un codice di errore viene memorizzato in SD0:

- Se viene specificato un valore oltre il campo ammesso. (Codice di errore: 4104)

Esempio di programma SPREF

Il programma seguente trasferisce dati fra la CPU e il buffer di memoria di due moduli funzione speciali. A ciascun modulo corrisponde una zona nel blocco parametri, contenuto nel file registri a partire da R100:

- Zona 1: Comunicazione con il modulo funzione speciale con l'indirizzo iniziale di I/O X/Y20

Le informazioni di questa zona sono elaborate quando SM1520 è attivo.

Il contenuto del buffer di memoria (indirizzi da 0 a 3) viene letto e memorizzato nel file registri da R0 a R3.

Il contenuto del file registri R10 e R11 viene scritto nel buffer di memoria agli indirizzi 10 e 11 del modulo funzione speciale.

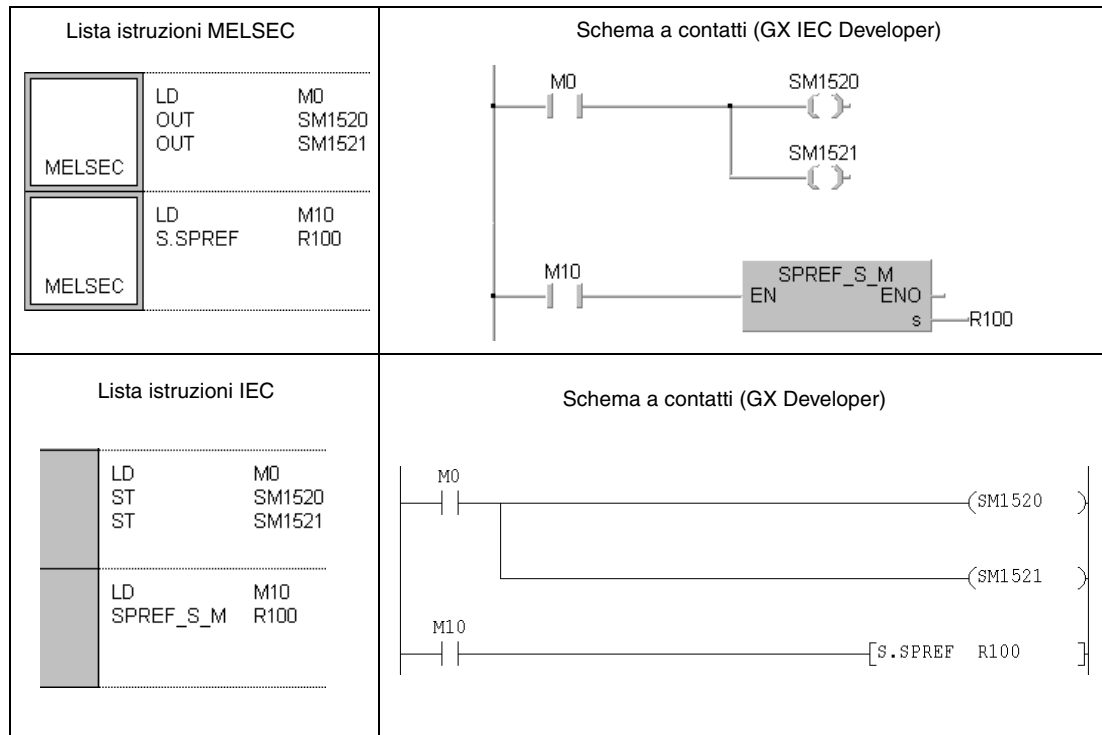
- Zona 2: Comunicazione con il modulo funzione speciale con l'indirizzo iniziale di I/O X/Y100

Le informazioni di questa zona sono elaborate quando SM1520 è attivo.

Il contenuto del buffer di memoria (indirizzi da 110 a 119) viene letto e memorizzato nei registri dati da D110 a D113.

Il blocco parametri per questo esempio contiene le seguenti costanti:

Blocco parametri		Significato	Nota	
Operando	Contenuto			
R100	2	Numero zone	—	
R101	2	Impostazioni zona 1	—	
R102	1	Impostazioni zona 2	—	
R103	2	Indirizzo iniziale di I/O del modulo (X/Y20)	Zona 1	Impostazione 1
R104	0	Indirizzo iniziale buffer di memoria		
R105	0	Lettura/scrittura (0 = lettura)		
R106	13	Codice operando (13 = R)		
R107	4	Numero operandi		
R108	0	Indirizzo iniziale operando (R0)		
R109	0			
R110	2	Indirizzo iniziale di I/O del modulo (X/Y20)		Impostazione 2
R111	10	Indirizzo iniziale buffer di memoria		
R112	1	Lettura/scrittura (1 = scrittura)		
R113	13	Codice operando (13 = R)		
R114	2	Numero operandi		
R115	10	Indirizzo iniziale operando (R10)		
R116	0			
R117	10	Indirizzo iniziale di I/O del modulo (X/Y100)	Zona 2	Impostazione 1
R118	110	Indirizzo iniziale buffer di memoria		
R119	0	Lettura/scrittura (0 = lettura)		
R120	11	Codice operando (11 = D)		
R121	10	Numero operandi		
R122	110	Indirizzo iniziale operando (D110)		
R123	0			



11 Istruzioni per moduli funzione speciali

debug	Funzione
Istruzioni per moduli di comunicazione seriale	Lettura dati ricevuti in un programma di interruzione; Lettura, registrazione o cancellazione di trame utente Trasmissione dati usando trame utente
Istruzioni per modulo interfaccia PROFIBUS/DP	Lettura o scrittura dati da e per il buffer di memoria di un modulo interfaccia PROFIBUS/DP.
Istruzioni per moduli interfaccia ETHERNET	Scrittura e lettura dati per e da un buffer fisso; Apertura a chiusura di connessioni, cancellazione codici di errore; Reinizializzazione di un modulo interfaccia ETHERNET
Istruzioni per MELSECNET/10	Impostazione stazioni per reti duplex
Istruzioni per CC-Link	Impostazione parametri. Impostazione dei parametri di rinfresco automatico Lettura dati dal buffer di memoria di una stazione collegata a CC-Link o dalla CPU del PLC di questa stazione Scrittura dati nel buffer di memoria di una stazione collegata a CC-Link o nella CPU del PLC di questa stazione; Lettura e scrittura da e per il buffer di memoria ad aggiornamento automatico

11.1 Istruzioni per moduli di comunicazione seriale

Funzione	Istruzione MELSEC in editor MELSEC	Istruzione MELSEC in editor IEC
Lettura dati ricevuti da una QJ71C24 in un programma di interruzione;	Z.BUFRCVS	BUFRCVS_M
Lettura di trame utente registrate	G.GETE	GETE_M
	GP.GETE	GETEP_M
Registrazione o cancellazione di trame utente registrate	G.PUTE	PUTE_M
	GP.PUTE	PUTEP_M
Trasmissione di trame utente	G.PRR	PRR_M
	GP.PRR	PRRP_M

11.1.1 BUFRCVS


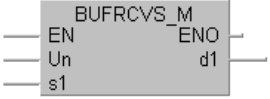

CPU

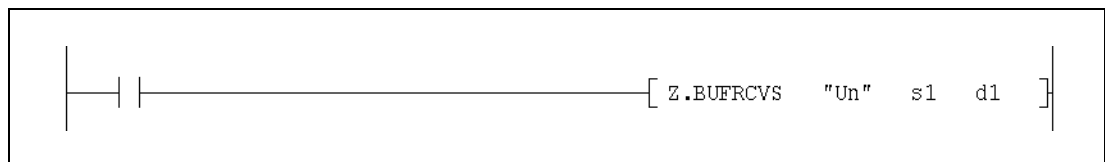
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					●

Operandi
MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	—	SM0	
d1	●	●	●	—	—	—	—	—	—		

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
 <p>Z.BUFRCVS "Un" s1 d1</p>		 <p>BUFRCVS_M "Un", s1, d1</p>

GX
Developer

Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati		
„Un“	Indirizzo iniziale di I/O del modulo di comunicazione seriale (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come „U10“)	da 0 a FE _H	Utente	BIN 16-bit		
s1	Numero canale di ricezione 1: Canale 1 (CH1) 2: Canale 2 (CH2)	1 o 2				
d1	Indirizzo iniziale dell'operando che memorizza i dati ricevuti			Indirizzo		
	Dati impostati	Significato	Descrizione		Campo	Contenuto inserito da
	(d1)+0	Lunghezza dati	Lunghezza dei dati ricevuti L'unità (byte o word) è impostata nei parametri.		—	Sistema
(d1)+1 a (d1)+n	Dati ricevuti	I dati letti dalla zona di ricezione del buffer di memoria vengono inseriti sequenzialmente in ordine crescente in questa zona.				

Funzioni **Letture dei dati ricevuti dalla QJ71C24**

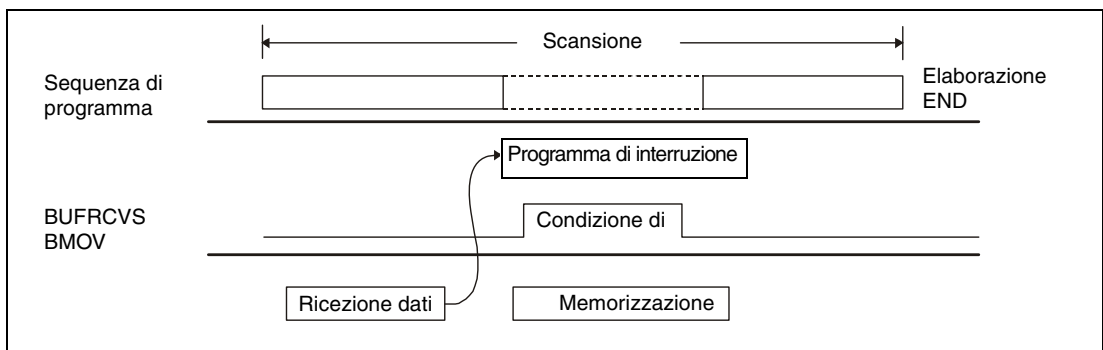
BUFRCVS Lettura dati

L'istruzione BUFRCVS legge i dati inviati da un dispositivo esterno verso il modulo di comunicazione QJ71C24, contenuti nel buffer di memoria del modulo, e li memorizza nel modulo CPU.

L'istruzione BUFRCVS può specificare l'indirizzo della zona di ricezione del buffer di memoria e leggere i dati relativi, inviandoli nell'area specificata da d1.

Al termine del trasferimento dati, la richiesta ricezione dati (X3/XA) o il segnale di rilevamento ricezione anomala (X4/XB) vengono disattivati automaticamente. Non è necessario disattivare il segnale di completamento ricezione (Y1/Y8) se i dati ricevuti vengono letti con l'istruzione BUFRCVS.

L'istruzione BUFRCVS viene usata da un programma di interruzione e la sua elaborazione viene completata in una scansione. La figura seguente mostra la temporizzazione durante l'esecuzione di una istruzione BUFRCVS:



NOTE

Se i dati ricevuti vengono letti con l'istruzione BUFRCVS da un programma di interruzione, i dati della stessa interfaccia non possono essere letti anche dal programma principale. Per questo, l'istruzione BUFRCVS non può essere usata con le seguenti istruzioni:

- istruzione INPUT
- istruzione BIDIN
- istruzione FROM in combinazione con i segnali di ingresso/uscita del modulo di comunicazione

Le istruzioni BUFRCVS e CSET non possono essere eseguite contemporaneamente.

L'area specificata con d1 nella CPU del PLC deve essere grande a sufficienza per contenere tutti i dati inviati dal modulo esterno. Se questa area è troppo piccola, i dati che non possono essere memorizzati vengono persi.

Errori di esecuzione

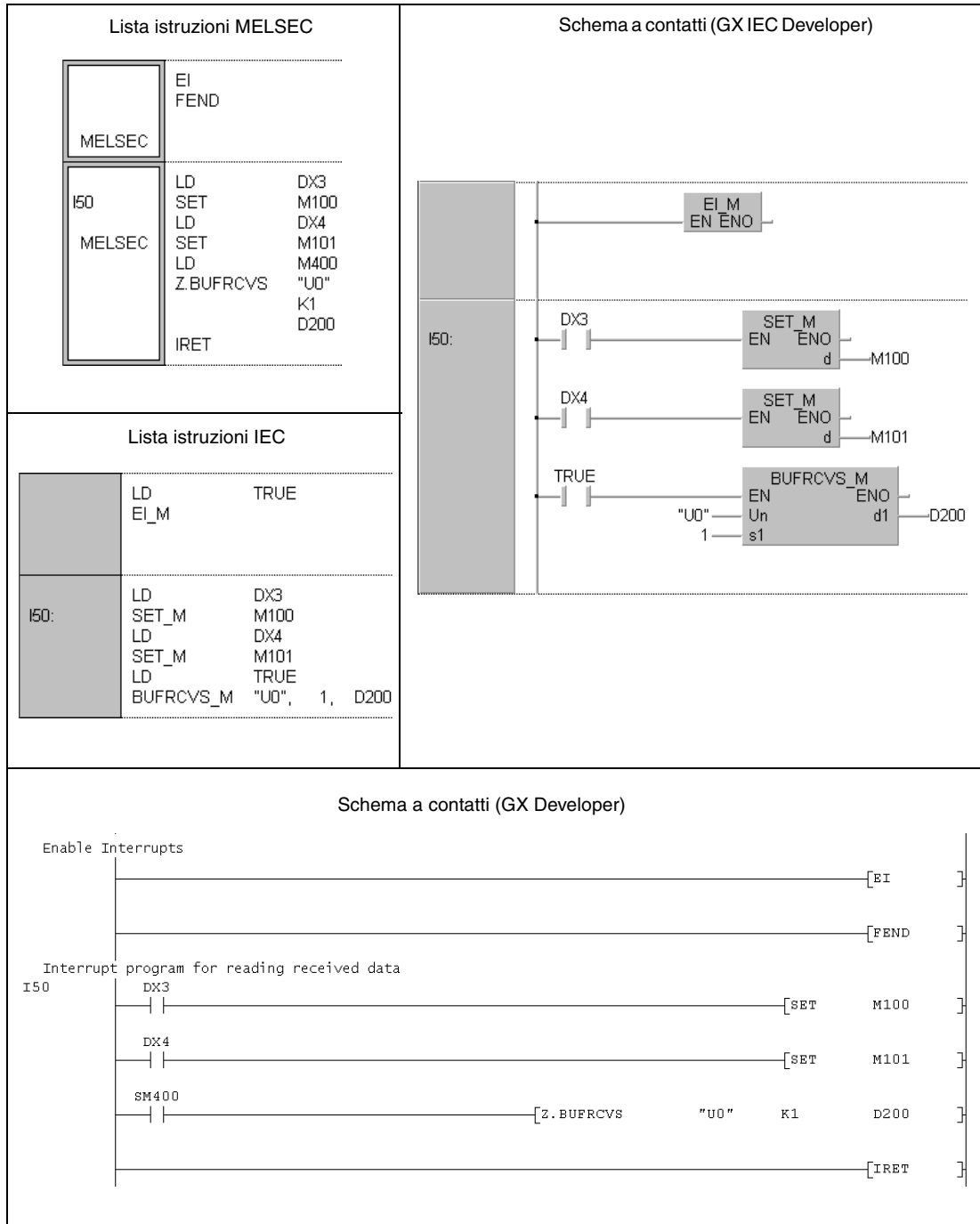
Se l'istruzione BUFRCVS viene completata con errori, viene attivato il flag di errore SM0, ed il codice di errore corrispondente viene memorizzato in SD0. Per ulteriori informazioni sui codici di errore, fare riferimento ai seguenti manuali:

- Con codici di errore pari a 4FFF_H o inferiori, fare riferimento al capitolo 13 di questo manuale per la diagnosi dell'errore.
- Con codici di errore pari a 7000_H o superiori, fare riferimento al manuale utente del modulo di comunicazione seriale QJ71C24.

Se si verifica un errore durante la ricezione dati (indicato dai segnali di ingresso X4 e XB), il codice di errore viene scritto nel buffer di memoria agli indirizzi 258_H e 268_H del modulo di comunicazione, e sono accessibili per la diagnostica..

Esempio di programma BUFRCVS

Il programma seguente legge i dati ricevuti dal canale 1 di un modulo QJ71C24 con indirizzo iniziale di I/O X/Y0 e memorizza i dati a partire da D200. Solo il canale 1 emette una interruzione. Alla ricezione dei dati, viene elaborato il programma di interruzione 50 (I50). I relé interni M100 e M101 sono usati come interfaccia con il programma principale. Se i dati sono ricevuti correttamente, M100 viene attivato. Se si verifica un errore durante la ricezione dati, viene attivato M101. Entrambi i relé vengono azzerati dal programma principale.



11.1.2 GETE, GETEP

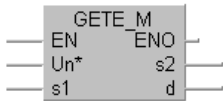
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

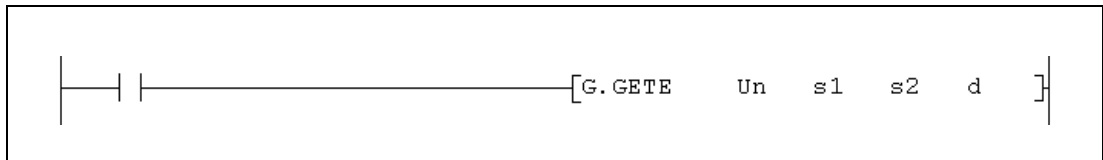
Operandi MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	—	SM0	
s2	—	●	●	—	—	—	—	—	—		
d	●	●	●	—	—	—	—	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p>MELSEC</p> </div> <p>G.GETE Un s1 s2 d</p>	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p>GETE_M Un, s1, s2, d</p> </div>
---	--	--

GX Developer



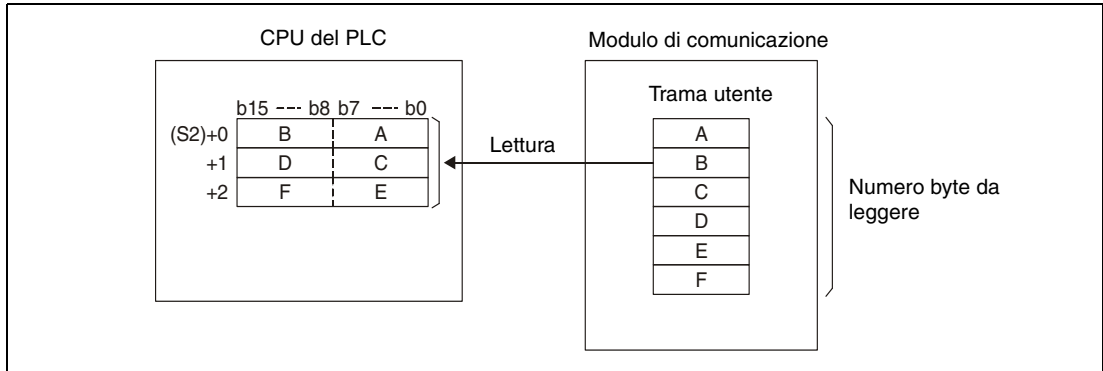
Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati		
Un	Indirizzo iniziale di I/O del modulo di comunicazione seriale (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come „U10“)		da 0 a FE _H	Utente	BIN 16-bit	
s1	Indirizzo iniziale degli operandi che memorizzano i dati di controllo				BIN 16-bit	
	Dati impostati	Significato	Descrizione	Campo		Contenuto inserito da
	(s1)+0	Dummy	Usata dal sistema	0		—
	(s1)+1	Risultato lettura	Indica se si è verificato un errore durante l'esecuzione dell'istruzione: 0000 _H : Nessun errore Qualsiasi valore diverso da 0000 _H : Si è verificato un errore e il valore contenuto è un codice di errore.			Sistema
	(s1)+2	Numero trama	Numero della trama utente	da 1000 a 1199		Utente
(s1)+3	Numero byte da leggere	Numero massimo dei byte della trama utente che possono essere contenuti nell'area specificata da s2	da 1 a 80			
		Numero byte letti	Numero dei byte letti della trama utente	da 1 a 80	Sistema	
s2	Indirizzo iniziale degli operandi che memorizzano i dati letti			Utente Commutazione di	Indirizzo	
d	Operando a bit attivato per una scansione al termine dell'istruzione GETE. (d)+1 segnala una eventuale terminazione anomala dell'istruzione.				Bit	
	Dati impostati	Significato	Descrizione	Campo		Contenuto inserito da
	(d)+0:	Istruzione completata	Indica lo stato di completamento dell'istruzione GETE ON: Istruzione completata OFF: Istruzione non completata	—		Sistema
(d)+1:	Istruzione completata con errore	Indica lo stato di completamento anomalo dell'istruzione GETE. ON: Completamento anomalo OFF: Completamento normale	—			

Funzioni Lettura trame utente registrate

GETE Lettura dati

L'istruzione GETE legge dati da una trama utente in un modulo di comunicazione seriale e memorizza i dati nella CPU del PLC. L'indirizzo iniziale del modulo di comunicazione è specificato con Un.

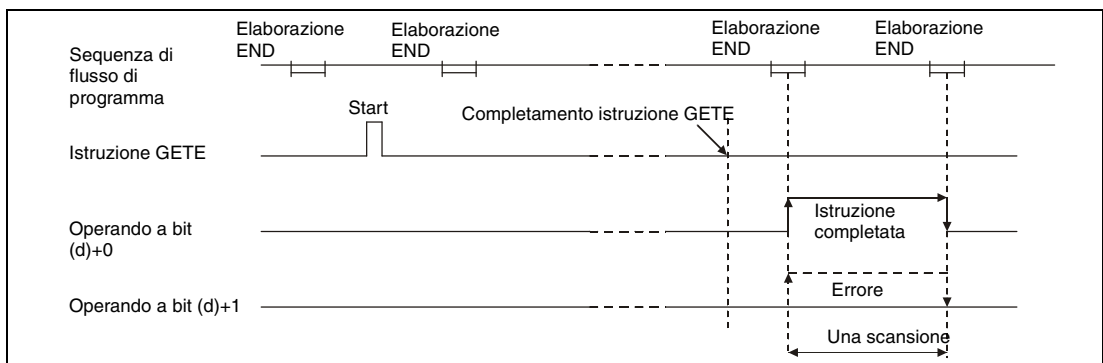


Durante l'esecuzione dell'istruzione GETE, non possono essere eseguite altre istruzioni GETE o PUTE. Se viene richiamata l'esecuzione di una istruzione GETE o PUTE mentre è in corso l'esecuzione di una istruzione GETE, il sistema attende fino al completamento dell'istruzione già in esecuzione.

Lo stato di esecuzione dell'istruzione GETE può essere controllato con gli operandi (d)+0 e (d)+1:

- L'operando a bit (d)+0 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione GETE, e si disattiva con l'elaborazione di END della scansione successiva.
- L'operando a bit (d)+1 segnala la presenza di un errore durante l'esecuzione di una istruzione GETE. Se l'istruzione termina normalmente, l'operando rimane OFF. Se si verifica un errore durante l'esecuzione dell'istruzione GETE, (d)+1 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione GETE, e si disattiva con l'elaborazione di END della scansione successiva.

La figura seguente mostra la temporizzazione durante l'esecuzione di una istruzione GETE:



Errori di esecuzione

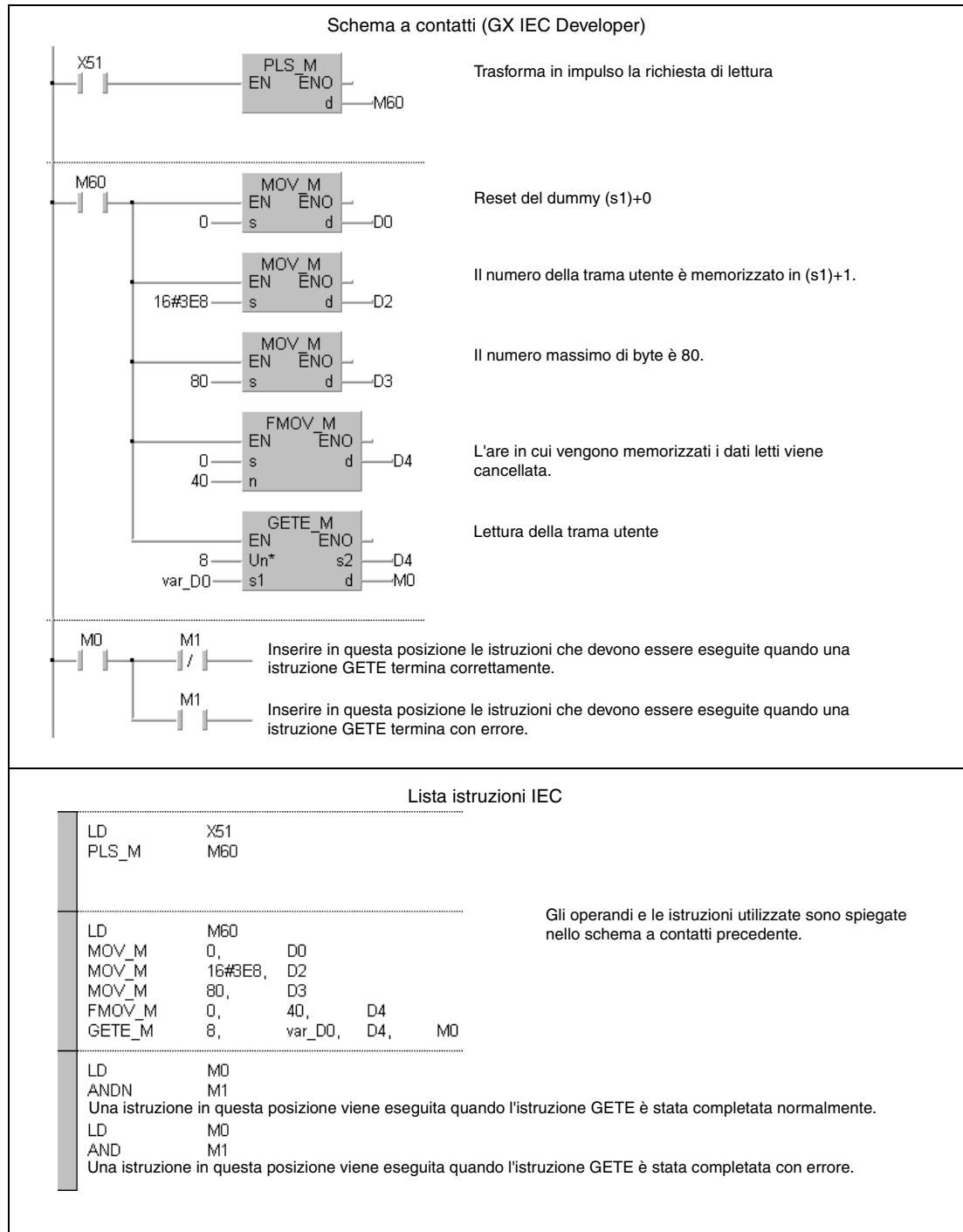
Se si verifica un errore durante l'esecuzione dell'istruzione GETE, l'operando a bit (d)+1 si attiva e in (s1)+1 viene scritto un codice di errore. Per ulteriori informazioni sui codici di errore, fare riferimento ai manuali seguenti:

- Con codici di errore pari a 4FFF_H o inferiori, fare riferimento al capitolo 13 di questo manuale per la diagnosi dell'errore.
- Con codici di errore pari a 7000_H o superiori, fare riferimento al manuale utente del modulo di comunicazione seriale.

Esempio di programma GETE

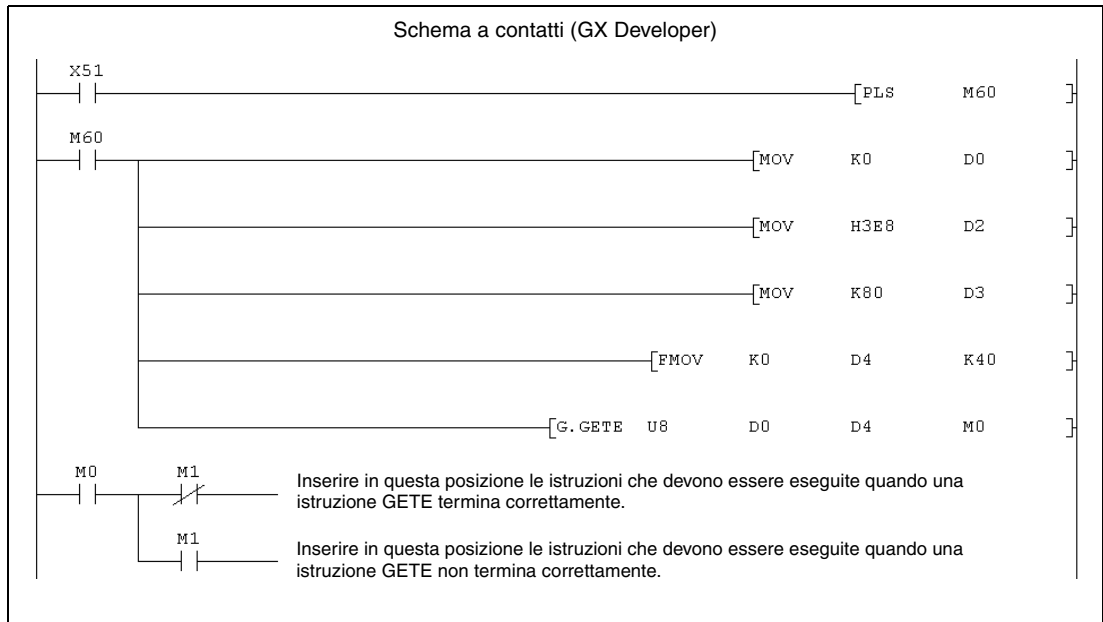
Il programma seguente legge i dati della trama utente di numero 3E8_H da un modulo QJ71C24 e memorizza i dati nella QCPU a partire dal registro dati D4. Il modulo di comunicazione occupa i segnali di ingresso/uscita da X/Y80 a X/Y9F.

● Editor IEC

**NOTA**

Negli editor IEC è necessario definire le variabili nello header della unità organizzativa di programma (POU) La mancata definizione delle variabili provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

- Lista istruzioni MELSEC e schema a contatti di GX Developer
 Per una spiegazione degli operandi e istruzioni utilizzate, vedere il programma di esempio per lo schema a contatti di GX IEC Developer nella pagina precedente.



Lista istruzioni MELSEC

MELSEC	LD X51 PLS M60
MELSEC	LD M60 MOV K0 D0 MOV H3E8 D2 MOV K80 D3 FMOV K0 D4 K40 G.GETE U8 D0 D4 M0
MELSEC	LD M0 MPS ANI M1 Una istruzione in questa posizione viene eseguita quando l'istruzione GETE è stata completata normalmente. MPP AND M1 Una istruzione in questa posizione viene eseguita quando l'istruzione GETE è stata completata con errore.

11.1.3 PUTE, PUTEP

CPU

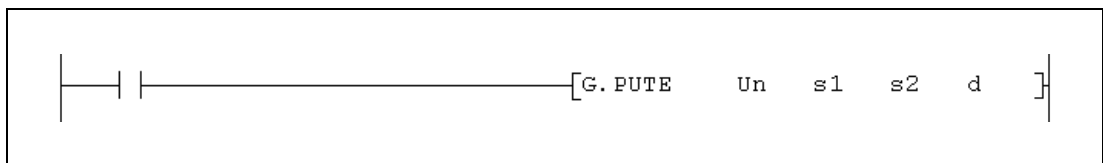
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

Operandi
MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	—	SM0	
s2	—	●	●	—	—	—	—	—	—		
d	●	●	●	—	—	—	—	—	—		

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC										
<div style="border: 1px solid black; padding: 5px; width: fit-content;">MELSEC</div> <table style="margin-left: 20px;"> <tr> <td>G.PUTE</td> <td>Un</td> </tr> <tr> <td></td> <td>s1</td> </tr> <tr> <td></td> <td>s2</td> </tr> <tr> <td></td> <td>d</td> </tr> </table>	G.PUTE	Un		s1		s2		d		<table style="margin-left: 20px;"> <tr> <td>PUTE_M</td> <td>Un, s1, s2, d</td> </tr> </table>	PUTE_M	Un, s1, s2, d
G.PUTE	Un											
	s1											
	s2											
	d											
PUTE_M	Un, s1, s2, d											

GX
Developer

Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati		
Un	Indirizzo iniziale di I/O del modulo di comunicazione seriale (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come „U10“)		da 0 a FE _H	Utente	BIN 16-bit	
s1	Indirizzo iniziale degli operandi che memorizzano i dati di controllo				BIN 16-bit	
	Dati impostati	Significato	Descrizione	Campo		Contenuto inserito da
	(s1)+0	Selezione: registrazione o cancellazione trama utente	Indica se cancellare o registrare la trama utente specificata da (s1)+2: <ul style="list-style-type: none"> • 1: Registrazione • 3: Cancellazione 	1 o 3		Utente
	(s1)+1	Risultato registrazione/cancellazione	Indica se si è verificato un errore durante l'esecuzione dell'istruzione: 0000 _H : Nessun errore Qualsiasi valore diverso da 0000 _H : Si è verificato un errore e il valore contenuto è un codice di errore.	—		Sistema
	(s1)+2	Numero trama	Numero della trama utente da registrare o cancellare	da 1000 a 1199		Utente
(s1)+3	Numero byte da registrare	Numero di byte della trama utente da registrare. Inserire un valore dummy compreso fra 1 e 80 per la cancellazione di una trama utente [(s1)+0 = 3].	da 1 a 80			
s2	Indirizzo del primo operando che memorizza i dati da registrare.			Utente	Indirizzo	
d	Operando a bit attivato per una scansione al termine dell'istruzione PUTE. (d)+1 segnala una eventuale terminazione anomala dell'istruzione.				Bit	
	Operando	Significato	Descrizione	Campo		Contenuto inserito da
	(d)+0:	Istruzione completata	Indica lo stato di completamento dell'istruzione PUTE ON: Istruzione completata OFF: Istruzione non completata	—		Sistema
(d)+1:	Istruzione completata con errore	Indica lo stato di completamento anomalo dell'istruzione PUTE ON: Completamento anomalo OFF: Completamento normale	—			

Funzioni **Registrazione o cancellazione di trame utente****PUTE** **Registrazione o cancellazione trame utente**

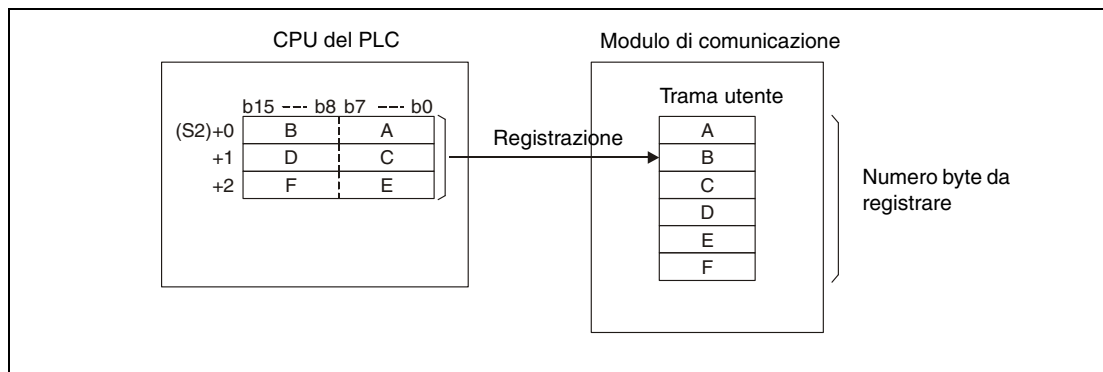
L'istruzione PUTE è utilizzata per registrare o cancellare trame utente in un modulo comunicazione seriale. L'indirizzo iniziale del modulo di comunicazione seriale è specificato con Un.

Registrazione di una trama utente

Per registrare una trama utente, scrivere „1“ nell'operando specificato in (s1)+0. I dati a partire dall'operando indicato da s2 vengono registrati in accordo con i dati di controllo.

Dato che ciascun operando può contenere due byte, il numero di operandi necessari è la metà dei byte dati.

Ad esempio, se si devono registrare sei byte in una trama utente, dopo s2 devono essere riservati altri due operandi:

**Cancellazione di una trama utente**

Per cancellare la trama utente il cui numero è inserito in (s1)+2, scrivere „3“ nell'operando specificato da (s1)+0.

Sebbene il numero di byte [(s1)+3] e l'area specificata da s2 non vengano usati per la cancellazione, questi dati sono resi necessari dal formato dell'istruzione PUTE. Scrivere un qualsiasi valore compreso fra 1 e 80 nell'operando indicato da (s1)+3 e scegliere un indirizzo dummy per s2.

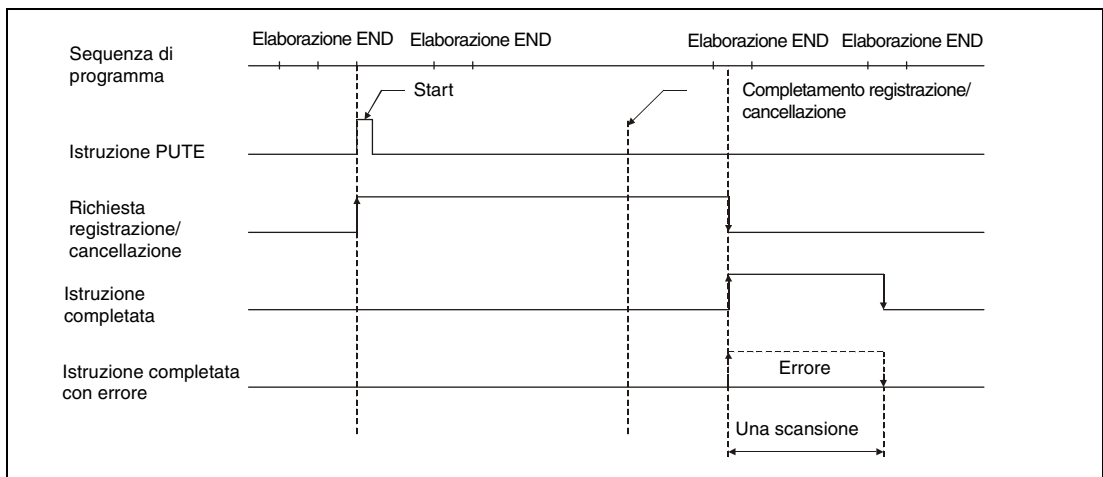
Condizioni operative

Durante l'esecuzione di una istruzione PUTE, non è possibile eseguire un'altra istruzione PUTE o GETE. Se si cerca di eseguire una di queste istruzioni mentre una istruzione PUTE è già in esecuzione, il sistema attende fino al completamento dell'esecuzione dell'istruzione già attiva.

Lo stato di esecuzione dell'istruzione PUTE può essere controllato con gli operandi (d)+0 e (d)+1:

- L'operando a bit (d)+0 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione PUTE, e si disattiva con l'elaborazione di END della scansione successiva.
- L'operando a bit (d)+1 segnala la presenza di un errore durante l'esecuzione di una istruzione PUTE. Se l'istruzione termina normalmente, l'operando rimane OFF. Se si verifica un errore durante l'esecuzione dell'istruzione PUTE, (d)+1 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione PUTE, e si disattiva con l'elaborazione di END della scansione successiva.

La figura seguente mostra la temporizzazione dell'istruzione PUTE:



Errori di esecuzione

Se si verifica un errore durante l'esecuzione dell'istruzione PUTE, l'operando a bit (d)+1 si attiva e in (s1)+1 viene scritto un codice di errore. Per ulteriori informazioni sui codici di errore, fare riferimento ai manuali seguenti:

- Con codici di errore pari a 4FFF_H o inferiori, fare riferimento al capitolo 13 di questo manuale per la diagnosi dell'errore.
- Con codici di errore pari a 7000_H o superiori, fare riferimento al manuale utente del modulo di comunicazione seriale.

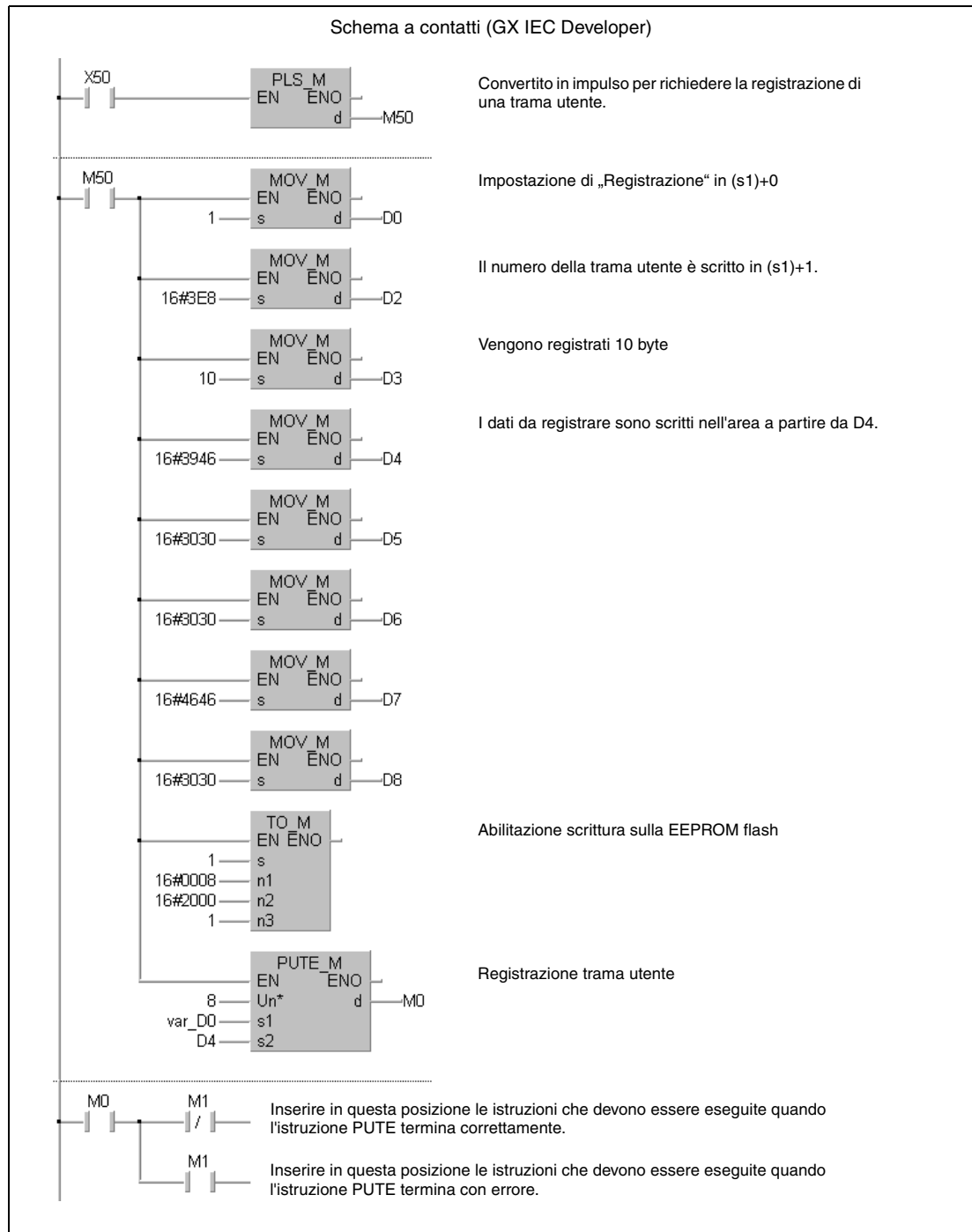
Esempio di programma PUTE

Il programma seguente registra i dati nella trama utente con il numero 3E8_H. Come modulo di comunicazione viene usato un modulo QJ71C24. Il modulo di comunicazione occupa i segnali di ingresso/uscita da X/Y80 a X/Y9F.

NOTA

Negli editor IEC è necessario definire le variabili nello header della unità organizzativa di programma (POU) La mancata definizione delle variabili provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

● Editor IEC

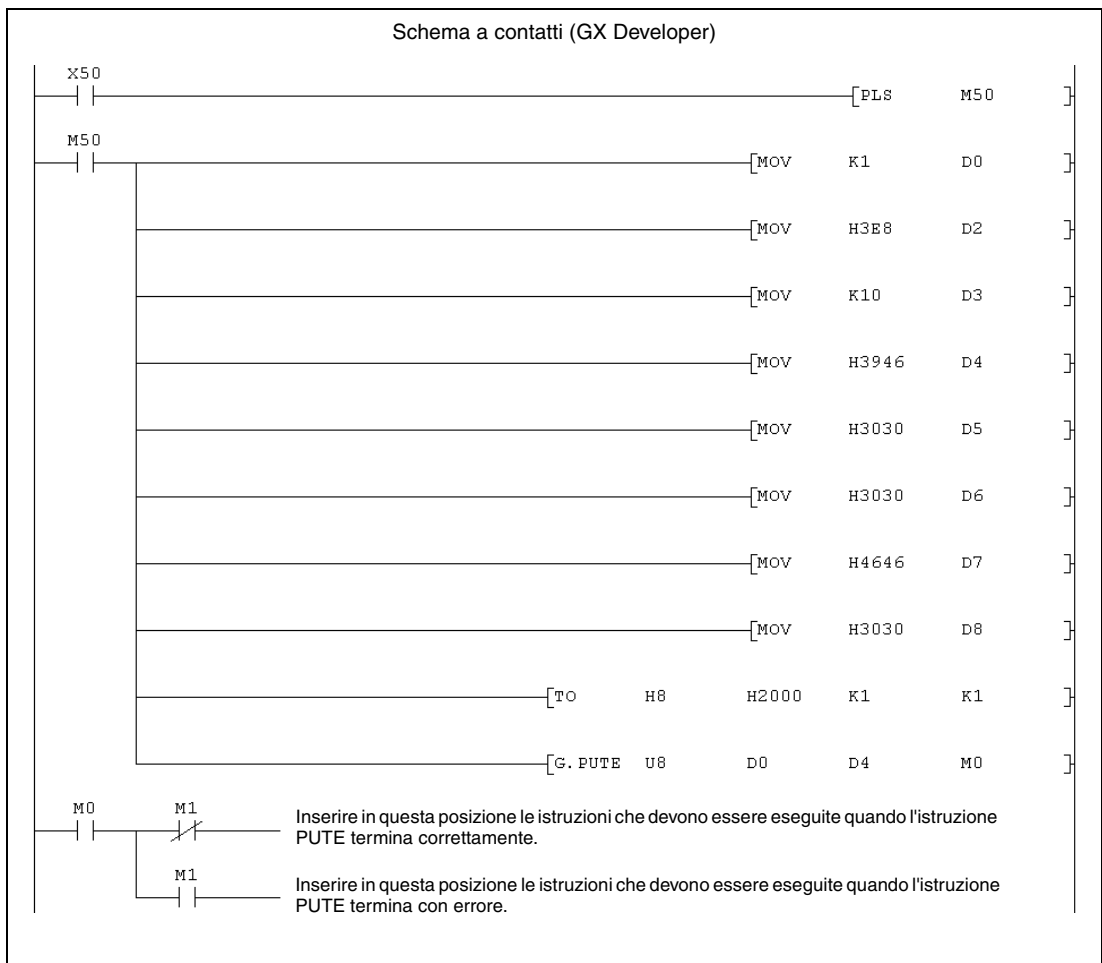


Lista istruzioni IEC

	LD	X50		
	PLS_M	M50		
	LD	M50		
	MOV_M	1,	D0	
	MOV_M	16#3E8,	D2	
	MOV_M	10,	D3	
	MOV_M	16#3946,	D4	
	MOV_M	16#3030,	D5	
	MOV_M	16#3030,	D6	
	MOV_M	16#4646,	D7	
	MOV_M	16#3030,	D8	
	TO_M	1,	16#0008,	16#2000,
	PUTE_M	8,	var_D0,	D4,
				M0
	LD	M0		
	ANDN	M1		
	Una istruzione in questa posizione viene eseguita quando l'istruzione PUTE è stata completata normalmente.			
	LD	M0		
	AND	M1		
	Una istruzione in questa posizione viene eseguita quando l'istruzione PUTE è stata completata con errore.			

Per una spiegazione su operandi e istruzioni usati, consultare lo schema a contatti a pagina precedente.

- Lista istruzioni MELSEC e schema a contatti di GX Developer
 Per una spiegazione degli operandi e istruzioni utilizzate, vedere il programma di esempio per lo schema a contatti di GX IEC Developer a pagina precedente.



Lista istruzioni MELSEC				
MELSEC	LD	X50		
	PLS	M50		
MELSEC	LD	M50		
	MOV	K1	D0	
	MOV	H3E8	D2	
	MOV	K10	D3	
	MOV	H3946	D4	
	MOV	H3030	D5	
	MOV	H3030	D6	
	MOV	H4646	D7	
	MOV	H3030	D8	
	TO	H8	H2000	K1
G.PUTE	U8	D0	D4	M0
MELSEC	LD	M0		
	MPS			
	ANI	M1		
	Una istruzione in questa posizione viene eseguita quando l'istruzione PUTE è stata completata normalmente.			
	MPP			
MELSEC	AND	M1		
	Inserire in questa posizione le istruzioni che devono essere eseguite quando l'istruzione PUTE termina con errore.			

11.1.4 PRR, PRRP

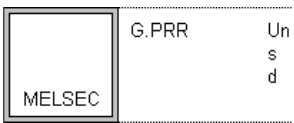
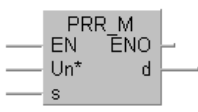
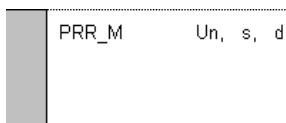
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

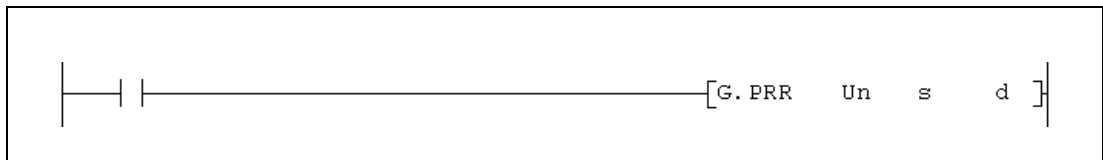
Operandi MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	—	SM0	
d1	●	●	●	—	—	—	—	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
--	--	---

GX Developer



Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati		
Un	Indirizzo iniziale di I/O del modulo di comunicazione seriale (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come „U10“)	da 0 a FE _H	Utente	BIN 16-bit		
s	Indirizzo iniziale degli operandi che memorizzano i dati di controllo.					
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da	
	(s)+0	Canale di trasmissione	Selezione del canale usato per trasmettere i dati 1: Canale 1 (CH1) 2: Canale 2 (CH2)	1 o 2	Utente	Indirizzo
	(s)+1	Risultato trasmissione	Indica se si è verificato un errore durante l'esecuzione dell'istruzione: 0000 _H : Nessun errore Qualsiasi valore diverso da 0000 _H : Si è verificato un errore e il valore contenuto è un codice di errore.	—	Sistema	
	(s)+2	Aggiunta di CR/LF	Indica se i caratteri CR/LF devono essere aggiunti ai dati trasmessi 0: Non aggiungere CR/LF 1: Aggiungi CR/LF	0 o 1	Utente	
	(s)+3	Puntatore di trasmissione	Puntatore al primo indirizzo dell'area operandi che contiene i dati da trasmettere.	da 1 a 100		
(s)+4	Numero di trame utente	Numero di byte della trama utente da trasmettere.	da 1 a 100			
d	Operando a bit attivato per una scansione al termine dell'istruzione PRR. (d)+1 segnala una eventuale terminazione anomala dell'istruzione.					
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da	
	(d)+0:	Istruzione completata	Indica lo stato di completamento dell'istruzione PRR ON: Istruzione completata OFF: Istruzione non completata	—	Sistema	Bit
(d)+1:	Istruzione completata con errore	Indica lo stato di completamento anomalo dell'istruzione PRR ON: Completamento anomalo OFF: Completamento normale	—			

Funzioni Trasmissione di trame utente

PRR Trasmissione trame utente

L'istruzione PRR trasmette dati usando trame utente con il modulo di comunicazione indicato da Un. Le informazioni relative all'elaborazione dell'istruzione sono contenute nell'operando indicato da s. I contenuti delle trame utente devono essere impostati nel modulo di comunicazione prima di eseguire l'istruzione PRR.

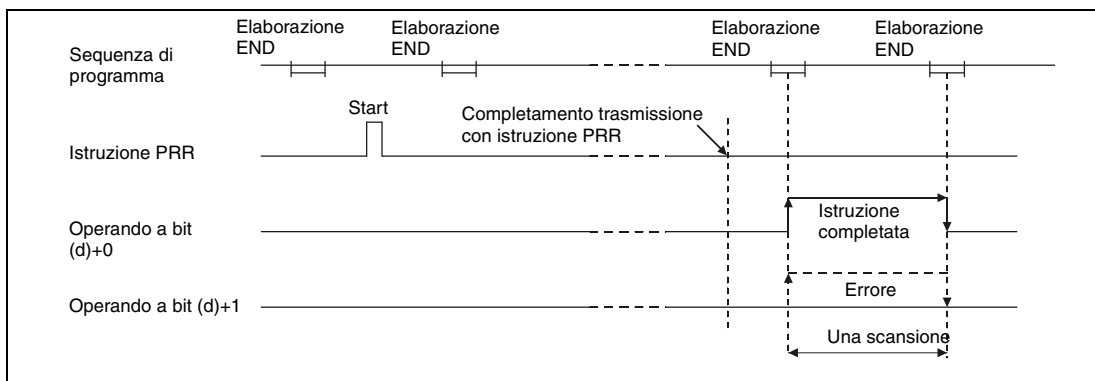
Durante l'esecuzione di una istruzione PRR, le istruzioni seguenti non possono essere eseguite sullo stesso canale del modulo di comunicazione: istruzione OUTPUT, istruzione ONDEMAND, istruzione BIDOUT ed altre istruzioni PRR.

Se si cerca di eseguire una delle istruzioni indicate durante l'elaborazione dell'istruzione PRR, il sistema attende fino al termine dell'esecuzione dell'istruzione PRR.

Lo stato di esecuzione dell'istruzione PRR può essere controllato con gli operandi (d)+0 e (d)+1:

- L'operando a bit (d)+0 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione PRR, e si disattiva con l'elaborazione di END della scansione successiva.
- L'operando a bit (d)+1 segnala la presenza di un errore durante l'esecuzione di una istruzione PRR. Se l'istruzione termina normalmente, l'operando rimane OFF. Se si verifica un errore durante l'esecuzione dell'istruzione GETE, (d)+1 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione PRR, e si disattiva con l'elaborazione di END della scansione successiva.

La figura seguente mostra la temporizzazione dell'istruzione PRR:



Errori di esecuzione

Se si verifica un errore durante l'esecuzione dell'istruzione PUTE, l'operando a bit (d)+1 si attiva e in (s1)+1 viene scritto un codice di errore. Per ulteriori informazioni sui codici di errore, fare riferimento ai manuali seguenti:

- Con codici di errore pari a 4FFF_H o inferiori, fare riferimento al capitolo 13 di questo manuale per la diagnosi dell'errore.
- Con codici di errore pari a 7000_H o superiori, fare riferimento al manuale utente del modulo di comunicazione seriale.

Esempio di programma PRR

Questo programma di esempio trasmette dati e le prime cinque trame utente. Viene usato il modulo di comunicazione QJ71C24. Il modulo di comunicazione occupa i segnali di ingresso/uscita da X/Y80 a X/Y9F. Il programma usa i seguenti registri dati:

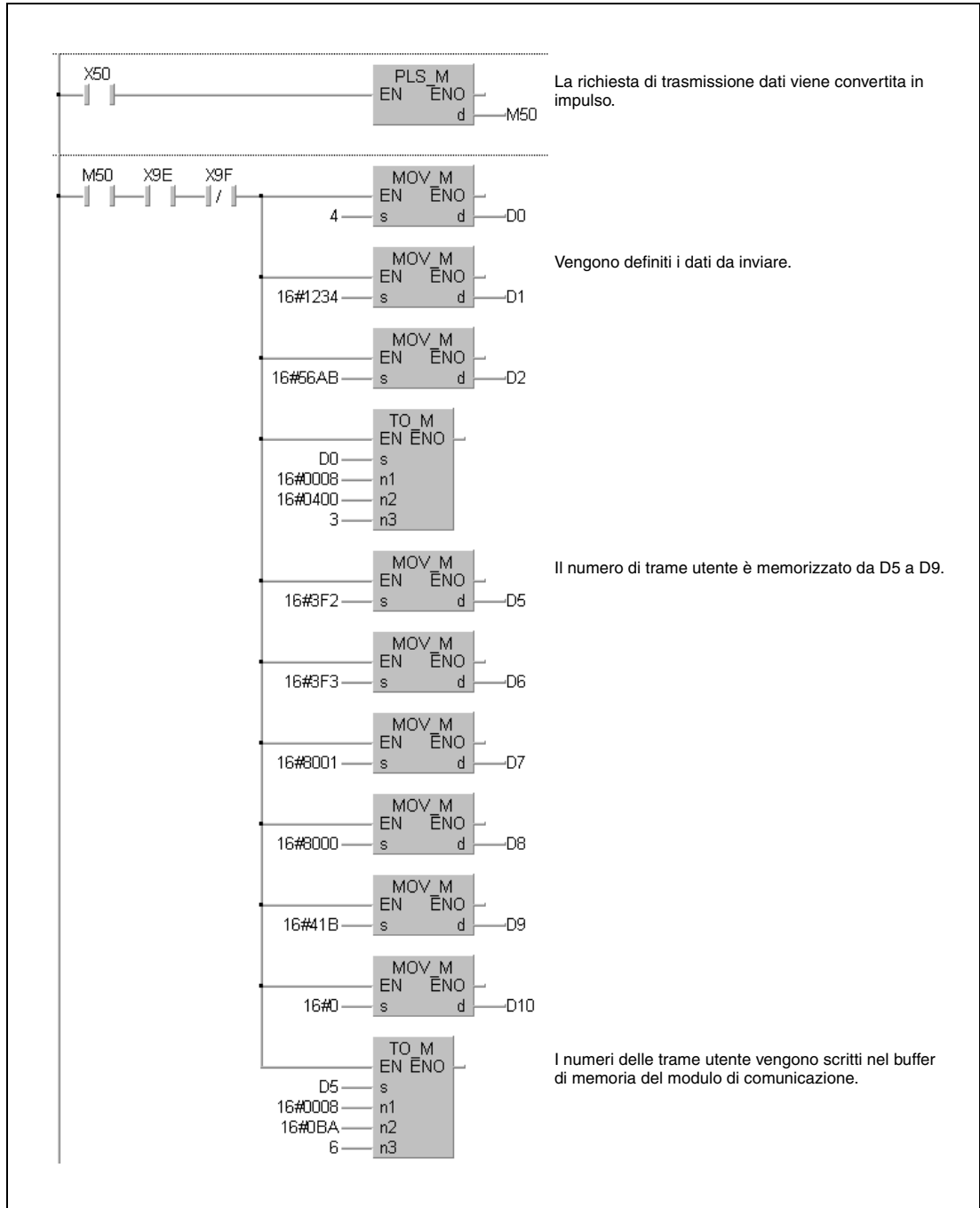
Registro dati	Contenuto	Significato	
D0	0004 _H	Numero byte da inviare	
D1	3412 _H	Dati da inviare	
D2	AB56 _H		
D5	03F2 _H	Numero di trame utente	
D6	03F3 _H		
D7	8001 _H		
D8	8000 _H		
D9	041B _H		
D10	0000 _H		
D11	0001 _H	(s)+0	Interfaccia: CH1
D12	0000 _H o codice di errore	(s)+1	Risultato trasmissione
D13	0000 _H	(s)+2	CR/LF non aggiunto
D14	0001 _H	(s)+3	Puntatore di trasmissione
D15	0005 _H	(s)+4	Numero trame dati da trasmettere

NOTA

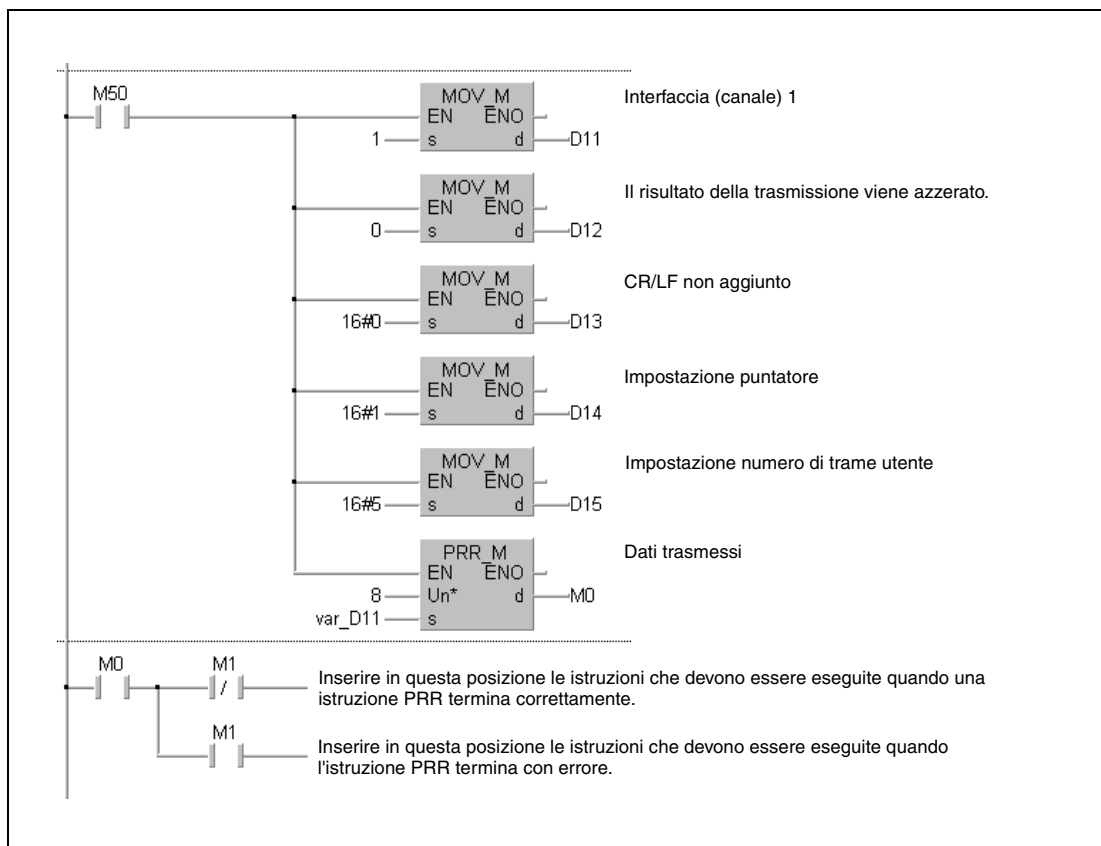
Negli editor IEC è necessario definire le variabili nello header della unità organizzativa di programma (POU) La mancata definizione delle variabili provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

● Editor IEC

Schema a blocchi di GX IEC Developer (parte 1)



Schema a blocchi di GX IEC Developer (continua)

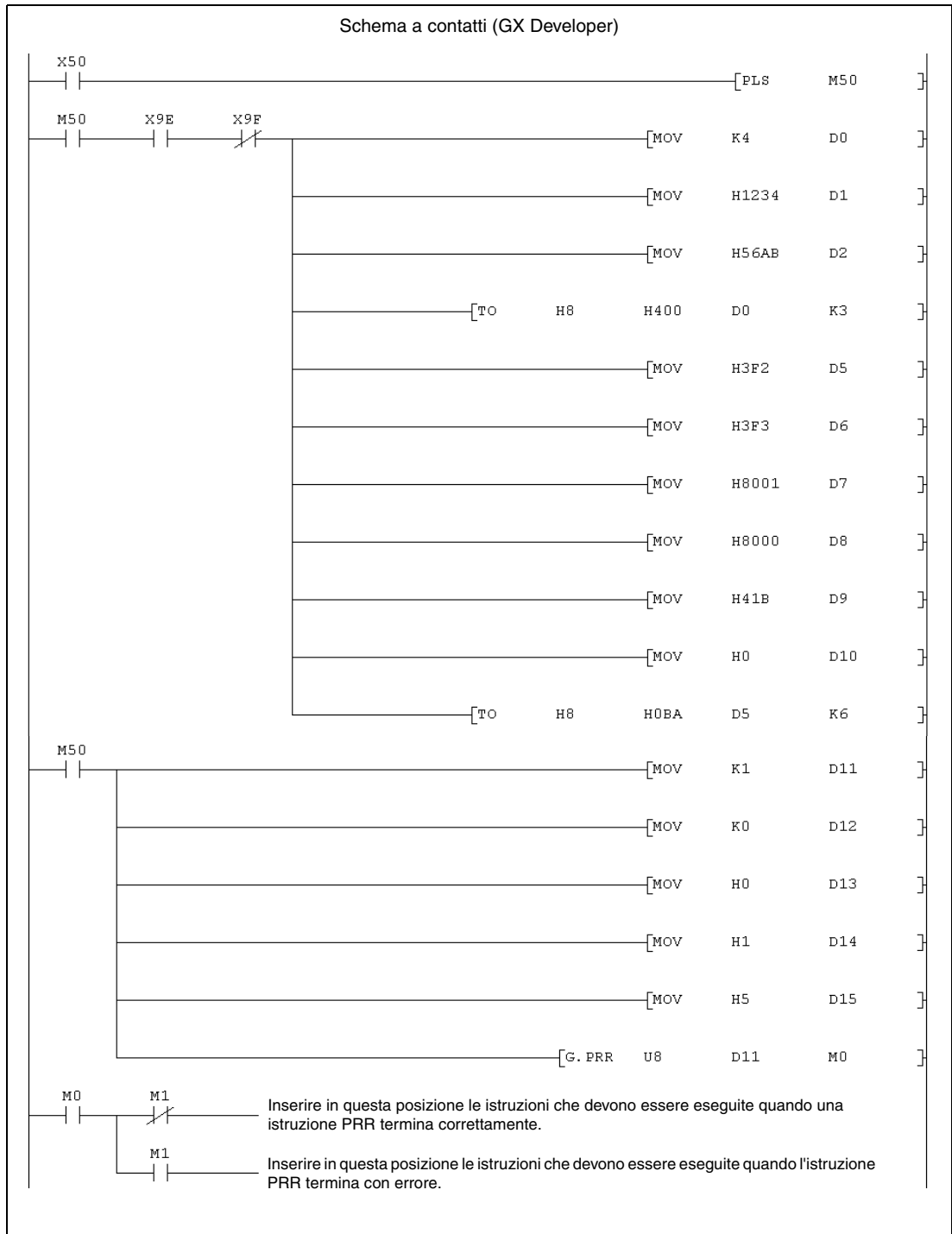


Lista istruzioni IEC

LD	X50		
PLS_M	M50		
LD	M50		
AND	X9E		
ANDN	X9F		
MOV_M	4,	D0	
MOV_M	16#1234,	D1	
MOV_M	16#56AB,	D2	
TO_M	D0,	16#0008,	16#0400, 3
MOV_M	16#3F2,	D5	
MOV_M	16#3F3,	D6	
MOV_M	16#0001,	D7	
MOV_M	16#0000,	D8	
MOV_M	16#41B,	D9	
MOV_M	16#0,	D10	
TO_M	D5,	16#0008,	16#0BA, 6
LD	M50		
MOV_M	1,	D11	
MOV_M	0,	D12	
MOV_M	16#0,	D13	
MOV_M	16#1,	D14	
MOV_M	16#5,	D15	
PRR_M	8,	var_D11,	M0
LD	M0		
ANDN	M1		
Una istruzione in questa posizione viene eseguita quando l'istruzione PRR è stata completata normalmente.			
LD	M0		
AND	M1		
Una istruzione in questa posizione viene eseguita quando l'istruzione PRR è stata completata con errore.			

Per una spiegazione su operandi e istruzioni usate, fare riferimento allo schema precedente.

- Lista istruzioni MELSEC e schema a contatti di GX Developer
 Per una spiegazione degli operandi e istruzioni utilizzate, vedere il programma di esempio per lo schema a contatti di GX IEC Developer.



Lista istruzioni MELSEC

MELSEC	LD	X50			
	PLS	M50			
MELSEC	LD	M50			
	AND	X9E			
	ANI	X9F			
	MOV	K4	D0		
	MOV	H1234	D1		
	MOV	H56AB	D2		
	TO	H8	H400	D0	K3
	MOV	H3F2	D5		
	MOV	H3F3	D6		
	MOV	H8001	D7		
	MOV	H8000	D8		
	MOV	H41B	D9		
	MOV	H0	D10		
TO	H8	H0BA	D5	K6	
MELSEC	LD	M50			
	MOV	K1	D11		
	MOV	K0	D12		
	MOV	H0	D13		
	MOV	H1	D14		
	MOV	H5	D15		
G.PRR	U8	D11	M0		
MELSEC	LD	M0			
	MPS				
	ANI	M1			
	MPP				
	AND	M1			

Una istruzione in questa posizione viene eseguita quando l'istruzione PRR è stata completata normalmente.

Inserire in questa posizione le istruzioni che devono essere eseguite quando l'istruzione PRR termina con errore.

11.2 Istruzioni per modulo interfaccia PROFIBUS/DP

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Lettura dati dal buffer di memoria di un modulo interfaccia PROFIBUS/DP.	G.BBLKRD	BBLKRD_M
	GP.BBLKRD	BBLKRDP_M
Scrittura dati nel buffer di memoria di un modulo interfaccia PROFIBUS/DP.	G.BBLKWR	BBLKWR_M
	GP.BBLKWR	BBLKWRP_M

11.2.1 BBLKRD, BBLKRD P


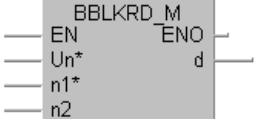

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					●

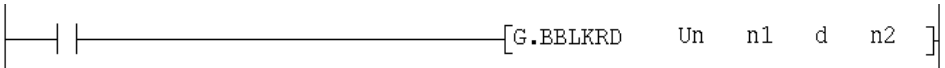
Operandi
MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
n1	—	●	●	—	—	—	—	●	—	SM0	
d	—	●	●	—	—	—	—	—	—		
n2	—	●	●	—	—	—	—	●	—		

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
<p>Lista istruzioni MELSEC</p>  <p>G.BBLKRD Un n1 d n2</p>	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p>  <p>BBLKRD_M Un, n1, n2, d</p>

GX
Developer


--

Variabili

Dati impostati	Significato	Tipo dati
Un	Indirizzo iniziale di I/O del modulo di interfaccia PROFIBUS sul telaio base	BIN 16-bit
n1	Indirizzo iniziale del buffer di memoria del modulo interfaccia PROFIBUS da cui si voglio leggere i dati.	
d	Indirizzo iniziale dell'area operandi nella CPU del PLC in cui vengono memorizzati i dati letti	Nome operando
n2	Numero dati da leggere	BIN 16-bit

Funzioni **Letture dati dal buffer di memoria di un modulo interfaccia PROFIBUS****BBLKRD / BBLKRD** **Letture dati**

L'istruzione BBLKRD viene utilizzata per leggere dal buffer di memoria dei moduli di interfaccia PROFIBUS QJ71PB92D e QJ71PB93D. Durante la lettura viene garantita la congruenza dei dati.

Il modulo QJ71PB93 deve essere preparato per l'istruzione BBLKRD, attivando il segnale di uscita Y0A. Quando il modulo PROFIBUS attiva a sua volta il segnale d'ingresso X0A, è possibile eseguire l'istruzione BBLKRD. Il segnale di uscita Y0A deve essere disattivato al termine della lettura dal buffer di memoria.

Campi ammessi e designazione degli operandi:

- Un (indirizzo iniziale di I/O del modulo interfaccia PROFIBUS): da 0 a FF_H
(vengono usate solo le due cifre più pesanti dell'indirizzo a tre cifre. Ad es. l'indirizzo iniziale X/Y100 viene impostato come 10_H.)
- n1 (Indirizzo di partenza nel buffer di memoria): L'indirizzo specificato deve esistere.
- d (indirizzo iniziale dell'area di destinazione): L'operando specificato deve esistere.
- n2 (Numero dati da leggere)
Per un modulo QJ71PB92D: da 1 a 960 word (da 1 a 3C0_H)
Per un modulo QJ71PB93D: da 1 a 122 word (da 1 a 7A_H)

NOTE

In una scansione può essere eseguita una sola istruzione BBLKRD.

Le istruzioni BBLKRD e BBLKWR (capitolo 11.2.2) lavorano indipendentemente.

Il ritardo di trasmissione aumenta se si utilizza l'istruzione BBLKRD.

L'istruzione BBLKRD non viene eseguita se il modulo di uscita non è stato inserito nelle impostazioni dei moduli dati della parametrizzazione della stazione master.

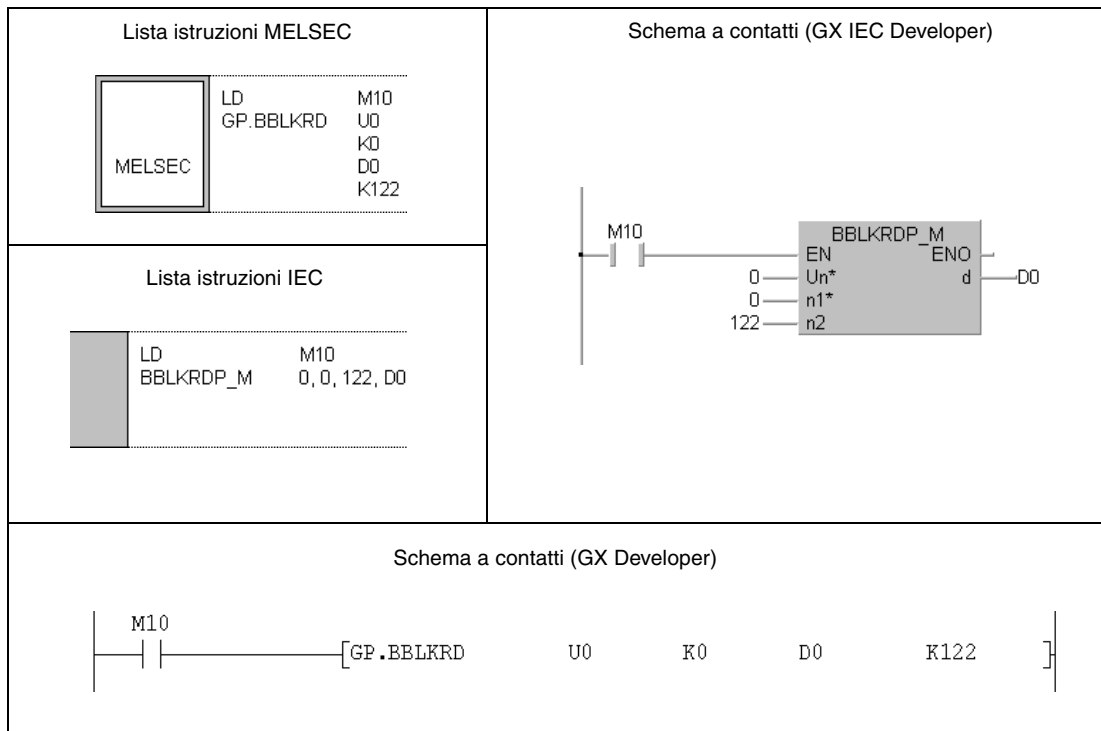
Errori di esecuzione

Nei casi seguenti, si verifica un errore di funzionamento, viene attivato il flag di errore SM0, ed un codice di errore viene memorizzato in SD0:

- Se viene impostato un valore oltre il campo ammesso fra i dati di impostazione (codice di errore: 4101).
- La dimensione del buffer di memoria viene superata sommando l'indirizzo iniziale del buffer specificato da n1 con il numero dei dati da leggere contenuto in n2 (codice di errore: 4101).
- Il numero di dati da leggere (indicato da n2) è maggiore dell'area operandi disponibile, partendo dall'indirizzo iniziale indicato da d (codice di errore: 4101).

Esempio di programma BBLKRDP

Se il relé M10 è attivo, vengono lette 122 word di dati dal buffer di memoria del modulo interfaccia PROFIBUS con indirizzo iniziale di I/O X/Y0. La lettura inizia dall'indirizzo 0 del buffer di memoria, mentre i dati vengono memorizzati a partire da D0.



11.2.2 BBLKWR, BBLKWRP


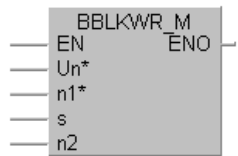
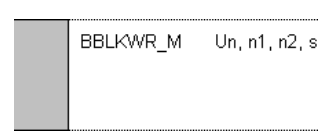
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					●

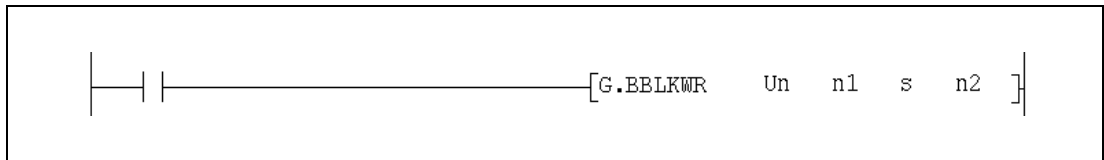
Operandi MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
n1	—	●	●	—	—	—	—	●	—	SM0	
s	—	●	●	—	—	—	—	—	—		
n2	—	●	●	—	—	—	—	●	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
---	--	--

GX Developer



Variabili

Dati impostati	Significato	Tipo dati
Un	Indirizzo iniziale di I/O del modulo di interfaccia PROFIBUS sul telaio base	BIN 16-bit
n1	Indirizzo iniziale del buffer di memoria del modulo interfaccia PROFIBUS in cui si voglio scrivere i dati.	
s	Indirizzo iniziale dell'are operandi della CPU del PLC in cui sono contenuti i dati da inviare al modulo interfaccia PROFIBUS.	Nome operando
n2	Numero dei dati da inviare al modulo interfaccia PROFIBUS	BIN 16-bit

Funzioni **Scrittura dati nel buffer di memoria di un modulo interfaccia PROFIBUS****BBLKWR / BBLKWRP** **Scrittura dati**

L'istruzione BBLKWR viene utilizzata per scrivere nel buffer di memoria dei moduli di interfaccia PROFIBUS QJ71PB92D e QJ71PB93D. Durante l'operazione di scrittura viene garantita la congruenza dei dati.

Il modulo QJ71PB93 deve essere preparato per l'istruzione BBLKWR, attivando il segnale di uscita Y0B. Quando il modulo PROFIBUS attiva a sua volta il segnale d'ingresso X0B, è possibile eseguire l'istruzione BBLKWR. Al completamento della scrittura nel buffer di memoria, il segnale di uscita Y0B deve essere azzerato.

Campi ammessi e designazione degli operandi:

- Un (indirizzo iniziale di I/O del modulo interfaccia PROFIBUS): da 0 a FF_H
(vengono usate solo le due cifre più pesanti dell'indirizzo a tre cifre. Ad es. l'indirizzo iniziale X/Y100 viene impostato come 10_H.)
- n1 (Indirizzo di partenza nel buffer di memoria): L'indirizzo specificato deve esistere.
L'indirizzo iniziale del modulo QJ71PB93 ha un offset di 100_H. Per la specifica di n1 è quindi necessario sottrarre 100_H dall'indirizzo iniziale desiderato. Ad esempio l'indirizzo iniziale 100_H viene specificato come „0_H“ e l'indirizzo iniziale 120_H viene specificato come „20_H“.
- d (indirizzo iniziale dell'area sorgente): L'operando specificato deve esistere.
- n2 (Numero dati da scrivere)
Per un modulo QJ71PB92D: da 1 a 960 word (da 1 a 3C0_H)
Per un modulo QJ71PB93D: da 1 a 122 word (da 1 a 7A_H)

NOTE

In una scansione può essere eseguita una sola istruzione BBLKWR.

Le istruzioni BBLKRD e BBLKWR (capitolo 11.2.2) lavorando indipendentemente.

Il ritardo di trasmissione aumenta se si utilizza l'istruzione BBLKWR.

L'istruzione BBLKWR non viene eseguita se il modulo di uscita non è stato inserito nelle impostazioni dei moduli dati della parametrizzazione della stazione master.

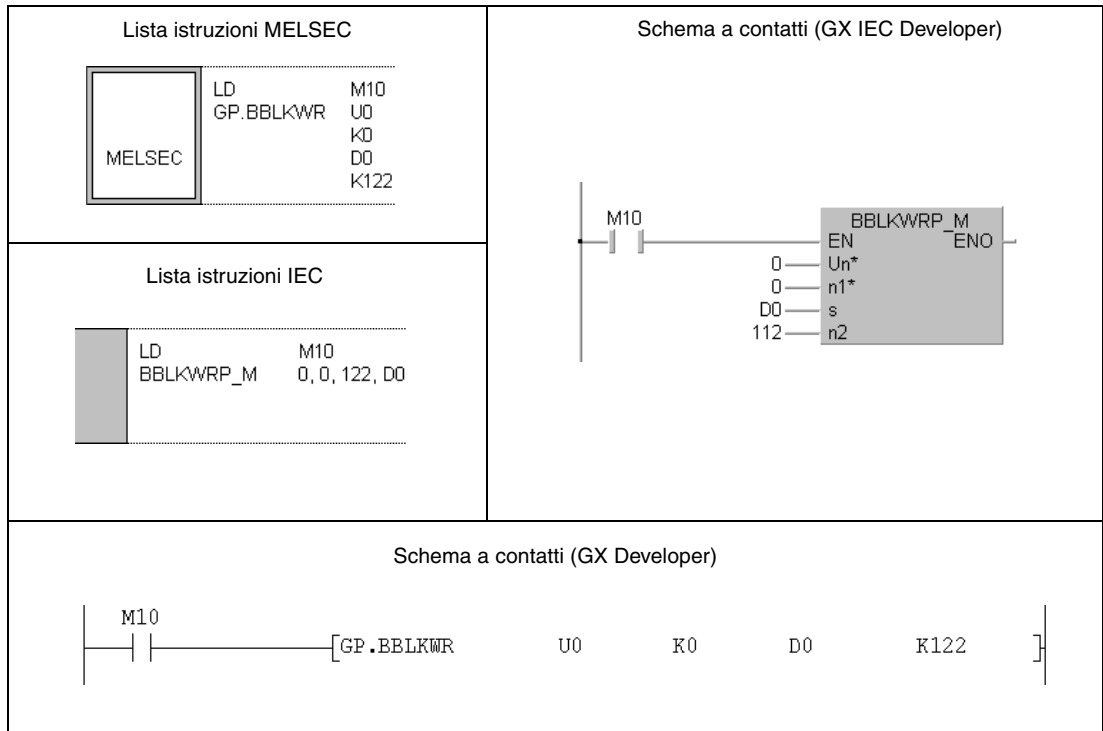
Errori di esecuzione

Nei casi seguenti, si verifica un errore di funzionamento, viene attivato il flag di errore SM0, ed un codice di errore viene memorizzato in SD0:

- Se viene impostato un valore oltre il campo ammesso fra i dati di impostazione. (codice di errore: 4101).
- La dimensione del buffer di memoria viene superata sommando l'indirizzo iniziale del buffer specificato da n1 con il numero dei dati da scrivere contenuto in n2 (codice di errore: 4101).
- Il numero di dati da scrivere (indicato da n2) è maggiore dell'area operandi disponibile, partendo dall'indirizzo iniziale indicato da d (codice di errore: 4101).

Esempio di programma BBLKWRP

Dopo l'attivazione del relé M10, il contenuto dei registri dati da D0 a D121 (122 word) viene scritto nell'area d'ingresso del modulo slave PROFIBUS/DP QJ71PB93D. L'area d'ingresso inizia dall'indirizzo 100_H del buffer di memoria. Notare che l'indirizzo iniziale specificato da n1 viene in questo caso impostato come „0_H“. L'indirizzo iniziale di I/O del modulo slave PROFIBUS/DP è X/Y0.



11.3 Istruzioni per moduli interfaccia ETHERNET

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Lettura di dati ricevuti da buffer fissi	ZP.BUFRVCV	BUFRVCV_M
	Z.BUFRCVS	BUFRCVS_M
Invio dati verso buffer fissi	ZP.BUFSND	BUFSND_M
Apertura di una connessione	ZP.OPEN	OPEN_M
Chiusura di una connessione	ZP.CLOSE	CLOSE_M
Cancellazione informazioni di errore	ZP.ERRCLR	ERRCLR_M
Lettura informazioni di errore	ZP.ERRRD	ERRRD_M
Reinizializzazione di un modulo interfaccia ETHERNET	ZP.UINI	UINI_M

11.3.1 BUFRCV

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					●

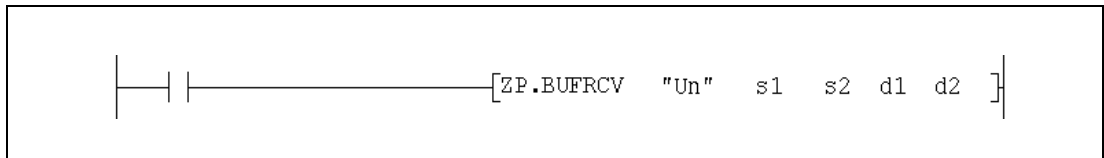
Operandi MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	●	—		
s2	—	●	●	—	—	—	—	—	—		
d1	—	●	●	—	—	—	—	—	—		
d2	●	●	●	—	—	—	—	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 10px auto;"> <p>MELSEC</p> </div> <table style="margin-left: auto; margin-right: auto;"> <tr> <td>ZP.BUFRCV</td> <td>"Un"</td> </tr> <tr> <td>s1</td> <td>s2</td> </tr> <tr> <td>d1</td> <td>d2</td> </tr> </table>	ZP.BUFRCV	"Un"	s1	s2	d1	d2	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p> <table style="margin-left: auto; margin-right: auto;"> <tr> <td style="background-color: #cccccc; width: 20px; height: 20px;"></td> <td>BUFRCV_M</td> <td>"Un", s1, s2, d1, d2</td> </tr> </table>		BUFRCV_M	"Un", s1, s2, d1, d2
ZP.BUFRCV	"Un"										
s1	s2										
d1	d2										
	BUFRCV_M	"Un", s1, s2, d1, d2									

GX Developer



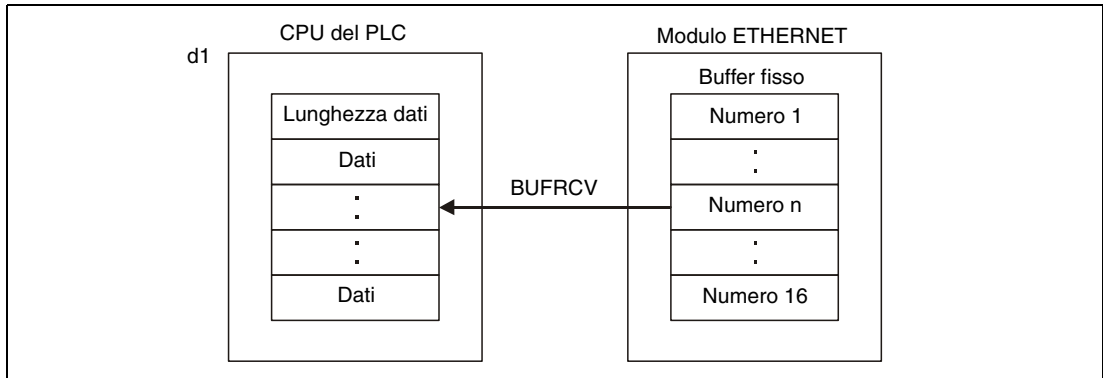
Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati	
„Un“	Indirizzo iniziale di I/O del modulo di interfaccia ETHERNET (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come „U10“)	da 0 a FE _H	Utente	BIN 16-bit	
s1	Numero connessione	da 1 a 16			
s2	Indirizzo iniziale dell'area operandi in cui sono contenuti i dati di controllo dell'istruzione.				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(s2)+0	Area di sistema	Usata dal sistema	—	Sistema
(s2)+1	Risultato dell'esecuzione dell'istruzione	Indica se si è verificato un errore durante l'esecuzione dell'istruzione: 0000 _H : Nessun errore Qualsiasi valore diverso da 0000 _H : Si è verificato un errore. Il valore memorizzato è un codice di errore, spiegato sia nel manuale utente del modulo interfaccia ETHERNET che nel capitolo 13 di questo manuale.			
d1	Indirizzo iniziale dell'area operandi in cui vengono memorizzati i dati ricevuti.				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(d1)+0	Lunghezza dei dati ricevuti	Con procedura (dati binari): Numero di word lette dal buffer fisso	da 1 a 1017 word	Sistema
			Con procedura (dati ASCII): Numero di word lette dal buffer fisso	da 1 a 508 word	
Senza procedura (dati binari): Numero di byte letti dal buffer fisso			da 1 a 2016 byte		
da (d1)+1 a (d1)+n	Dati ricevuti	I dati letti dal buffer fisso vengono memorizzati sequenzialmente in questa area in ordine crescente.	—		
d2	Operando a bit attivato per una scansione al termine dell'istruzione BUFRCV. (d)+1 segnala una eventuale terminazione anomala dell'istruzione.				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(d2)+0	Istruzione completata	Indica lo stato di completamento dell'istruzione BUFRCV ON: Istruzione completata OFF: Istruzione non completata	—	Sistema
(d2)+1	Istruzione completata con errore	Indica lo stato di completamento anomalo dell'istruzione BUFRCV ON: Completamento anomalo OFF: Completamento normale	—		

Funzioni **Letture dati ricevuti da buffer fisso (esecuzione dell'istruzione nel programma principale)**

BUFRCV Lettura dati

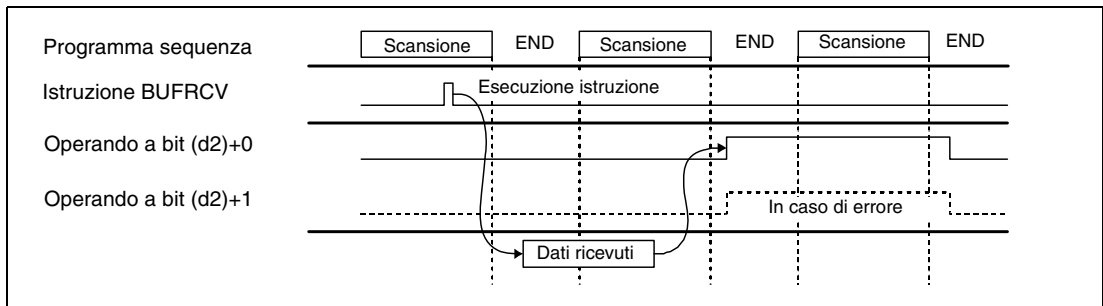
Con l'istruzione BUFRCV, i dati inviati da una stazione esterna verso un modulo interfaccia ETHERNET tramite comunicazione con buffer fisso, possono essere letti dal modulo ETHERNET e memorizzati nella CPU del PLC. L'istruzione BUFRCV viene eseguita nel programma principale, mentre l'istruzione BUFRCVS viene usata nei programmi di interruzione. L'area in cui devono essere memorizzati i dati è specificata da d1:



Lo stato di esecuzione dell'istruzione BUFRCV può essere controllato con gli operandi (d2)+0 e (d2)+1:

- L'operando a bit (d2)+0 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione BUFRCV, e si disattiva con l'elaborazione di END della scansione successiva.
- L'operando a bit (d2)+1 segnala la presenza di un errore durante l'esecuzione di una istruzione BUFRCV. Se l'istruzione termina normalmente, l'operando rimane OFF. Se si verifica un errore durante l'esecuzione dell'istruzione BUFRCV, (d2)+1 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione BUFRCV, e si disattiva con l'elaborazione di END della scansione successiva.

La temporizzazione dell'istruzione BUFRCV viene mostrata nella figura seguente:



L'istruzione BUFRCV può essere eseguita quando il modulo interfaccia ETHERNET indica la presenza di dati ricevuti. Per ciascuna delle 16 possibili connessioni viene riservato un bit all'indirizzo 5005_H del buffer di memoria, attivato quando vengono ricevuti dei dati.

NOTA *Non è possibile leggere dati ricevuti dalla stessa connessione con l'istruzione BUFRCV nel programma principale e con l'istruzione BUFRCVS in un programma di interruzione.*

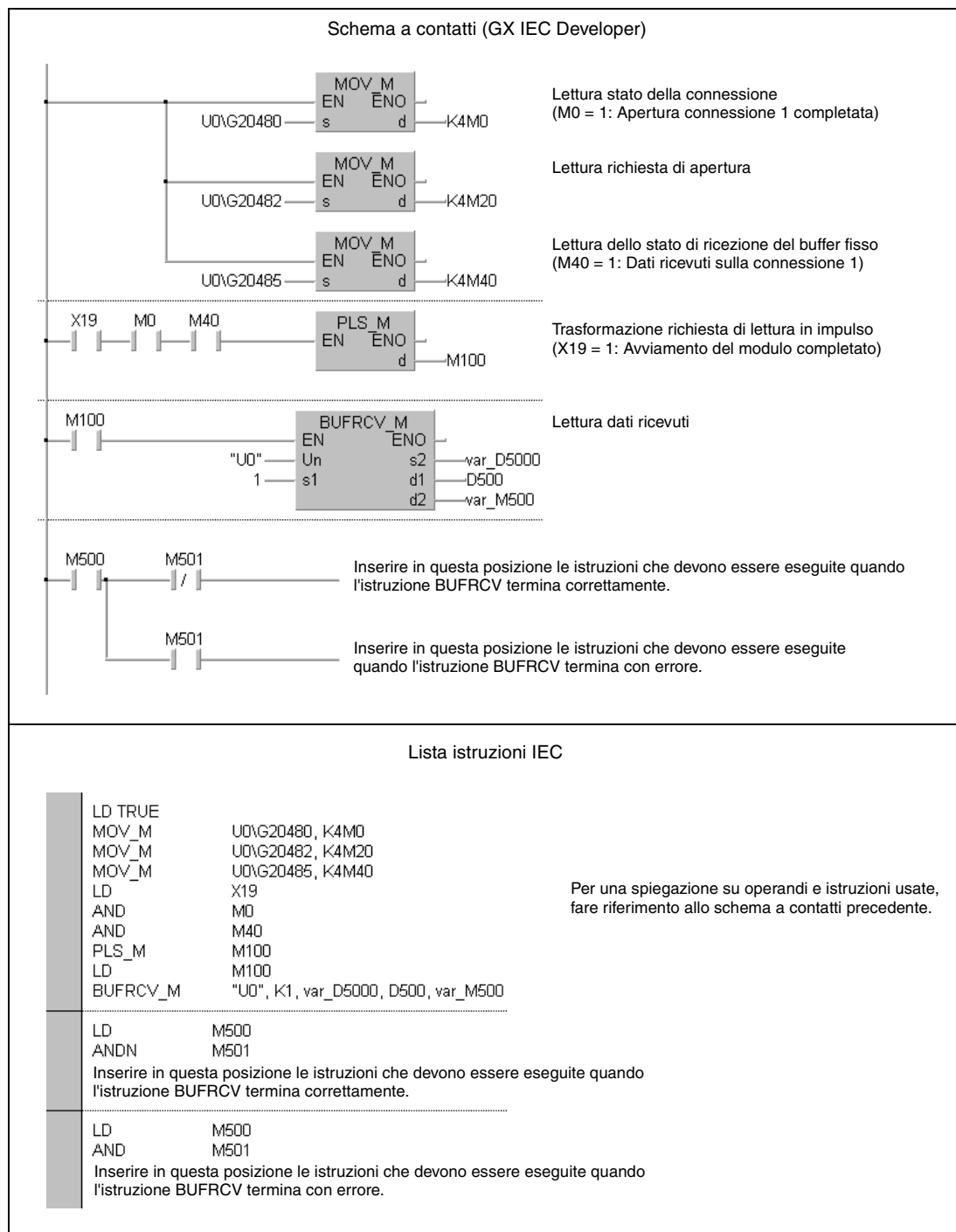
Errori di esecuzione Se l'istruzione BUFRCV termina in modo anomalo, l'operando a bit (d2)+1 viene attivato, e viene memorizzato un codice di errore in (s2)+1. Per ulteriori informazioni sui codici di errore, consultare i seguenti manuali:

- Con codice di errore 4FFF_H o inferiore, ulteriori informazioni sono disponibili nel capitolo 13 di questo manuale.
- Con codici di errore pari a C001_H o superiori, fare riferimento al manuale utente del modulo di interfaccia ETHERNET.

Esempio di programma**BUFRCV**

Il programma seguente legge i dati ricevuti dal buffer fisso per la connessione numero 1. I punti di ingresso/uscita da X/Y0 a X/Y1F sono occupati dal modulo ETHERNET.

- Editor IEC (questo programma di esempio è mostrato a pagina seguente con lista istruzioni MELSEC e schema a contatti di GX Developer).

**NOTA**

Negli editor IEC è necessario definire le variabili nello header della unità organizzativa di programma (POU). La mancata definizione delle variabili provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

- Lista istruzioni MELSEC e schema a contatti di GX Developer
 Per una spiegazione degli operandi e istruzioni utilizzate, vedere il programma di esempio per lo schema a contatti di GX IEC Developer nella pagina precedente.

Schema a contatti (GX Developer)

Lista istruzioni MELSEC

MELSEC	LD SM400 MOV U0\G20480 K4M0 MOV U0\G20482 K4M20 MOV U0\G20485 K4M40 LD X19 AND M0 AND M40 PLS M100 LD M100 ZP.BUFRCV "U0" K1 D5000 D500 M500
MELSEC	LD M500 ANI M501 Inserire in questa posizione le istruzioni che devono essere eseguite quando l'istruzione BUFRCV termina correttamente.
MELSEC	LD M500 AND M501 Inserire in questa posizione le istruzioni che devono essere eseguite quando l'istruzione BUFRCV termina con errore.

11.3.2 BUFRCVS


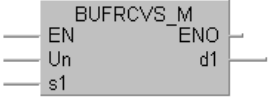

CPU

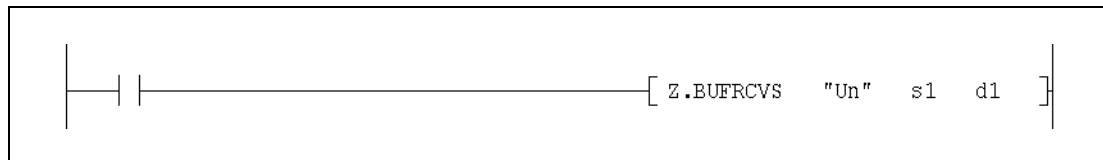
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					●

Operandi
MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	●	—	SM0	
d1	—	●	●	—	—	—	—	—	—		

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
 <p>Z.BUFRCVS "Un" s1 d1</p>		 <p>BUFRCVS_M "Un", s1, d1</p>

GX
Developer

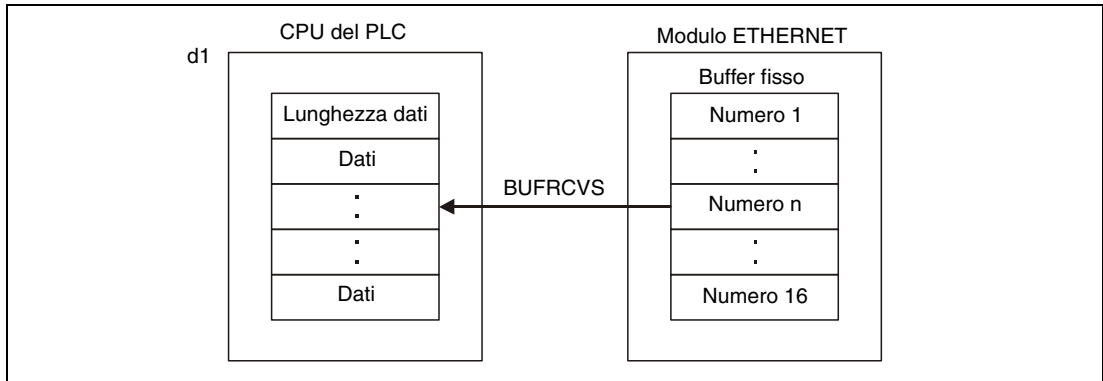
Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati	
„Un“	Indirizzo iniziale di I/O del modulo di interfaccia ETHERNET (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di XY100 viene impostato come „U10“)	da 0 a FE _H	Utente	BIN 16-bit	
s1	Numero connessione	da 1 a 16			
d1	Indirizzo iniziale dell'area operandi in cui vengono memorizzati i dati ricevuti.				
	Dati impostati	Significato	Descrizione	Campo	
	(d1)+0	Lunghezza dei dati ricevuti (Numero di word o byte letti dal buffer fisso)	Con procedura (dati binari):	da 1 a 1017 word	Sistema
			Con procedura (dati ASCII):	da 1 a 508 word	
Senza procedura (dati binari):			da 1 a 2016 byte		
(d1)+1 a (d1)+n	Dati ricevuti	I dati letti dal buffer fisso vengono memorizzati sequenzialmente in questa area in ordine crescente.	—	BIN 16-bit	

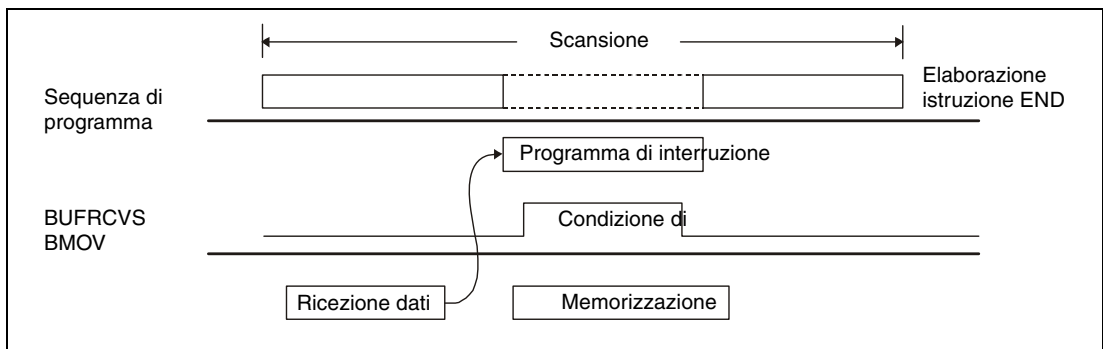
Funzioni **Letture dati ricevuti da buffer fisso (esecuzione dell'istruzione in un programma di interruzione)**

BUFRCVS Lettura dati

Con l'istruzione BUFRCVS, i dati inviati da una stazione esterna verso un modulo interfaccia ETHERNET tramite comunicazione con buffer fisso, possono essere letti dal modulo ETHERNET e memorizzati nella CPU del PLC. L'istruzione BUFRCVS viene eseguita nei programmi di interruzione, mentre l'istruzione BUFRCV viene usata nel programma principale. L'area in cui devono essere memorizzati i dati è specificata da d1:



L'elaborazione dell'istruzione BUFRCVS viene completata in una scansione. La figura seguente mostra la temporizzazione dell'istruzione BUFRCVS:



Per poter leggere da un programma di interruzione i dati ricevuti, è necessario impostare sia l'interruzione che il puntatore di interrupt con l'impostazione parametri di GX (IEC) Developer.

NOTE

Non è possibile leggere dati ricevuti dalla stessa connessione con l'istruzione BUFRCV nel programma principale e con l'istruzione BUFRCVS in un programma di interruzione.

L'istruzione BUFRCVS viene utilizzata anche con il modulo di comunicazione seriale QJ71C24 (vedi capitolo 11.1.1).

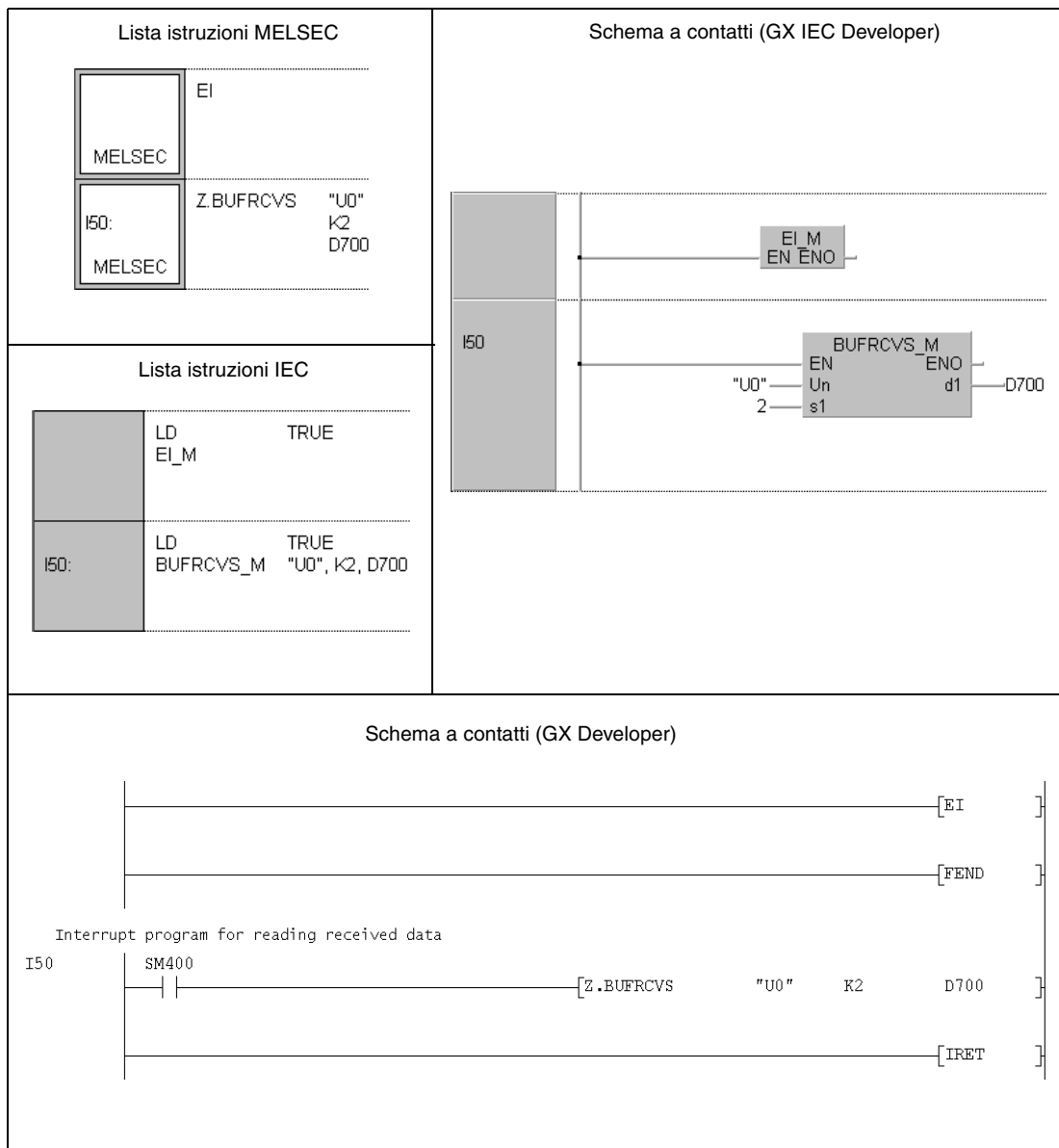
Errori di esecuzione

Se l'istruzione BUFRCVS viene completata con errori, viene attivato il flag di errore SM0, ed il codice di errore corrispondente viene memorizzato in SD0. Per ulteriori informazioni sui codici di errore, fare riferimento ai seguenti manuali:

- Con codice di errore 4FFF_H o inferiore, ulteriori informazioni sono disponibili nel capitolo 13 di questo manuale.
- Con codici di errore pari a C001_H o superiori, fare riferimento al manuale utente del modulo di interfaccia ETHERNET.

Esempio di programma BUFRCVS

Il programma seguente legge i dati ricevuti dal buffer fisso per la connessione numero 2. L'indirizzo iniziale di I/O del modulo ETHERNET è X/Y0.



11.3.3 BUFSND

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					●

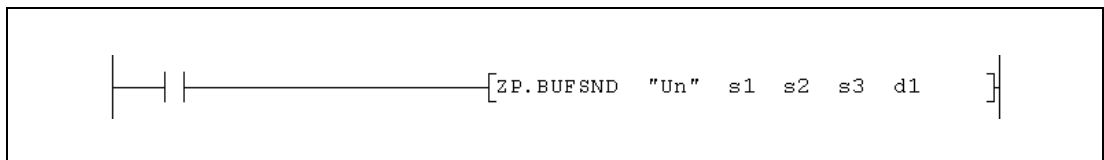
Operandi MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	●	—		
s2	—	●	●	—	—	—	—	—	—		
s3	—	●	●	—	—	—	—	—	—		
d1	●	●	●	—	—	—	—	—	—		

GX IEC Developer

<p style="text-align: center;">Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p style="margin: 0;">MELSEC</p> </div> <table style="margin-left: 20px; border-collapse: collapse;"> <tr> <td style="border-bottom: 1px dotted black; padding: 2px;">ZP.BUFSND</td> <td style="padding: 2px;">"Un"</td> </tr> <tr> <td style="padding: 2px;">s1</td> <td style="padding: 2px;"></td> </tr> <tr> <td style="padding: 2px;">s2</td> <td style="padding: 2px;"></td> </tr> <tr> <td style="padding: 2px;">s3</td> <td style="padding: 2px;"></td> </tr> <tr> <td style="padding: 2px;">d1</td> <td style="padding: 2px;"></td> </tr> </table>	ZP.BUFSND	"Un"	s1		s2		s3		d1		<p>Schema a contatti</p>	<p style="text-align: center;">Lista istruzioni IEC</p> <table style="margin-left: 20px; border-collapse: collapse;"> <tr> <td style="border-bottom: 1px dotted black; padding: 2px;">BUFSND_M</td> <td style="padding: 2px;">"Un", s1, s3, s2, d1</td> </tr> </table>	BUFSND_M	"Un", s1, s3, s2, d1
ZP.BUFSND	"Un"													
s1														
s2														
s3														
d1														
BUFSND_M	"Un", s1, s3, s2, d1													

GX Developer



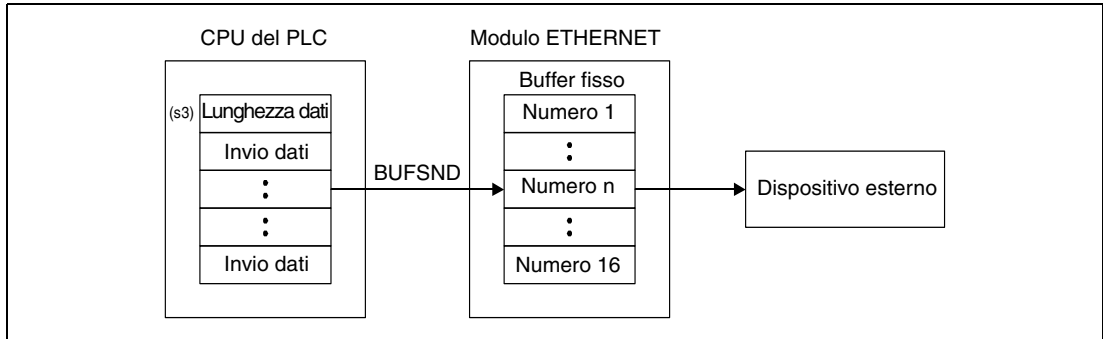
Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati	
„Un“	Indirizzo iniziale di I/O del modulo di interfaccia ETHERNET (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come „U10“)	da 0 a FE _H	Utente	BIN 16-bit	
s1	Numero connessione	da 1 a 16			
s2	Indirizzo iniziale dell'area operandi in cui sono contenuti i dati di controllo dell'istruzione.				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(s2)+0	Area di sistema	Usata dal sistema	—	Sistema
(s2)+1	Risultato dell'esecuzione dell'istruzione	Indica se si è verificato un errore durante l'esecuzione dell'istruzione: 0000 _H : Nessun errore Qualsiasi valore diverso da 0000 _H : Si è verificato un errore. Il valore memorizzato è un codice di errore, spiegato sia nel manuale utente del modulo interfaccia ETHERNET che nel capitolo 13 di questo manuale.			
s3	Indirizzo iniziale dell'area operandi contenente i dati da inviare.				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(s3)+0	Lunghezza dei dati da inviare	Indica la quantità di dati da trasferire nel buffer fisso quando viene utilizzata una procedura (dati binari) per la comunicazione.	da 1 a 1017 word	Utente
			Indica la quantità di dati da trasferire nel buffer fisso quando viene utilizzata una procedura (dati ASCII) per la comunicazione.	da 1 a 508 word	
Indica la quantità di dati da trasferire nel buffer fisso quando non viene utilizzata una procedura (dati binari) per la comunicazione.			da 1 a 2046 byte		
da (s3)+1 a (s3)+n	Dati da inviare	I dati contenuti in questa area vengono inviati al modulo ETHERNET.	—		
d1	Operando a bit attivato per una scansione al termine dell'istruzione BUFSND. (d)+1 segnala una eventuale terminazione anomala dell'istruzione.				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(d1)+0	Istruzione completata	Indica lo stato di completamento dell'istruzione BUFSND ON: Istruzione completata OFF: Istruzione non completata	—	Sistema
(d1)+1	Istruzione completata con errore	Indica lo stato di completamento anomalo dell'istruzione BUFSND ON: Completamento anomalo OFF: Completamento normale	—		

Funzioni **Invio dati verso buffer fisso**

BUFSND **Invio dati**

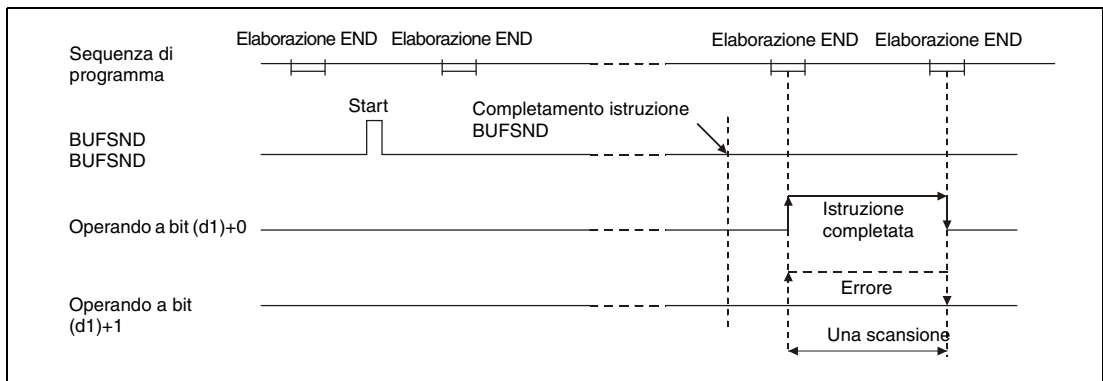
I dati da inviare tramite comunicazione con buffer fisso verso un dispositivo esterno collegato a un modulo di interfaccia ETHERNET vengono inviati per mezzo dell'istruzione BUFSND. I dati sono contenuti nella CPU del PLC a partire dall'operando indicato da (s3)+1:



Lo stato di esecuzione dell'istruzione BUFSND può essere controllato con gli operandi (d)+0 e (d)+1:

- L'operando a bit (d)+0 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione BUFSND, e si disattiva con l'elaborazione di END della scansione successiva.
- L'operando a bit (d)+1 segnala la presenza di un errore durante l'esecuzione di una istruzione BUFSND. Se l'istruzione termina normalmente, l'operando rimane OFF. Se si verifica un errore durante l'esecuzione dell'istruzione BUFSND, (d)+1 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione BUFSND, e si disattiva con l'elaborazione di END della scansione successiva.

La figura seguente mostra la temporizzazione durante l'esecuzione di una istruzione BUFSND:



L'istruzione BUFSND viene eseguita sul fronte di salita della variabile d'ingresso dell'istruzione.

Errori di esecuzione

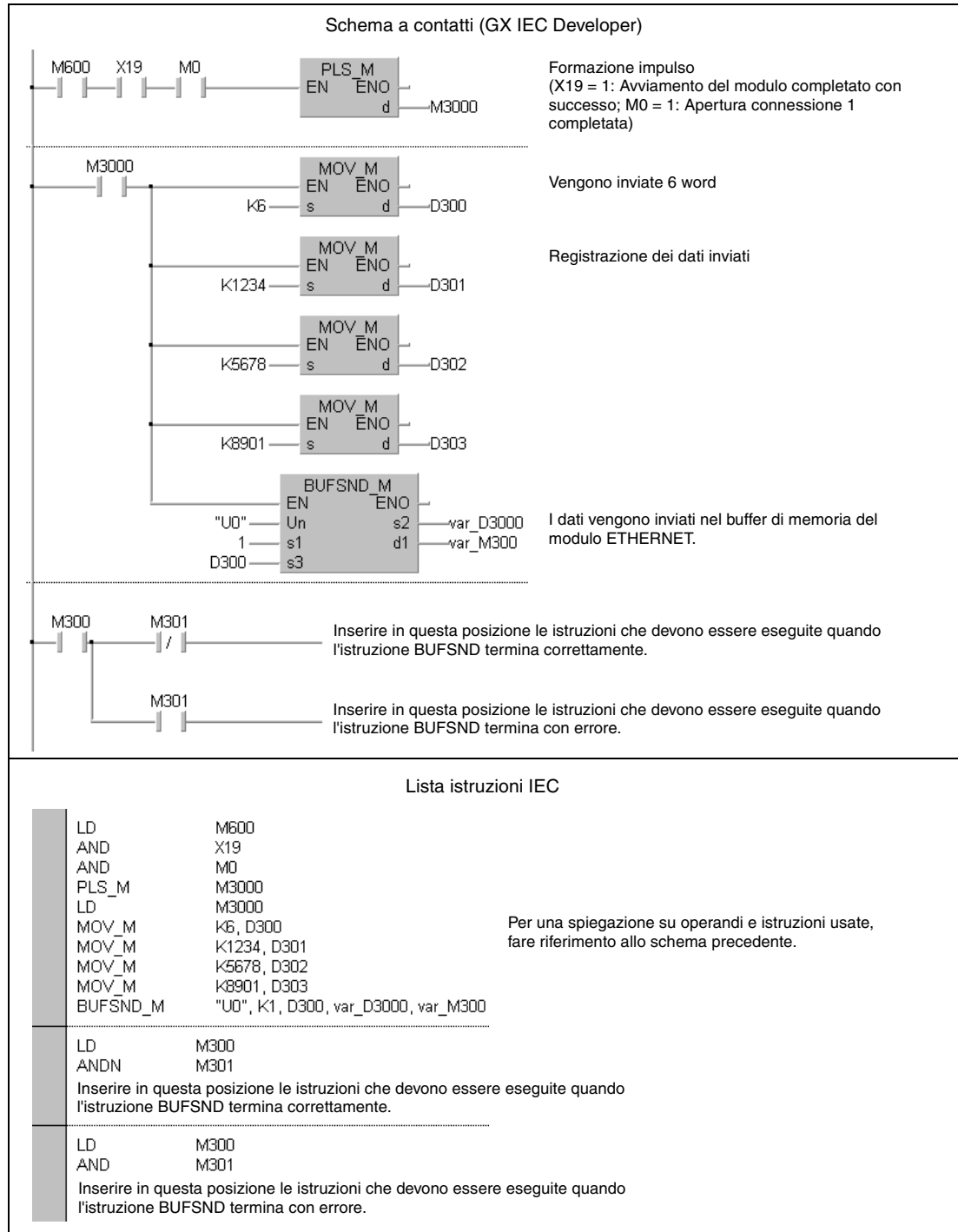
Se l'istruzione BUFSND termina in modo anomalo, l'operando a bit (d1)+1 viene attivato, e viene memorizzato un codice di errore in (s2)+1. Per ulteriori informazioni sui codici di errore, consultare i seguenti manuali:

- Con codice di errore 4FFF_H o inferiore, ulteriori informazioni sono disponibili nel capitolo 13 di questo manuale.
- Con codici di errore pari a C001_H o superiori, fare riferimento al manuale utente del modulo di interfaccia ETHERNET.

Esempio di programma BUFSND

Il programma seguente scrive dati nel buffer fisso per la connessione numero 1. L'indirizzo iniziale di I/O del modulo ETHERNET è X/Y0.

- Editor IEC (questo programma di esempio è mostrato a pagina seguente con lista istruzioni MELSEC e schema a contatti di GX Developer).

**NOTA**

Negli editor IEC è necessario definire le variabili nello header della unità organizzativa di programma (POU) La mancata definizione delle variabili provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

- Lista istruzioni MELSEC e schema a contatti di GX Developer
 Per una spiegazione degli operandi e istruzioni utilizzate, vedere il programma di esempio per lo schema a contatti di GX IEC Developer nella pagina precedente.

Schema a contatti (GX Developer)

Lista istruzioni MELSEC

MELSEC	LD M6000							
	AND X19							
	AND M0							
	PLS M3000							
	LD M3000							
	MOV K6	D300						
	MOV K1234	D301						
	MOV K5678	D302						
	MOV K8901	D303						
	ZP.BUFSND "U0"	K1	D3000	D300	M300			
MELSEC	LD M300							
	ANI M301							
	Inserire in questa posizione le istruzioni che devono essere eseguite quando l'istruzione BUFSND termina correttamente.							
MELSEC	LD M300							
	AND M301							
	Inserire in questa posizione le istruzioni che devono essere eseguite quando l'istruzione BUFSND termina con errore.							

11.3.4 OPEN

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					●

Operandi MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	●	—		
s2	—	●	●	—	—	—	—	—	—		
d1	●	●	●	—	—	—	—	—	—		

GX IEC Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p style="text-align: center;">MELSEC</p> </div> <p>ZP.OPEN "Un" s1 s2 d1</p>		<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p style="text-align: center;">OPEN_M "Un", s1, s2, d1</p> </div>

GX Developer

--

Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati
„Un“	Indirizzo iniziale di I/O del modulo di interfaccia ETHERNET (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come „U10“)	da 0 a FE _H	Utente	BIN 16-bit
s1	Numero connessione	da 1 a 16		

Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati	
s2	Indirizzo iniziale dell'area operandi in cui sono contenuti i dati di controllo dell'istruzione.				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(s2)+0	Sorgente per l'impostazione dei parametri	Indica l'impostazione da usare per l'apertura della connessione: <ul style="list-style-type: none"> • 0000_H: La connessione viene aperta con le impostazioni effettuate da GX (IEC) Developer. • 8000_H: La connessione viene aperta con le impostazioni contenute negli operandi da (s2)+2 a (s2)+9. 	0000 _H o 8000 _H	Utente
	(s2)+1	Risultato dell'esecuzione dell'istruzione	Indica se si è verificato un errore durante l'esecuzione dell'istruzione: 0000 _H : Nessun errore Qualsiasi valore diverso da 0000 _H : Si è verificato un errore. Il valore memorizzato è un codice di errore, spiegato sia nel manuale utente del modulo interfaccia ETHERNET che nel capitolo 13 di questo manuale.	—	Sistema
(s2)+2	Area impostazioni applicazione	I bit di questo operando vengono usati per impostare la connessione: <ul style="list-style-type: none"> • Bit 0: Uso dei buffer fissi 0: Il buffer è usato per la trasmissione, le comunicazioni con buffer fisso non sono eseguite 1: Il buffer è usato per la ricezione • Bit 1: Conferma esistenza destinazione 0: Nessuna conferma 1: Conferma • Bit 7: Impostazione apertura accoppiata 0: Non accoppiata 1: Accoppiata • Bit 8: Metodo (protocollo) di comunicazione 0: TCP/IP 1: UDP/IP • Bit 9: Comunicazione con buffer fisso 0: Con procedura 1: Senza procedura • Bit 13: Emissione interrupt alla ricezione di un buffer fisso 0: Nessuna interruzione 1: Emissione interruzione • Bit 14 e 15: Apertura attiva o passiva Bit 15/14 = 00: Apertura attiva o UDP/IP Bit 15/14 = 10: Apertura "unpassive" Bit 15/14 = 11: Apertura "full passive" 	Vedere la descrizione sulla sinistra.	Utente	
				BIN 16-bit	

Variabili

Dati impostati	Significato			Campo	Contenuto inserito da	Tipo dati
	Dati impostati	Significato	Descrizione			
s2	(s2)+3	Numero porta del modulo ETHERNET	Indica il numero di porta del modulo interfaccia ETHERNET.	408 _H a 1388 _H 138B _H a FFFE _H	Utente	BIN 16-bit
	(s2)+4 (s2)+5	Indirizzo IP di destinazione	Indirizzo IP del dispositivo esterno con cui si vuole comunicare. Se è impostato l'indirizzo IP FFFFFFFF _H , i dati vengono trasferiti in modalità broadcast.	1 _H a FFFFFFF _H		
	(s2)+6	Numero porta di destinazione	Numero della porta del dispositivo esterno con cui si vuole comunicare. (FFFF _H = broadcast simultaneo)	401 _H a FFFF _H		
	(s2)+7 a (s2)+9	Indirizzo ETHERNET di destinazione	Se il dispositivo esterno supporta la funzione ARP impostare 000000000000 _H o FFFFFFFF _H . Se il dispositivo esterno non supporta la funzione ARP, impostare l'indirizzo ETHERNET di destinazione.	Vedere la descrizione sulla sinistra.		
d1	Operando a bit attivato per una scansione al termine dell'istruzione OPEN. (d)+1 segnala una eventuale terminazione anomala dell'istruzione.					Bit
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da	
	(d1)+0	Istruzione completata	Indica lo stato di completamento dell'istruzione OPEN ON: Istruzione completata OFF: Istruzione non completata	—	Sistema	
(d1)+1	Istruzione completata con errore	Indica lo stato di completamento anomalo dell'istruzione OPEN ON: Completamento anomalo OFF: Completamento normale	—			

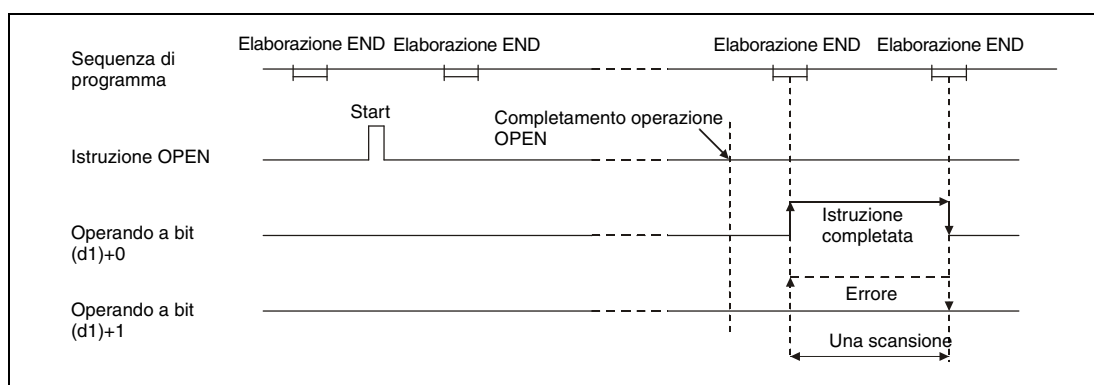
Funzioni **Apertura di una connessione****OPEN** **Apertura connessione**

Questa istruzione esegue il processo di apertura della connessione specificata da s1 per il modulo indicato da Un.

Lo stato di esecuzione dell'istruzione OPEN può essere controllato con gli operandi (d1)+0 e (d1)+1:

- L'operando a bit (d1)+0 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione OPEN, e si disattiva con l'elaborazione di END della scansione successiva.
- L'operando a bit (d1)+1 segnala la presenza di un errore durante l'esecuzione di una istruzione OPEN. Se l'istruzione termina normalmente, l'operando rimane OFF. Se si verifica un errore durante l'esecuzione dell'istruzione OPEN, (d1)+1 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione OPEN, e si disattiva con l'elaborazione di END della scansione successiva.

La figura seguente mostra la temporizzazione durante l'esecuzione di una istruzione OPEN:



L'istruzione OPEN viene eseguita sul fronte di salita della variabile d'ingresso dell'istruzione.

NOTA

Non eseguire contemporaneamente il processo di apertura/chiusura usando i segnali di ingresso/uscita e le istruzioni dedicate OPEN o CLOSE sulla stessa connessione. In questo caso si verificano malfunzionamenti.

Errori di esecuzione

Se si verifica un errore durante l'esecuzione dell'istruzione OPEN, l'operando a bit (d1)+1 si attiva e in (s2)+1 viene scritto un codice di errore. Per ulteriori informazioni sui codici di errore, fare riferimento ai manuali seguenti:

- Con codice di errore 4FFF_H o inferiore, ulteriori informazioni sono disponibili nel capitolo 13 di questo manuale.
- Con codici di errore pari a C001_H o superiori, fare riferimento al manuale utente del modulo di interfaccia ETHERNET.

Esempio di programma OPEN

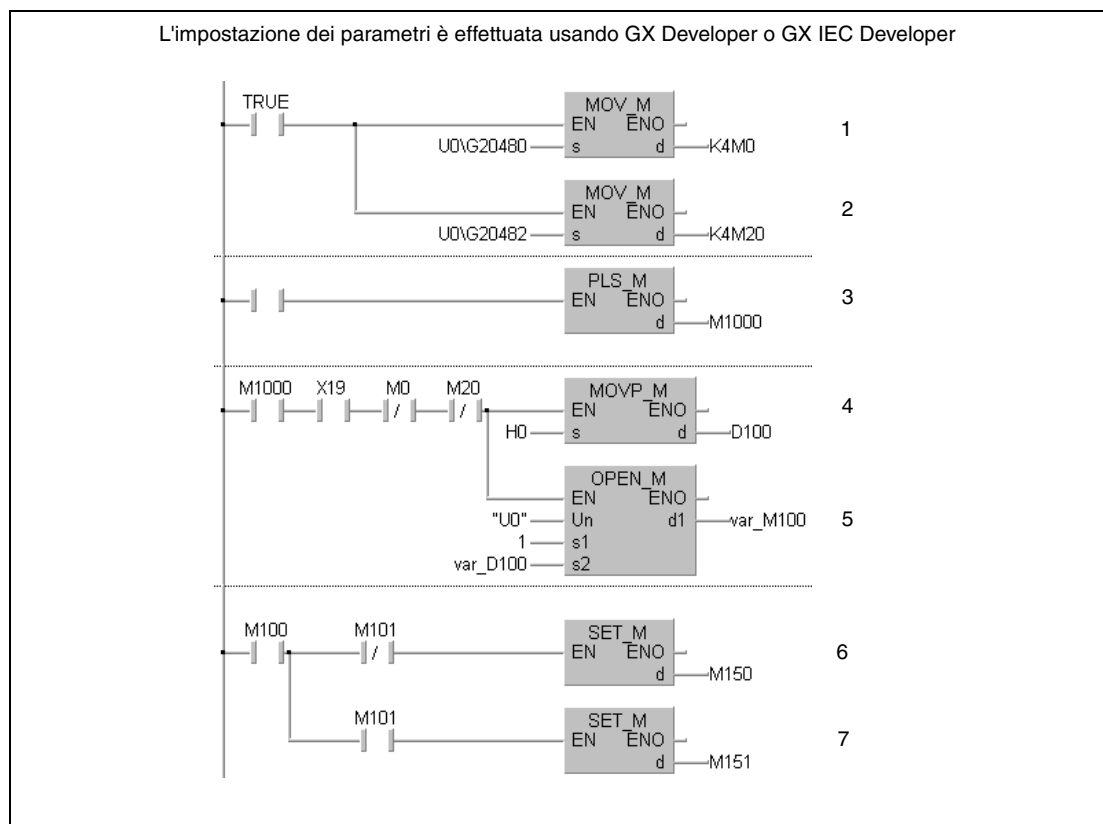
Il programma seguente apre la connessione numero 1 per una comunicazione TCP/IP. L'indirizzo iniziale di I/O è X/Y0.

NOTA

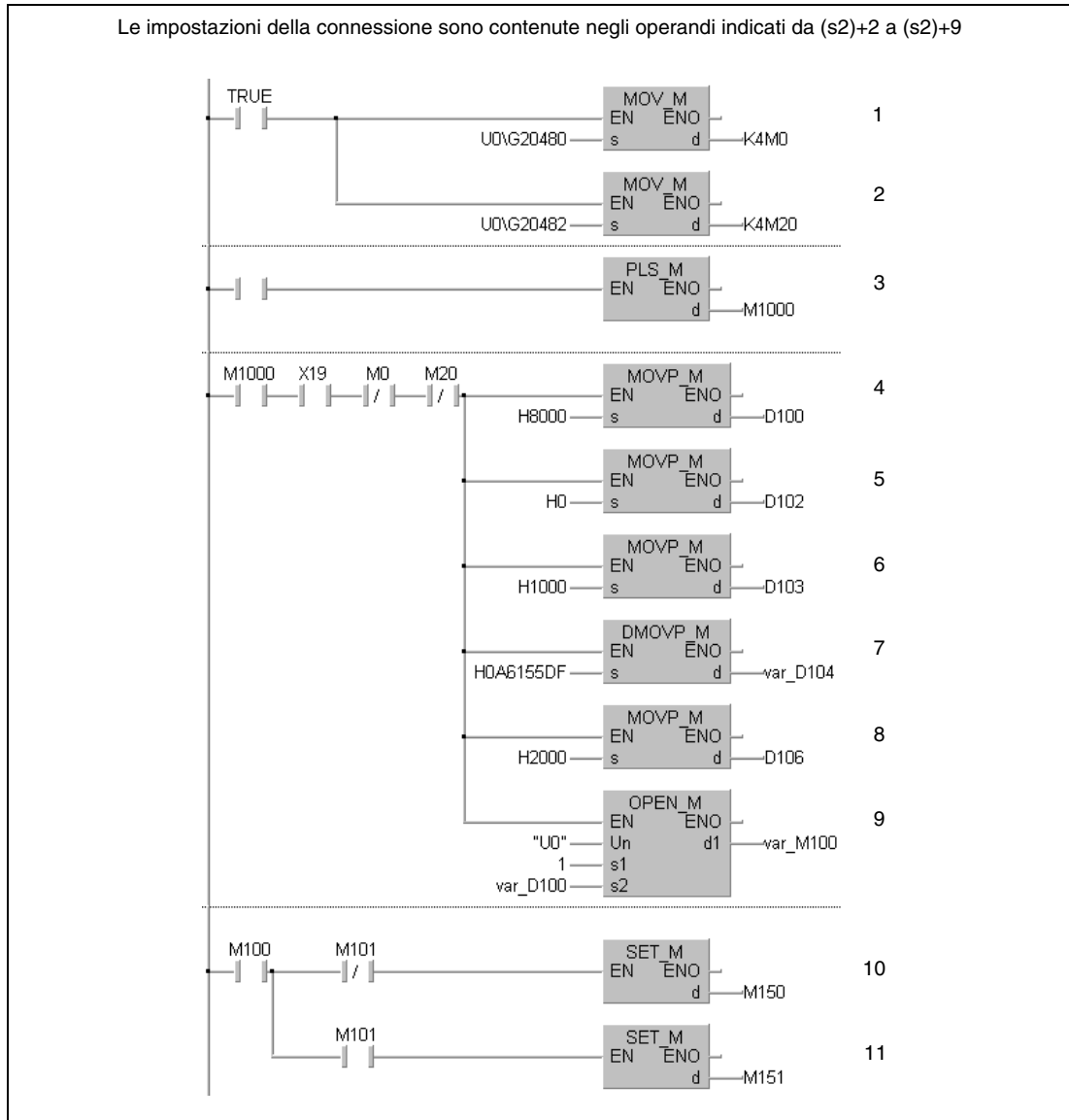
Negli editor IEC è necessario definire le variabili nello header della unità organizzativa di programma (POU) La mancata definizione delle variabili provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

- Schema a contatti (GX IEC Developer)

Per l'esempio seguente è necessario impostare anticipatamente i parametri con GX (IEC) Developer. Un altro esempio in cui le impostazioni vengono fatte con l'istruzione OPEN è mostrato a pagina seguente.



- 1 Lettura stato della connessione (M0 = 1: Apertura connessione 1 completata)
- 2 Lettura della richiesta apertura (M20 = 1: Richiesta apertura connessione 1)
- 3 Il segnale di apertura della connessione viene convertito in impulso.
- 4 La sorgente della parametrizzazione è impostata con (0000_H = impostazione esterna).
- 5 Apertura connessione 1
- 6 M150 è attivato al termine dell'apertura della connessione senza errori.
- 7 M151 si attiva in caso di errore durante l'apertura della connessione.



- ¹ Lettura stato della connessione (M0 = 1: Apertura connessione 1 completata)
- ² Lettura della richiesta apertura (M20 = 1: Richiesta apertura connessione 1)
- ³ Il segnale di apertura della connessione viene convertito in impulso.
- ⁴ La sorgente della parametrizzazione è impostata con (8000_H = parametri contenuti da (s2)+2 a (s2)+9))
- ⁵ Le impostazioni di applicazione sono memorizzate in (s2)+2.
- ⁶ Il numero di porta del modulo ETHERNET è scritto in (s2)+3.
- ⁷ L'indirizzo IP (10.97.85.223) del dispositivo esterno è contenuto in (s2)+4 e (s2)+5.
- ⁸ Il numero di porta del dispositivo esterno è contenuto in (s2)+6.
- ⁹ Apertura connessione 1
- ¹⁰ M150 è attivato al termine dell'apertura della connessione senza errori.
- ¹¹ M151 si attiva in caso di errore durante l'apertura della connessione.

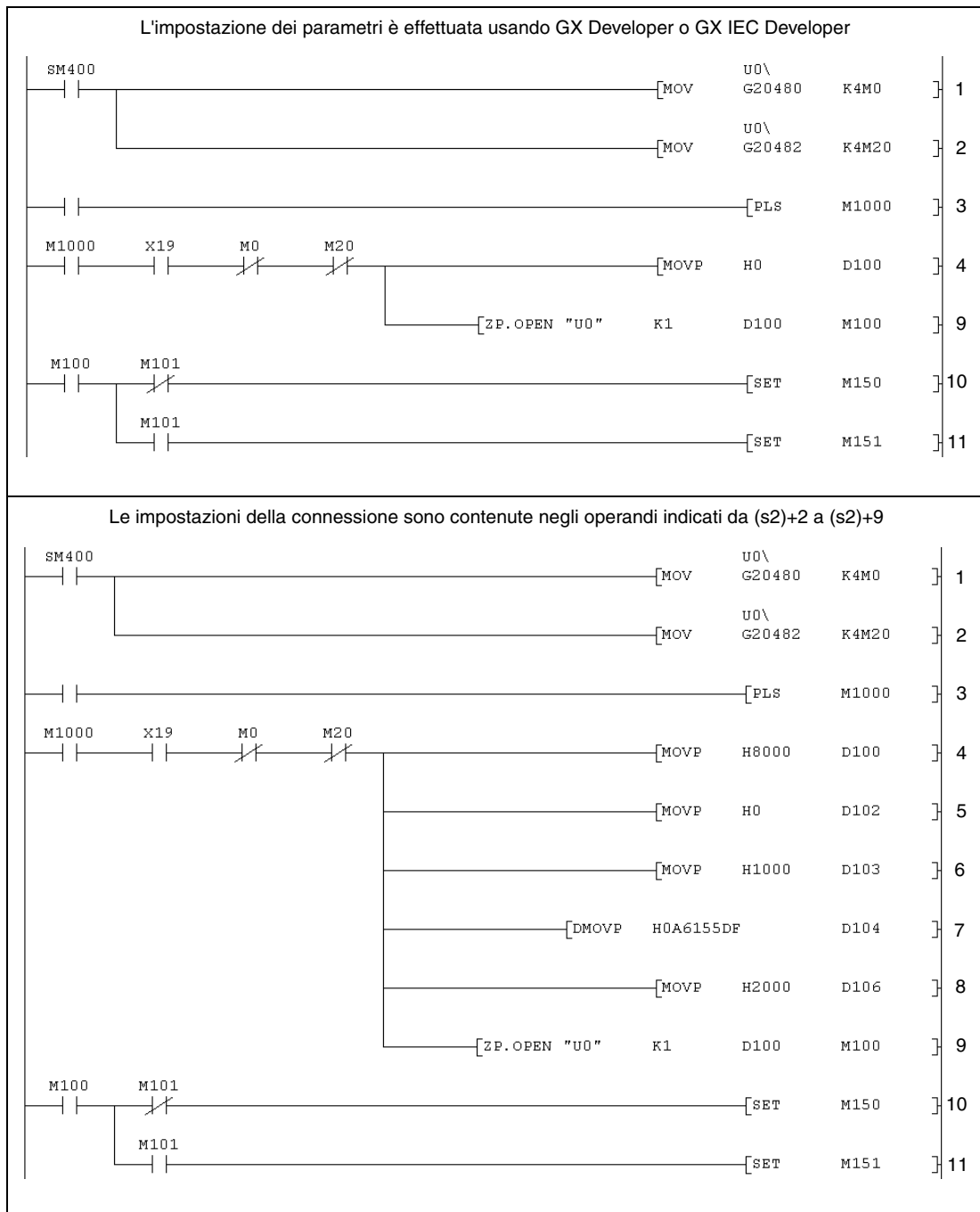
● Lista istruzioni IEC

L'impostazione dei parametri è effettuata usando GX Developer o GX IEC Developer				
LD	TRUE			
MOV_M	U0\G20480, K4M0	_____		1
MOV_M	U0\G20482, K4M20	_____		2
LD	M123			
PLS_M	M1000	_____		3
LD	M1000			
AND	X19			
ANDN	M0			
ANDN	M20			
MOVP_M	H0, D100	_____		4
OPEN_M	"U0", K1, var_D100, var_M100	_____		9
LD	M100			
ANDN	M101			
SET_M	M150	_____		10
LD	M100			
AND	M101			
SET_M	M151	_____		11

Le impostazioni della connessione sono contenute negli operandi indicati da (s2)+2 a (s2)+9				
LD	TRUE			
MOV_M	U0\G20480, K4M0	_____		1
MOV_M	U0\G20482, K4M20	_____		2
LD	M123			
PLS_M	M1000	_____		3
LD	M1000			
AND	X19			
ANDN	M0			
ANDN	M20			
MOVP_M	H8000, D100	_____		4
MOVP_M	H0, D102	_____		5
MOVP_M	H1000, D103	_____		6
DMOVP_M	H0A6155DF, var_D104	_____		7
MOVP_M	H2000, D106	_____		8
OPEN_M	"U0", K1, var_D100, var_M100	_____		9
LD	M100			
ANDN	M101			
SET_M	M150	_____		10
LD	M100			
AND	M101			
SET_M	M151	_____		11

- ¹ Lettura stato della connessione (M0 = 1: Apertura connessione 1 completata)
- ² Lettura della richiesta apertura (M20 = 1: Richiesta apertura connessione 1)
- ³ Il segnale di apertura della connessione viene convertito in impulso.
- ⁴ La sorgente della parametrizzazione è impostata con (0000_H = esterna, 8000_H = parametri contenuti da (s2)+2 a (s2)+9))
- ⁵ Le impostazioni di applicazione sono memorizzate in (s2)+2.
- ⁶ Il numero di porta del modulo ETHERNET è scritto in (s2)+3.
- ⁷ L'indirizzo IP (10.97.85.223) del dispositivo esterno è contenuto in (s2)+4 e (s2)+5.
- ⁸ Il numero di porta del dispositivo esterno è contenuto in (s2)+6.
- ⁹ Apertura connessione 1
- ¹⁰ M150 è attivato al termine dell'apertura della connessione senza errori.
- ¹¹ M151 si attiva in caso di errore durante l'apertura della connessione.

● Schema a contatti (GX Developer)



- 1 Lettura stato della connessione (M0 = 1: Apertura connessione 1 completata)
- 2 Lettura della richiesta apertura (M20 = 1: Richiesta apertura connessione 1)
- 3 Il segnale di apertura della connessione viene convertito in impulso.
- 4 La sorgente della parametrizzazione è impostata con (0000_H = esterna, 8000_H = parametri contenuti da (s2)+2 a (s2)+9))
- 5 Le impostazioni di applicazione sono memorizzate in (s2)+2.
- 6 Il numero di porta del modulo ETHERNET è scritto in (s2)+3.
- 7 L'indirizzo IP (10.97.85.223) del dispositivo esterno è contenuto in (s2)+4 e (s2)+5.
- 8 Il numero di porta del dispositivo esterno è contenuto in (s2)+6.
- 9 Apertura connessione 1
- 10 M150 è attivato al termine dell'apertura della connessione senza errori.
- 11 M151 si attiva in caso di errore durante l'apertura della connessione.

● Lista istruzioni MELSEC

L'impostazione dei parametri è effettuata usando GX Developer o GX IEC Developer		
MELSEC	LD SM400	
	MOV U0VG20480 K4M0	1
	MOV U0VG20482 K4M20	2
	LD M123	
	PLS M1000	3
	LD M1000	
	AND X19	
	ANI M0	
	ANI M20	
MOV H0 D100	4	
ZP.OPEN "U0" K1 D100 M100	9	
MELSEC	LD M100	
	ANI M101	
	SET M150	10
MELSEC	LD M100	
	AND M101	
	SET M151	11

Le impostazioni della connessione sono contenute negli operandi indicati da (s2)+2 a (s2)+9		
MELSEC	LD SM400	
	MOV U0VG20480 K4M0	1
	MOV U0VG20482 K4M20	2
	LD M123	
	PLS M1000	3
	LD M1000	
	AND X19	
	ANI M0	
	ANI M20	
MOV H8000 D100	4	
MOV H0 D102	5	
MOV H1000 D103	6	
DMOV H0A6155DF D104	7	
MOV H2000 D106	8	
ZP.OPEN "U0" K1 D100 M100	9	
MELSEC	LD M100	
	ANI M101	
	SET M150	10
MELSEC	LD M100	
	AND M101	
	SET M151	11

- 1 Lettura stato della connessione (M0 = 1: Apertura connessione 1 completata)
- 2 Lettura della richiesta apertura (M20 = 1: Richiesta apertura connessione 1)
- 3 Il segnale di apertura della connessione viene convertito in impulso.
- 4 La sorgente della parametrizzazione è impostata con (0000_H = esterna, 8000_H = parametri contenuti da (s2)+2 a (s2)+9))
- 5 Le impostazioni di applicazione sono memorizzate in (s2)+2.
- 6 Il numero di porta del modulo ETHERNET è scritto in (s2)+3.
- 7 L'indirizzo IP (10.97.85.223) del dispositivo esterno è contenuto in (s2)+4 e (s2)+5.
- 8 Il numero di porta del dispositivo esterno è contenuto in (s2)+6.
- 9 Apertura connessione 1
- 10 M150 è attivato al termine dell'apertura della connessione senza errori.
- 11 M151 si attiva in caso di errore durante l'apertura della connessione.

11.3.5 CLOSE

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					●

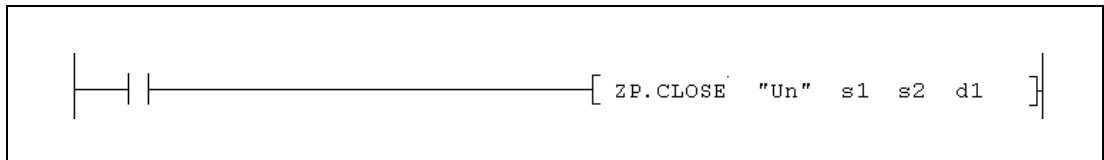
Operandi MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	●	—		
s2	—	●	●	—	—	—	—	—	—		
d1	●	●	●	—	—	—	—	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p>MELSEC</p> </div> <p>ZP.CLOSE "Un" s1 s2 d1</p>	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p>CLOSE_M</p> </div> <p>"Un", s1, s2, d1</p>
--	--------------------------	--

GX Developer



Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati	
„Un“	Indirizzo iniziale di I/O del modulo di interfaccia ETHERNET (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come „U10“)	da 0 a FE _H	Utente	BIN 16-bit	
s1	Numero della connessione	da 1 a 16			
s2	Indirizzo iniziale dell'area operandi in cui sono contenuti i dati di controllo dell'istruzione.				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(s2)+0	Area di sistema	Usata dal sistema	—	Sistema
(s2)+1	Risultato dell'esecuzione dell'istruzione	Indica se si è verificato un errore durante l'esecuzione dell'istruzione: 0000 _H : Nessun errore Qualsiasi valore diverso da 0000 _H : Si è verificato un errore. Il valore memorizzato è un codice di errore, spiegato sia nel manuale utente del modulo interfaccia ETHERNET che nel capitolo 13 di questo manuale.			
d1	Operando a bit attivato per una scansione al termine dell'istruzione CLOSE. (d)+1 segnala una eventuale terminazione anomala dell'istruzione.				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(d1)+0	Istruzione completata	Indica lo stato di completamento dell'istruzione CLOSE ON: Istruzione completata OFF: Istruzione non completata	—	Sistema
(d1)+1	Istruzione completata con errore	Indica l'occorrenza di un errore durante l'elaborazione dell'istruzione CLOSE ON: Completamento anomalo OFF: Completamento normale	—		

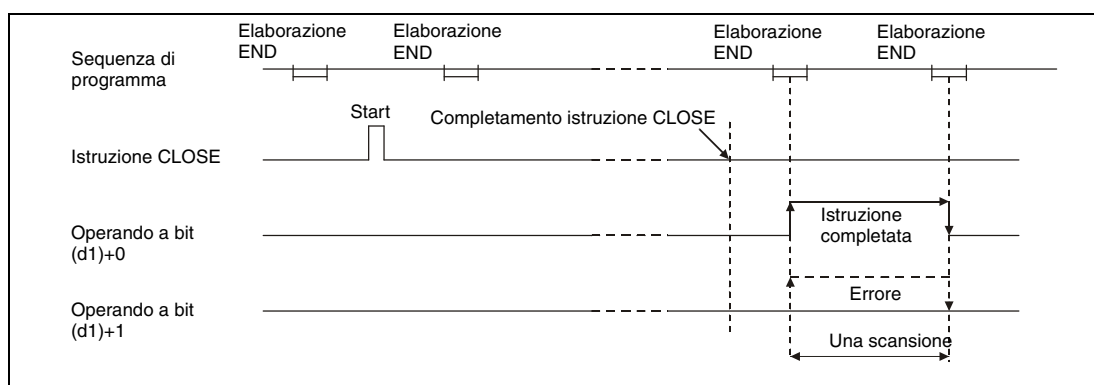
Funzioni **Chiusura di una connessione****CLOSE** **Chiusura connessione**

Questa istruzione chiude la connessione specificata da s1 per il modulo indicato da Un (scollegamento connessione).

Lo stato di esecuzione dell'istruzione CLOSE può essere controllato con gli operandi (d1)+0 e (d1)+1:

- L'operando a bit (d1)+0 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione CLOSE, e si disattiva con l'elaborazione di END della scansione successiva.
- L'operando a bit (d1)+1 segnala la presenza di un errore durante l'esecuzione di una istruzione CLOSE. Se l'istruzione termina normalmente, l'operando rimane OFF. Se si verifica un errore durante l'esecuzione dell'istruzione CLOSE, (d1)+1 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione CLOSE, e si disattiva con l'elaborazione di END della scansione successiva.

La temporizzazione dell'istruzione CLOSE viene mostrata nella figura seguente:



L'istruzione CLOSE viene eseguita sul fronte di salita della variabile d'ingresso dell'istruzione.

NOTA

Non eseguire contemporaneamente il processo di apertura/chiusura usando i segnali di ingresso/uscita e le istruzioni dedicate OPEN o CLOSE sulla stessa connessione. In questo caso si verificano malfunzionamenti.

Errori di esecuzione

Se si verifica un errore durante l'esecuzione dell'istruzione CLOSE, l'operando a bit (d1)+1 si attiva e in (s2)+1 viene scritto un codice di errore. Per ulteriori informazioni sui codici di errore, fare riferimento ai manuali seguenti:

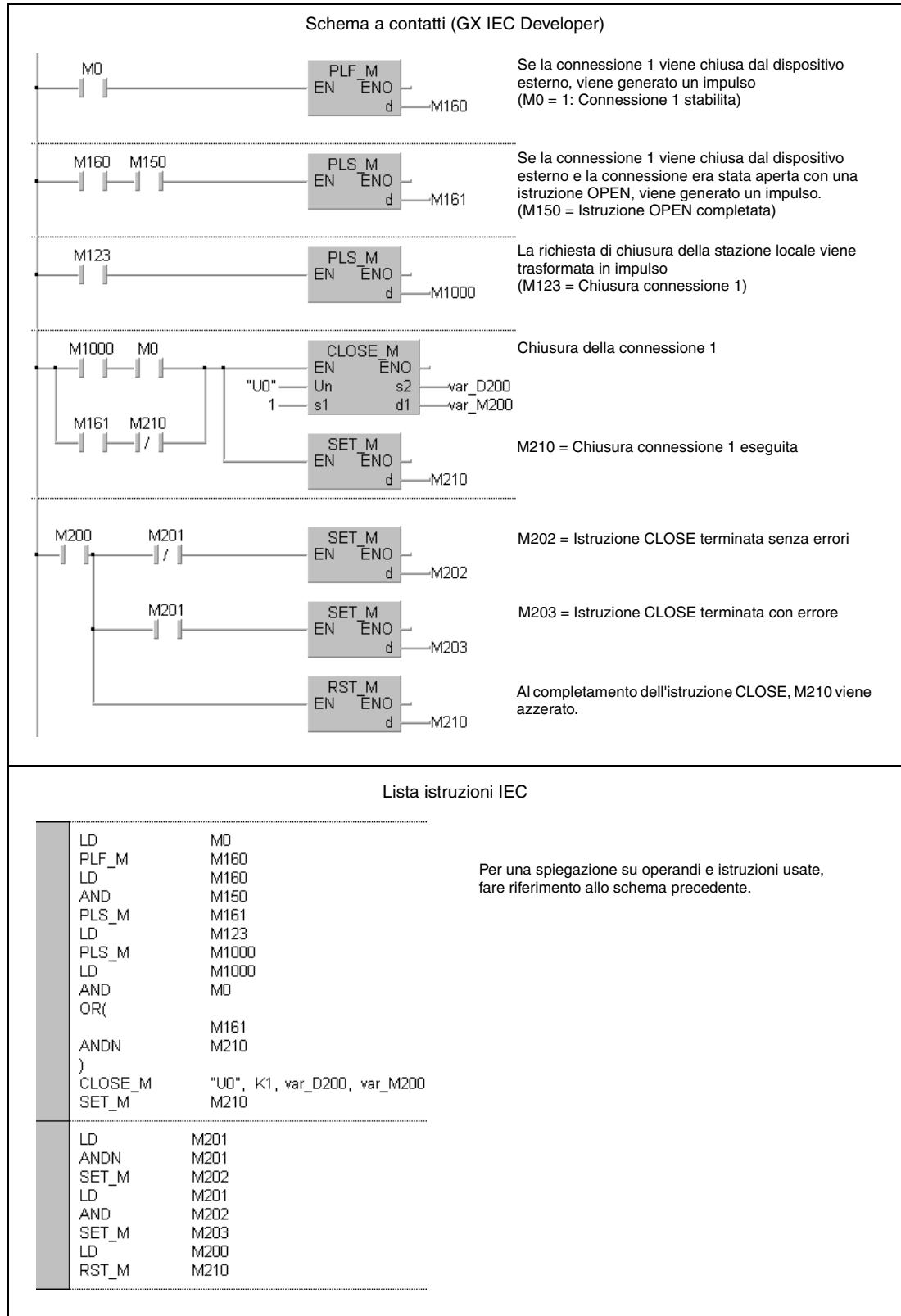
- Con codice di errore 4FFF_H o inferiore, ulteriori informazioni sono disponibili nel capitolo 13 di questo manuale.
- Con codici di errore pari a C001_H o superiori, fare riferimento al manuale utente del modulo di interfaccia ETHERNET.

Esempio di programma

CLOSE

Il programma seguente chiude la connessione numero 1 sul modulo ETHERNET con indirizzo iniziale di I/O X/Y0.

- Editor IEC (questo programma di esempio è mostrato a pagina seguente con lista istruzioni MELSEC e schema a contatti di GX Developer).



NOTA

Negli editor IEC è necessario definire le variabili nello header della unità organizzativa di programma (POU) La mancata definizione delle variabili provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

- Lista istruzioni MELSEC e schema a contatti di GX Developer
Per una spiegazione degli operandi e istruzioni utilizzate, vedere il programma di esempio per lo schema a contatti di GX IEC Developer nella pagina precedente.

Schema a contatti (GX Developer)

Lista istruzioni MELSEC

MELSEC	LD	M0			
	PLF	M160			
	LD	M160			
	AND	M150			
	PLS	M161			
	LD	M123			
	PLS	M1000			
	LD	M1000			
	AND	M0			
	LD	M161			
	ANI	M210			
	ORB				
	ZP.CLOSE	"U0"	K1	D200	M200
SET	M210				
<hr/>					
MELSEC	LD	M200			
	MPS				
	ANI	M201			
	SET	M202			
	MRD				
	AND	M201			
	SET	M203			
	MPP				
RST	M210				

11.3.6 ERRCLR

CPU

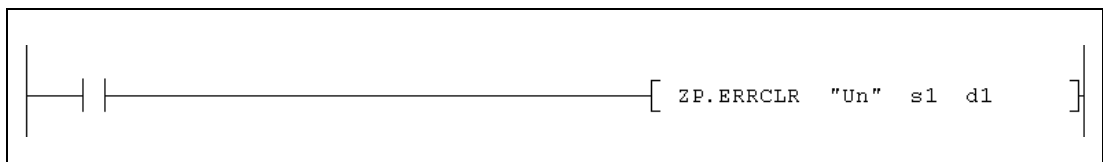
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					●

Operandi
MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	—	SM0	
d1	●	●	●	—	—	—	—	—	—		

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <div style="border: 1px solid black; padding: 2px; display: inline-block;">MELSEC</div> <div style="margin-left: 10px;">ZP.ERRCLR "Un" s1 d1</div> </div>		<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <div style="border: 1px solid black; padding: 2px; display: inline-block;">ERRCLR_M</div> <div style="margin-left: 10px;">"U0", s1, d1</div> </div>

GX
Developer

Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati	
„Un“	Indirizzo iniziale di I/O del modulo di interfaccia ETHERNET (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come „U10“)	da 0 a FE _H	Utente	BIN 16-bit	
s1	Indirizzo iniziale dell'area operandi in cui sono contenuti i dati di controllo dell'istruzione.				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(s1)+0	Area di sistema	Usata dal sistema	—	Sistema
	(s1)+1	Risultato dell'esecuzione dell'istruzione	Indica se si è verificato un errore durante l'esecuzione dell'istruzione: 0000 _H : Nessun errore Qualsiasi valore diverso da 0000 _H : Si è verificato un errore. Il valore memorizzato è un codice di errore, spiegato sia nel manuale utente del modulo interfaccia ETHERNET che nel capitolo 13 di questo manuale.		
	(s1)+2	Informazione di errore da cancellare	A seconda del valore inserito, viene cancellato un codice di errore memorizzato nel buffer di memoria ed il LED „ERR.“ del modulo ETHERNET si spegne. Fare riferimento alla descrizione di pagina seguente.	0000 _H 0001 _H a 0016 _H 0100 _H 0101 _H 0102 _H 0103 _H FFFF _H	Utente
	(s1)+3	Funzione	Sceglie se cancellare un codice di errore o spegnere il LED „ERR.“. Fare riferimento alla descrizione di pagina seguente.	0000 _H o FFFF _H	
(s1)+4 a (s1)+7	Area di sistema	Usata dal sistema	—	Sistema	
d1	Operando a bit attivato per una scansione al termine dell'istruzione ERRCLR. (d1)+1 segnala una eventuale terminazione anomala dell'istruzione.				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(d1)+0	Istruzione completata	Indica lo stato di completamento dell'istruzione ERRCLR ON: Istruzione completata OFF: Istruzione non completata	—	Sistema
(d1)+1	Istruzione completata con errore	Indica l'occorrenza di un errore durante l'elaborazione dell'istruzione ERRCLR ON: Completamento anomalo OFF: Completamento normale	—		

Funzioni Cancellazione di un codice di errore e spegnimento del LED „ERR.“**ERRCLR Operazione di cancellazione**

L'istruzione ERRCLR cancella un codice di errore contenuto nel buffer di memoria del modulo interfaccia ETHERNET. Se il LED „ERR.“ sul frontalino della CPU è acceso, la spia si spegne quando viene processata l'istruzione ERRCLR. L'istruzione cancella anche le zone del buffer di memoria in cui viene memorizzato lo stato della comunicazione.

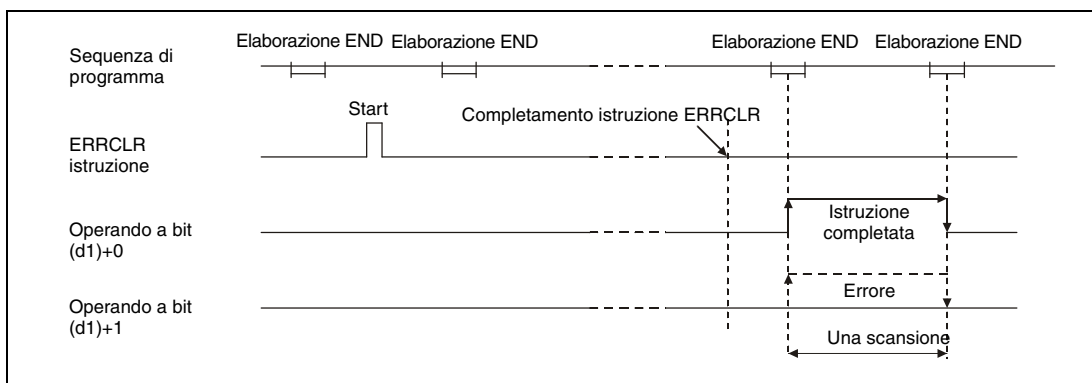
La zona del buffer di memoria che viene cancellata dipende dal contenuto degli operandi indicati da (s1)+2 e (s2)+3:

Zona errore o stato della comunicazione		Contenuto degli operandi		Azione intrapresa
		(s1)+2	(s2)+3	
Errore iniziale		0000 _H	0000 _H	<ul style="list-style-type: none"> Viene cancellato l'indirizzo 69_H del buffer di memoria. Il LED "ERR." viene spento.
Errore nell'apertura di una connessione		0001 _H a 0016 _H (Numero della connessione)		<ul style="list-style-type: none"> Viene cancellato il dato all'indirizzo del buffer di memoria relativo alla connessione con errori (7C_H, 86_H...). Il LED "ERR." viene spento.
Error log		0100 _H	FFFF _H	<ul style="list-style-type: none"> La registrazione errori (indirizzi da E3_H a 174_H) viene cancellata.
Stato della comunicazione	Stato del protocollo	0101 _H		<ul style="list-style-type: none"> Gli indirizzi del buffer di memoria da 178_H a 1FF_H vengono cancellati.
	Stato ricezione email	0102 _H		<ul style="list-style-type: none"> Gli indirizzi del buffer di memoria da 5871_H a 5B38_H vengono cancellati.
	Stato invio email	0103 _H		<ul style="list-style-type: none"> Gli indirizzi del buffer di memoria da 5B39_H a 5CA0_H vengono cancellati.
Tutte le zone dei codici di errore o dello stato della comunicazione		FFFF _H	<ul style="list-style-type: none"> Vengono cancellate tutte le zone di memoria sopra citate. Il LED "ERR." viene spento. 	

Lo stato di esecuzione dell'istruzione ERRCLR può essere controllato con gli operandi (d1)+0 e (d1)+1:

- L'operando a bit (d1)+0 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione ERRCLR, e si disattiva con l'elaborazione di END della scansione successiva.
- L'operando a bit (d1)+1 segnala la presenza di un errore durante l'esecuzione di una istruzione ERRCLR. Se l'istruzione termina normalmente, l'operando rimane OFF. Se si verifica un errore durante l'esecuzione dell'istruzione ERRCLR, (d1)+1 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione ERRCLR, e si disattiva con l'elaborazione di END della scansione successiva.

La temporizzazione durante l'esecuzione dell'istruzione ERRCLR è mostrato di seguito:



Errori di esecuzione

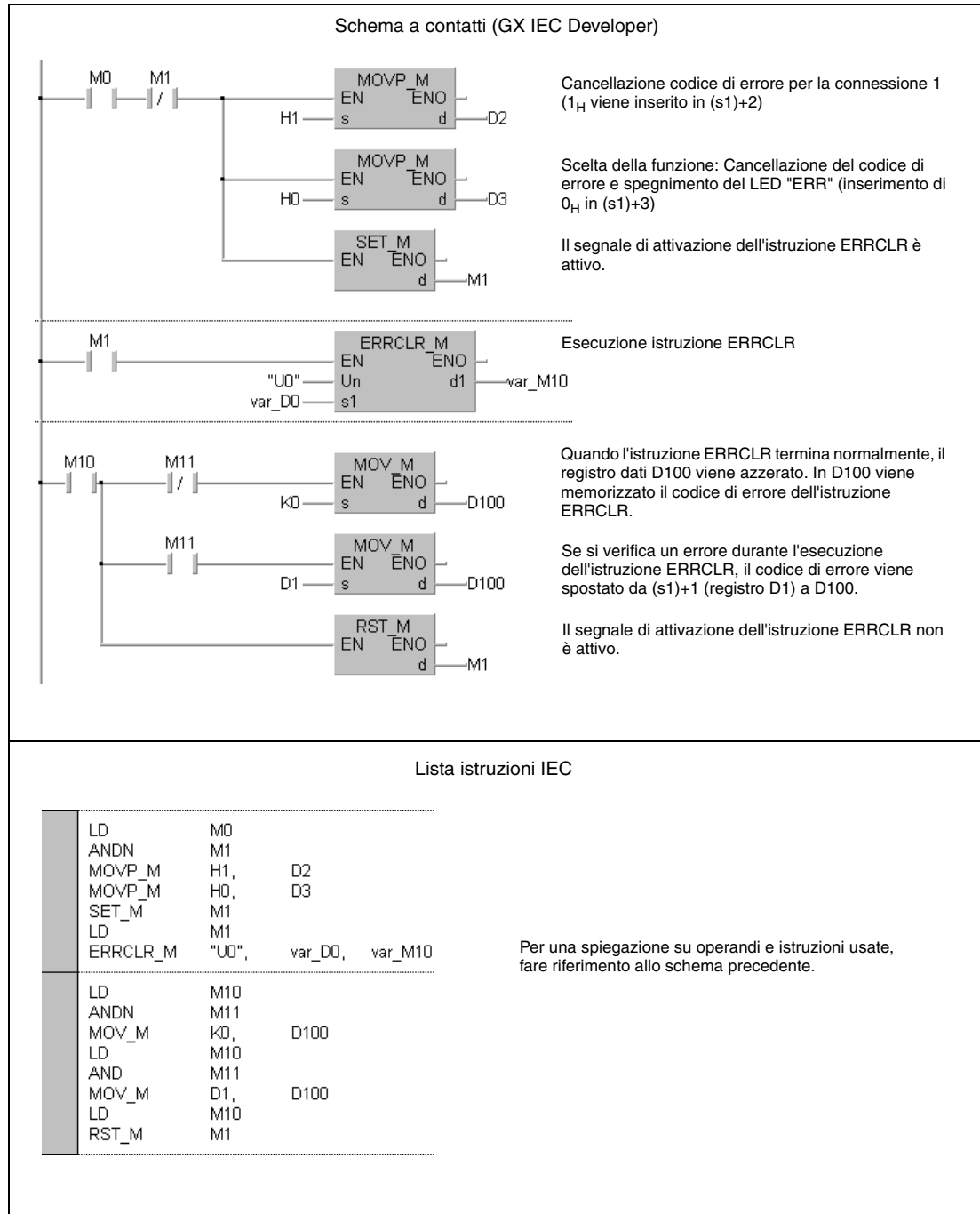
Se si verifica un errore durante l'esecuzione dell'istruzione ERRCLR, l'operando a bit (d1)+1 si attiva e in (s1)+1 viene scritto un codice di errore. Per ulteriori informazioni sui codici di errore, fare riferimento ai manuali seguenti:

- Con codice di errore 4FFF_H o inferiore, ulteriori informazioni sono disponibili nel capitolo 13 di questo manuale.
- Con codici di errore pari a C001_H o superiori, fare riferimento al manuale utente del modulo di interfaccia ETHERNET.

Esempio di programma ERRCLR

Il programma seguente viene usato per cancellare il codice di errore relativo alla connessione 1. Il modulo ETHERNET occupa gli ingressi e uscite a partire da X/Y0.

- Editor IEC (questo programma di esempio è mostrato a pagina seguente con lista istruzioni MELSEC e schema a contatti di GX Developer).

**NOTA**

Negli editor IEC è necessario definire le variabili nello header della unità organizzativa di programma (POU) La mancata definizione delle variabili provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

- Lista istruzioni MELSEC e schema a contatti di GX Developer
 Per una spiegazione degli operandi e istruzioni utilizzate, vedere il programma di esempio per lo schema a contatti di GX IEC Developer nella pagina precedente.

Schema a contatti (GX Developer)

The diagram shows a ladder logic network with three main rungs. The first rung starts with a normally open contact M0 and a normally closed contact M1. This rung branches into three parallel paths: the top path contains instruction [MOV H1 D2], the middle path contains [MOV H0 D3], and the bottom path contains [SET M1]. The second rung starts with a normally open contact M1 and contains instruction [ZP.ERRCLR "U0" D0 M10]. The third rung starts with a normally open contact M10 and a normally closed contact M11. This rung branches into two parallel paths: the top path contains instruction [MOV K0 D100] and the bottom path contains [MOV D1 D100]. A final path from the M11 contact contains instruction [RST M1].

Lista istruzioni MELSEC

MELSEC	LD	M0		
	ANI	M1		
	MOV	H1	D2	
	MOV	H0	D3	
	SET	M1		
	LD	M1		
	ZP.ERRCLR	"U0"	D0	M10
MELSEC	LD	M10		
	MPS			
	ANI	M11		
	MOV	K0	D100	
	MRD			
	AND	M11		
	MOV	D1	D100	
	MPP			
RST	M1			

11.3.7 ERRRD


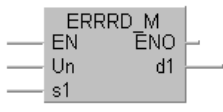

CPU

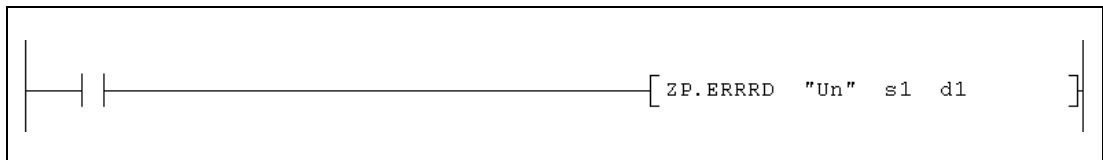
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					●

Operandi
MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	—	SM0	
d1	●	●	●	—	—	—	—	—	—		

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC
 <p>ZP.ERRRD "Un" s1 d1</p>	 <p>ERRRD_M EN ENO Un d1 s1</p>	 <p>ERRRD_M "Un", s1, d1</p>

GX
Developer

Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati	
„Un“	Indirizzo iniziale di I/O del modulo di interfaccia ETHERNET (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come „U10“)	da 0 a FE _H	Utente	BIN 16-bit	
s1	Indirizzo iniziale dell'area operandi in cui sono contenuti i dati di controllo dell'istruzione.				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(s1)+0	Area di sistema	Usata dal sistema		
	(s1)+1	Risultato dell'esecuzione dell'istruzione	Indica se si è verificato un errore durante l'esecuzione dell'istruzione: 0000 _H : Nessun errore Qualsiasi valore diverso da 0000 _H : Si è verificato un errore. Il valore memorizzato è un codice di errore, spiegato sia nel manuale utente del modulo interfaccia ETHERNET che nel capitolo 13 di questo manuale.	—	Sistema
	(s1)+2	Codice errore da leggere	A seconda del valore inserito, viene letto un codice di errore memorizzato nel buffer di memoria: • 0000 _H : Codice di errore iniziale inserito nel buffer di memoria all'indirizzo 69 _H . • 0001 _H a 0016 _H : Codice di errore per la connessione con questo numero (indirizzo buffer di memoria 7C _H , 86 _H ...)	0000 _H 0001 _H a 0016 _H	Utente
	(s1)+3	Funzione	Letture dell'ultimo codice di errore emesso	0000 _H	
	(s1)+4	Letture codice di errore	Memorizza il codice di errore letto dal modulo ETHERNET 0000 _H = Nessun errore Diverso da 0000 _H : codice di errore	—	Sistema
	(s1)+5 a (s1)+7	Area di sistema	Usata dal sistema	—	Sistema
d1	Operando a bit attivato per una scansione al termine dell'istruzione ERRRD. (d)+1 segnala una eventuale terminazione anomala dell'istruzione.				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(d1)+0	Istruzione completata	Indica lo stato di completamento dell'istruzione ERRRD ON: Istruzione completata OFF: Istruzione non completata	—	Sistema
(d1)+1	Istruzione completata con errore	Indica l'occorrenza di un errore durante l'elaborazione dell'istruzione ERRRD ON: Completamento anomalo OFF: Completamento normale	—		

Funzioni **Lettura di un codice di errore da un modulo ETHERNET****ERRRD** **Lettura codice di errore**

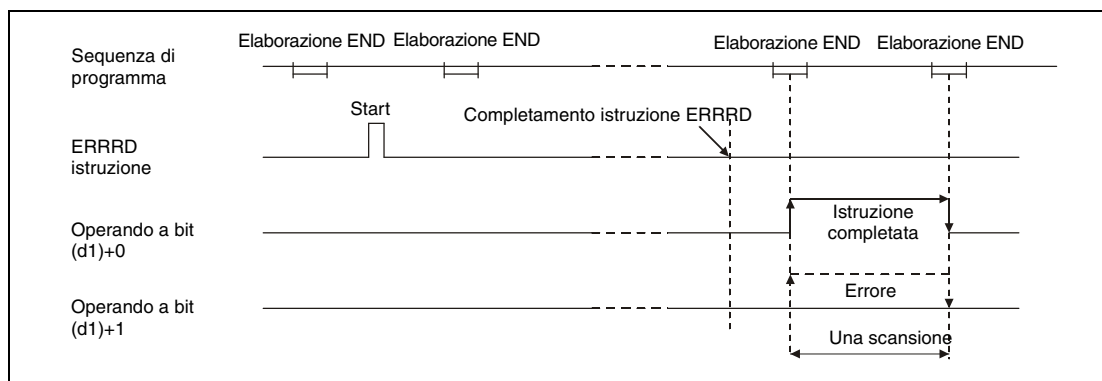
Questa istruzione legge il codice di errore contenuto nel buffer di memoria del modulo interfaccia ETHERNET con indirizzo iniziale di I/O indicato da „Un“.

L'operando indicato da (s1)+2 contiene informazioni sull'indirizzo del buffer di memoria da cui leggere.

Lo stato di esecuzione dell'istruzione ERRRD può essere controllato con gli operandi (d1)+0 e (d1)+1:

- L'operando a bit (d1)+0 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione ERRRD, e si disattiva con l'elaborazione di END della scansione successiva.
- L'operando a bit (d1)+1 segnala la presenza di un errore durante l'esecuzione di una istruzione ERRRD. Se l'istruzione termina normalmente, l'operando rimane OFF. Se si verifica un errore durante l'esecuzione dell'istruzione ERRRD, (d1)+1 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione ERRRD, e si disattiva con l'elaborazione di END della scansione successiva.

La figura seguente mostra la temporizzazione durante l'esecuzione di una istruzione ERRRD:

**Errori di esecuzione**

Se si verifica un errore durante l'esecuzione dell'istruzione ERRRD, l'operando a bit (d1)+1 si attiva e in (s1)+1 viene scritto un codice di errore. Per ulteriori informazioni sui codici di errore, fare riferimento ai manuali seguenti:

- Con codice di errore $4FFF_H$ o inferiore, ulteriori informazioni sono disponibili nel capitolo 13 di questo manuale.
- Con codici di errore pari a $C001_H$ o superiori, fare riferimento al manuale utente del modulo di interfaccia ETHERNET.

Esempio di programma

ERRRD

Il programma seguente legge il codice di errore emesso in caso di fallimento nell'apertura della connessione 1. Il modulo ETHERNET ha indirizzo iniziale di I/O X/Y0.

- Editor IEC (questo programma di esempio è mostrato a pagina seguente con lista istruzioni MELSEC e schema a contatti di GX Developer).

Schema a contatti (GX IEC Developer)

Letture codice di errore per la connessione 1 (1_H viene inserito in (s1)+2)

Impostazione della funzione: Lettura ultimo codice di errore (0_H viene inserito in (s1)+3)

Il segnale di attivazione dell'istruzione ERRRD è attivo.

L'istruzione ERRRD viene eseguita sul fronte di salita (OFF-ON) di M1.

Quando l'istruzione ERRRD termina normalmente, il registro dati D101 viene azzerato. In D101 viene memorizzato il codice di errore dell'istruzione ERRRD.

Se si verifica un errore durante l'esecuzione dell'istruzione ERRRD, il codice di errore viene spostato da (s1)+1 (registro D1) a D100.

Il segnale di attivazione dell'istruzione ERRRD non è attivo.

Schema a contatti (GX IEC Developer)

Letture codice di errore per la connessione 1 (1_H viene inserito in (s1)+2)

Impostazione della funzione: Lettura ultimo codice di errore (0_H viene inserito in (s1)+3)

Il segnale di attivazione dell'istruzione ERRRD è attivo.

L'istruzione ERRRD viene eseguita sul fronte di salita (OFF-ON) di M1.

Quando l'istruzione ERRRD termina normalmente, il registro dati D101 viene azzerato. In D101 viene memorizzato il codice di errore dell'istruzione ERRRD.

Se si verifica un errore durante l'esecuzione dell'istruzione ERRRD, il codice di errore viene spostato da (s1)+1 (registro D1) a D100.

Il segnale di attivazione dell'istruzione ERRRD non è attivo.

Lista istruzioni IEC

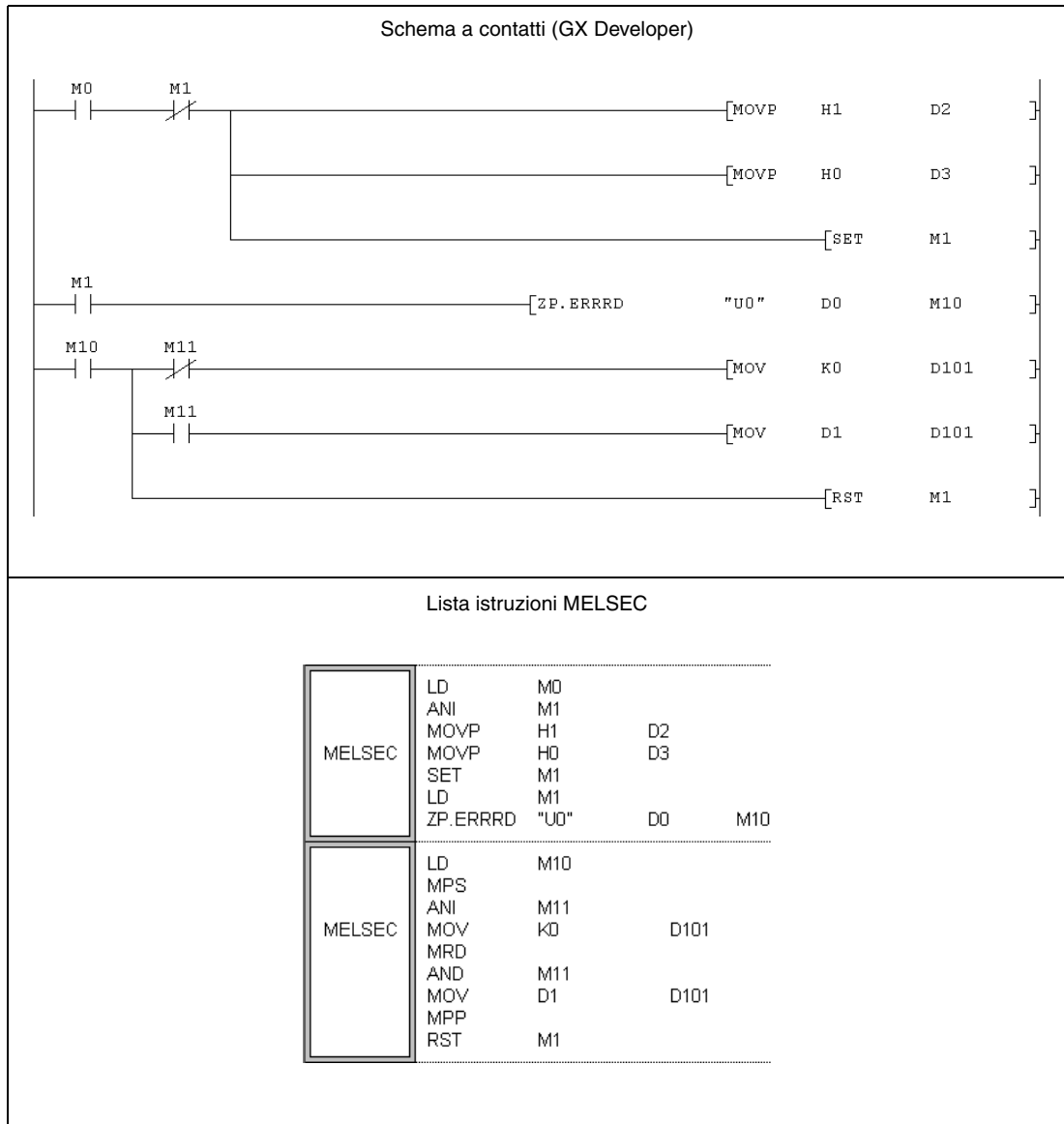
LD	M0		
ANDN	M1		
MOV_P_M	H1,	D2	
MOV_P_M	H0,	D3	
SET_M	M1		
LD	M1		
ERRRD_M	"U0",	var_DO,	var_M10
LD	M10		
ANDN	M11		
MOV_M	K0,	D101	
LD	M10		
AND	M11		
MOV_M	D1,	D101	
LD	M10		
RST_M	M1		

Per una spiegazione su operandi e istruzioni usate, fare riferimento allo schema precedente.

NOTA

Negli editor IEC è necessario definire le variabili nello header della unità organizzativa di programma (POU) La mancata definizione delle variabili provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

- Lista istruzioni MELSEC e schema a contatti di GX Developer
Per una spiegazione degli operandi e istruzioni utilizzate, vedere il programma di esempio per lo schema a contatti di GX IEC Developer nella pagina precedente.



11.3.8 UINI

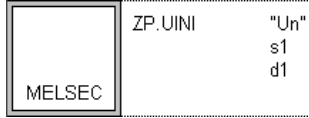
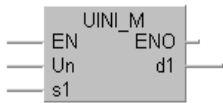
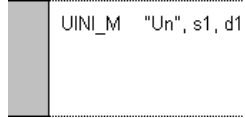
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					●

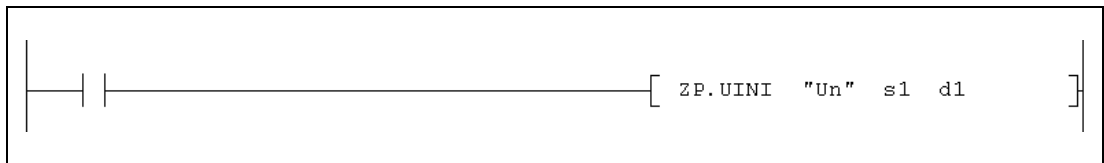
Operandi MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	—	SM0	
d1	●	●	●	—	—	—	—	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
--	---	---

GX Developer



Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati	
„Un“	Indirizzo iniziale di I/O del modulo di interfaccia ETHERNET (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come „U10“)	da 0 a FE _H	Utente	BIN 16-bit	
s1	Indirizzo iniziale dell'area operandi in cui sono contenuti i dati di controllo dell'istruzione.				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(s1)+0	Area di sistema	Usata dal sistema	—	Sistema
	(s1)+1	Risultato dell'esecuzione dell'istruzione	Indica se si è verificato un errore durante l'esecuzione dell'istruzione: 0000 _H : Nessun errore Qualsiasi valore diverso da 0000 _H : Si è verificato un errore. Il valore memorizzato è un codice di errore, spiegato sia nel manuale utente del modulo interfaccia ETHERNET che nel capitolo 13 di questo manuale.		
	(s1)+2	Oggetto della modifica	I bit 0 e 1 di questo operando a word vengono usati per specificare i parametri da modificare: <ul style="list-style-type: none"> • Bit 0: Modifica indirizzo IP della stazione locale (il nuovo indirizzo viene inserito in (s1)+3 e (s1)+4). 0: L'indirizzo IP non cambia 1: Modifica indirizzo IP • Bit 1: Modifica impostazioni operazione (inserire le nuove impostazioni in (s1)+5). 0: Impostazioni invariate 1: Impostazioni modificate Accertarsi di impostare tutti gli altri bit (da b2 a b15) a „0“.	0000 _H a 0003 _H	BIN 16-bit
	(s1)+3 (s1)+4	Indirizzo IP della stazione locale	Nuovo indirizzo IP della stazione locale	0000001 _H a FFFFFFFE _H	
(s1)+5	Impostazioni operazione	I bit di questo operando specificano il modo di funzionamento dell'operazione: <ul style="list-style-type: none"> • Bit 1: Impostazione codice dati della comunicazione 0: Comunicazione in codice binario 1: Comunicazione in codice ASCII • Bit 5: Impostazione trama di invio 0: Trama ETHERNET 1: Trama IEEE802.3 • Bit 6: Abilitazione/disabilitazione scrittura del programma quando la CPU è in modo RUN. 0: Scrittura disabilitata 1: Scrittura abilitata • Bit 8: Impostazione tempo iniziale 0: Non attendere durante l'apertura (con la CPU in STOP non sono possibili comunicazioni). 1: Attesa durante l'apertura (la comunicazione è possibile se la CPU è in STOP). Tutti gli altri bit dell'operando devono essere a „0“.	—	Utente	

Variabili

Dati impostati	Significato			Campo	Contenuto inserito da	Tipo dati
d1	Operando a bit attivato per una scansione al termine dell'istruzione UINI. (d1)+1 indica se si è verificato un errore durante l'esecuzione dell'istruzione.					Bit
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da	
	(d1)+0	Istruzione completata	Indica lo stato di completamento dell'istruzione UINI ON: Istruzione completata OFF: Istruzione non completata	—	Sistema	
(d1)+1	Istruzione completata con errore	Indica l'occorrenza di un errore durante l'elaborazione dell'istruzione UINI ON: Completamento anomalo OFF: Completamento normale	—			

NOTA

Durante il processo di reinizializzazione del solo modulo ETHERNET, cioè senza modificare l'indirizzo IP della stazione locale e le impostazioni operative, i dati di controllo devono essere definiti in modo da inserire il valore (0_H) in (s1)+2, la specifica dell'oggetto della modifica, prima di eseguire l'istruzione UINI.

Il modulo ETHERNET cancella le informazioni sull'indirizzo del dispositivo esterno che stava mantenendo ed effettua il processo di reinizializzazione per consentire il riavviamento della comunicazione dati. (il segnale di completamento iniziale normale (X19) è attivo).

Funzioni

Esecuzione reinizializzazione di un modulo interfaccia ETHERNET

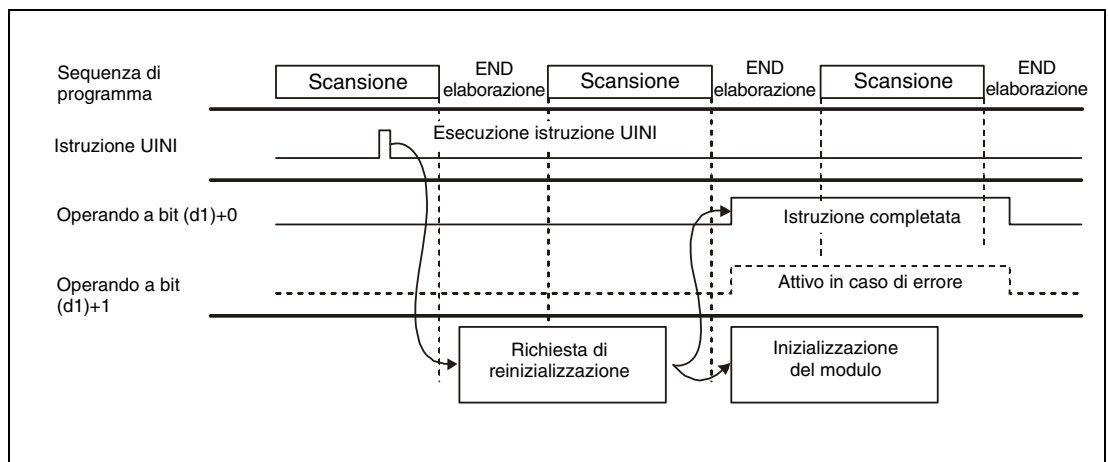
UINI Avvio reinizializzazione

L'istruzione UINI esegue il processo di reinizializzazione del modulo ETHERNET specificato con Un.

Lo stato di esecuzione dell'istruzione UINI può essere controllato con gli operandi (d1)+0 e (d1)+1:

- L'operando a bit (d1)+0 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione UINI, e si disattiva con l'elaborazione di END della scansione successiva.
- L'operando a bit (d1)+1 segnala la presenza di un errore durante l'esecuzione di una istruzione UINI. Se l'istruzione termina normalmente, l'operando rimane OFF. Se si verifica un errore durante l'esecuzione dell'istruzione UINI, (d1)+1 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione UINI, e si disattiva con l'elaborazione di END della scansione successiva.

La figura seguente mostra la temporizzazione durante l'esecuzione di una istruzione UINI:



NOTE

Nel procedere alla reinizializzazione di un modulo ETHERNET, considerare i punti che seguono. (La mancata osservanza provoca errori nella comunicazione dati con dispositivi esterni).

- *Accertarsi di terminare tutte le comunicazioni dati attive su dispositivi esterni e chiudere tutte le connessioni prima di eseguire il processo di reinizializzazione.*
- *Non mischiare un processo di reinizializzazione eseguito scrivendo direttamente nel buffer di memoria, ad esempio usando una istruzione TO instruction, con in processo di reinizializzazione tramite l'istruzione UINI. Inoltre, non richiamare un altro processo di reinizializzazione mentre una istruzione UNI sta già eseguendo.*
- *Accertarsi di resettare i dispositivi esterni se l'indirizzo IP del modulo ETHERNET è stato modificato. (se un dispositivo esterno mantiene l'indirizzo ETHERNET di un altro dispositivo con cui sta comunicando, la comunicazione non può proseguire dopo la modifica dell'indirizzo IP del modulo ETHERNET).*

Errori di esecuzione

Se si verifica un errore durante l'esecuzione dell'istruzione UINI, l'operando a bit (d1)+1 si attiva e in (s1)+1 viene scritto un codice di errore. Per ulteriori informazioni sui codici di errore, fare riferimento ai manuali seguenti:

- Con codice di errore 4FFF_H o inferiore, ulteriori informazioni sono disponibili nel capitolo 13 di questo manuale.
- Con codici di errore pari a C001_H o superiori, fare riferimento al manuale utente del modulo di interfaccia ETHERNET.

Esempio di programma

UINI

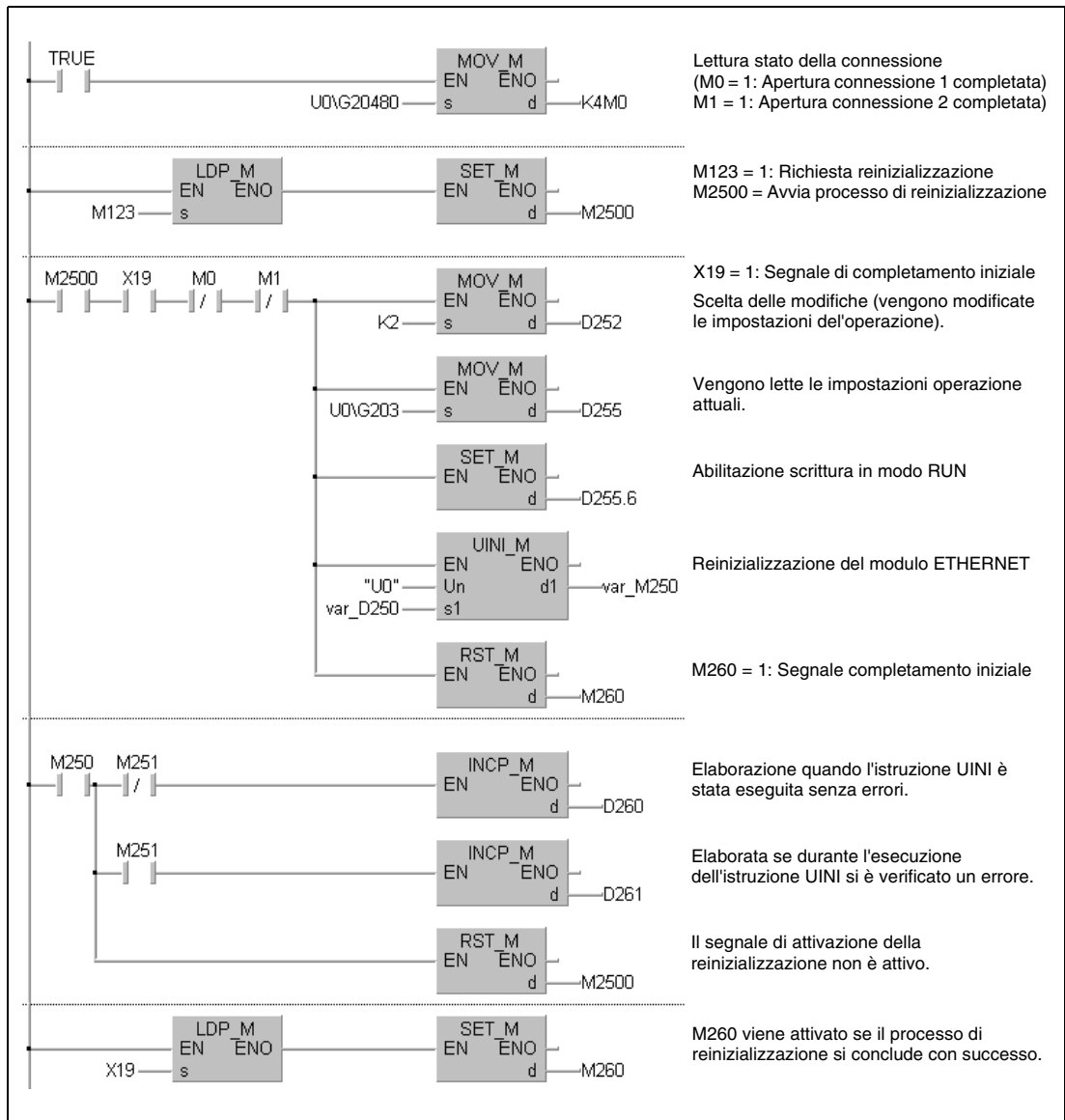
Viene eseguito un processo di reinizializzazione sul modulo ETHERNET con indirizzo iniziale di I/O X/Y0 (campo da X/Y0 a X/Y1F).

NOTE

In questo programma di esempio vengono usate solo le connessioni 1 e 2. Se si utilizzano connessioni diverse, si devono usare i segnali corrispondenti.

Negli editor IEC è necessario definire le variabili nello header della unità organizzativa di programma (POU) La mancata definizione delle variabili provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

● Schema a contatti (GX IEC Developer)



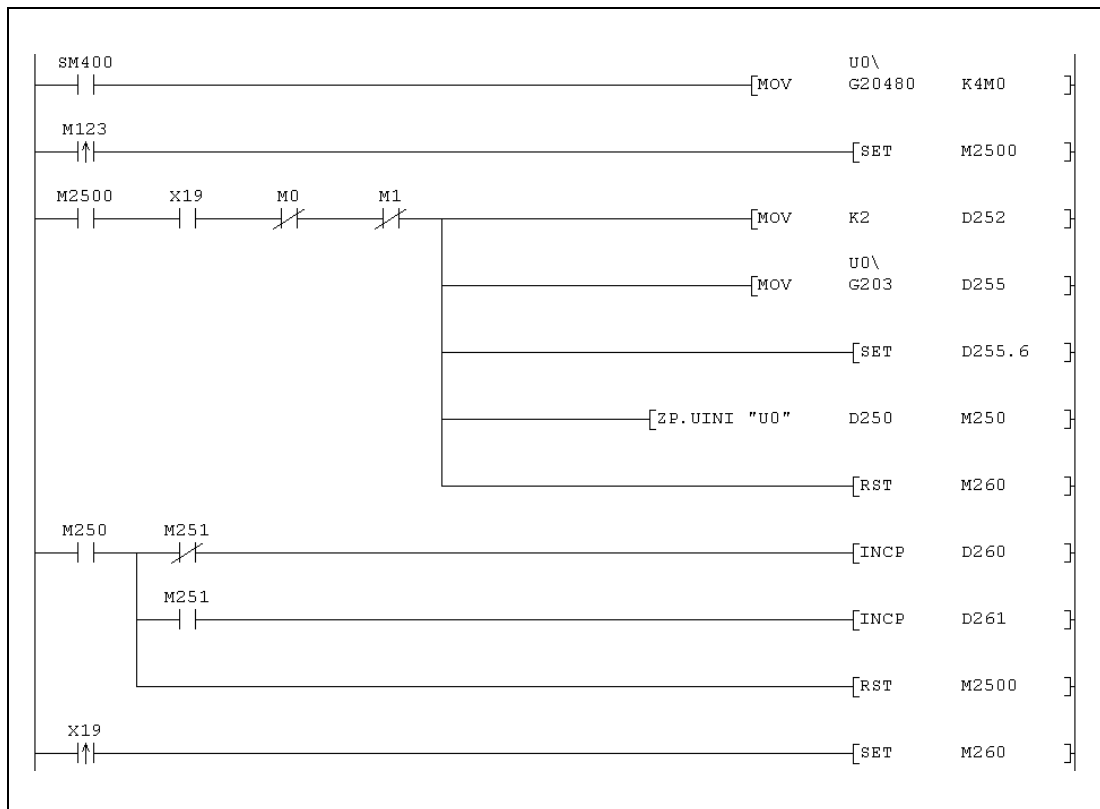
- Lista istruzioni IEC e lista istruzioni MELSEC

Per una spiegazione degli operandi e istruzioni utilizzate, vedere il programma di esempio per lo schema a contatti di GX IEC Developer nella pagina precedente.

Lista istruzioni IEC				Lista istruzioni MELSEC			
LD	TRUE			MELSEC	LD	SM400	
MOV_M	U0\G20480, K4M0				MOV	U0\G20480	K4M0
PLS_M	M123				LDP	M123	
SET_M	M2500				SET	M2500	
AND	M2500				LD	M2500	
AND	X19				AND	X19	
ANDN	M0				ANI	M0	
ANDN	M1				ANI	M1	
MOV_M	K2, D252				MOV	K2	D252
MOV_M	U0\G203, D255				MOV	U0\G203	D255
SET_M	D255.6				SET	D255.6	
UINI_M	"U0", var_D250, var_M250				ZP.UINI	"U0"	D250 M250
RST_M	M260				RST	M260	
LD	M250			MELSEC	LD	M250	
ANDN	M251				MPS		
INCP_M	D260				ANI	M251	
LD	M250				INCP	D260	
AND	M251				MRD		
INCP_M	D261				AND	M251	
LD	M250				INCP	D261	
RST_M	M2500				MPP		
PLS_M	X19				RST	M2500	
SET_M	M260				LDP	X19	
					SET	M260	

- Schema a contatti (GX Developer)

Per una spiegazione degli operandi e istruzioni utilizzate, vedere il programma di esempio per lo schema a contatti di GX IEC Developer nella pagina precedente.



11.4 Istruzioni per MELSECNET/10

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Impostazione stazioni per reti duplex	J.PAIRSET	PAIRSET_M

11.4.1 PAIRSET

CPU

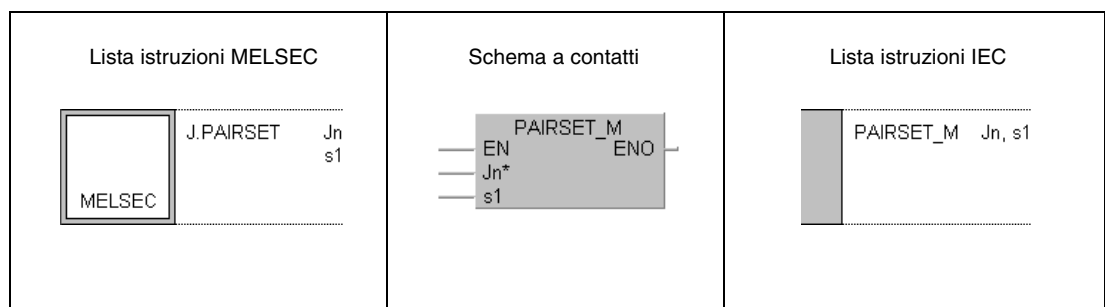
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				● ¹	

¹ Solo per Q4AR

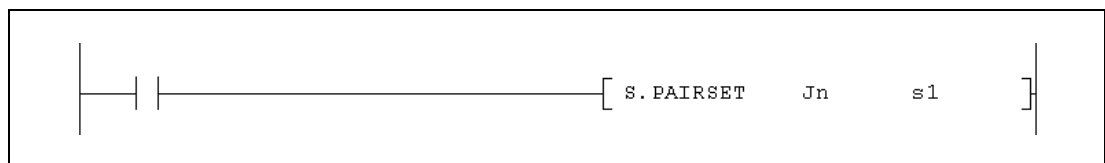
Operandi MELSEC Q

	Operandi utilizzabili								Errore	Numero di passi	
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□G□	Registro indice Zn	Costanti K, H (16#)			Altro
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	—	SM0	

GX IEC Developer



GX Developer



Variabili

Dati impostati	Significato	Tipo dati
Jn	Numero della rete (da 1 a 239)	BIN 16-bit
s1	Indirizzo iniziale dell'area operandi contenente le impostazioni di accoppiamento. Si possono usare file registri (R, ZR) o gli operandi T, ST, C, D e W della zona retentiva. Se si usano file registri, è necessaria una memory card.	

Funzioni **Impostazione accoppiamento di stazioni**

PAIRSET Istruzione per impostazione accoppiamento

Questa istruzione specifica quali numeri di stazione vengono accoppiati (duplex). Necessaria per la configurazione della stazione attiva di un sistema ridondante.

Struttura dell'area operandi che contiene le impostazioni

- L'impostazione delle stazioni negli operandi indicati da s1 non può essere fatta dal programma di sequenza. È necessario caricarla precedentemente nella CPU del PLC tramite un dispositivo periferico.
- Vengono utilizzate quattro word, indipendentemente dal numero di stazioni connesse.
- È possibile accoppiare solo stazioni con numero di stazione adiacente. Per l'accoppiamento, attivare in s1 il bit corrispondente alla stazione con numero più alto.
- Ciascun bit contenuto negli operandi da (s1)+0 a (s1)+3 corrisponde a un numero di stazione fra 1 e 64:

Dati impostati	Bit															
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
(s1)+0	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
(s1)+1	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17
(s1)+2	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33
(s1)+3	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49

NOTE

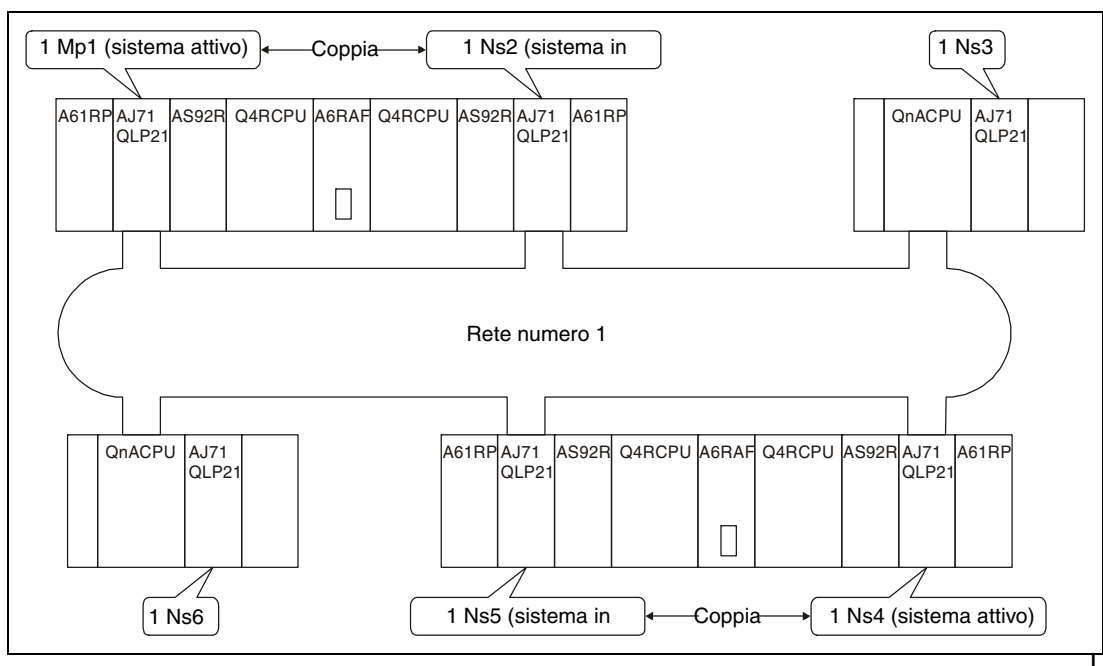
L'istruzione di impostazione accoppiamento è valida solo sulle stazioni attive di sistemi ridondanti. Qualsiasi impostazione eseguita sulle stazioni normali non ha effetto.

Se il modulo di rete del sistema attivo di una configurazione ridondante composta da Q4ARCPU va in avaria per interruzione del cavo di collegamento, la commutazione dal sistema attivo a quello in standby avviene solo se è stata eseguita la parametrizzazione di accoppiamento.

Esempio di programma

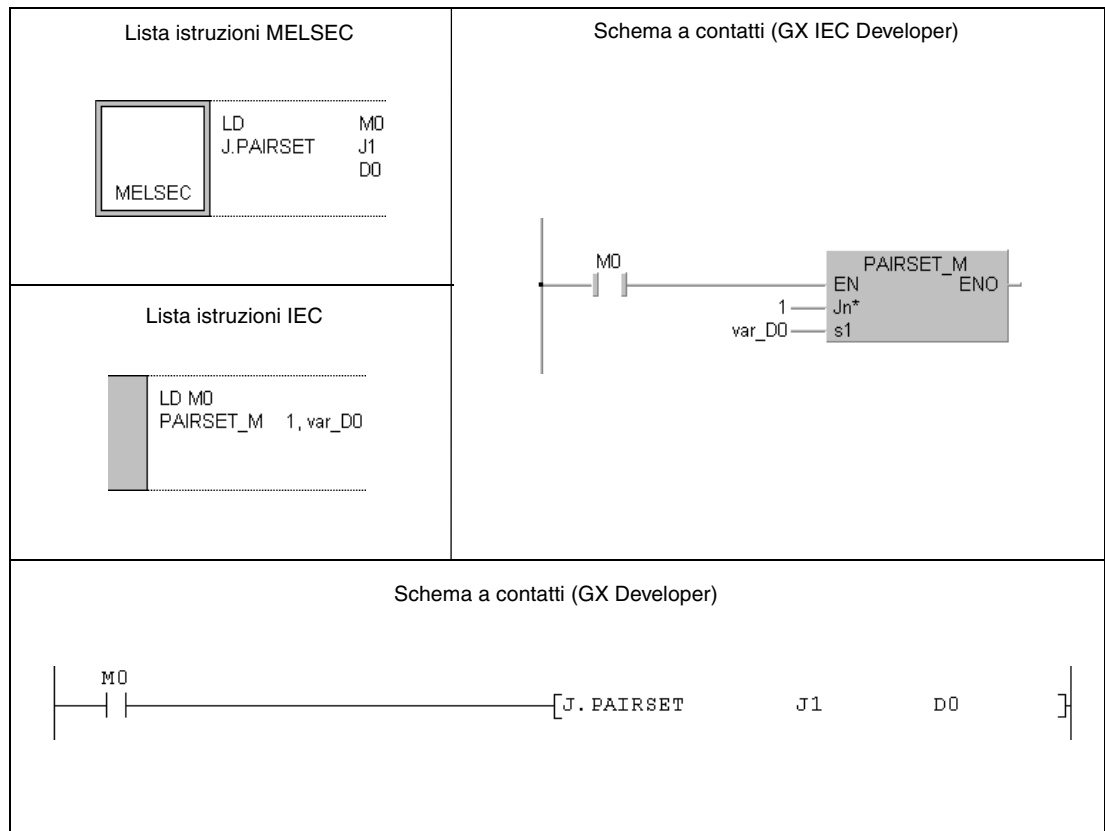
PAIRSET

Viene eseguito l'accoppiamento fra le stazioni 1 e 2 e fra le stazioni 4 e 5 di un sistema ridondante:



e impostazioni sono contenute nei registri dati da D0 a D3. Il bit 1 (b1) di D0 si attiva per l'accoppiamento fra le stazioni 1 e 2, mentre b4 si attiva per l'accoppiamento delle stazioni 4 e 5:

Dati impostati	Bit															
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
D1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

**NOTA**

Negli editor IEC è necessario definire le variabili nello header della unità organizzativa di programma (POU) La mancata definizione delle variabili provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

11.5 Istruzioni per CC-Link

Funzione	Istruzione MELSEC con Editor MELSEC	Istruzione MELSEC con Editor IEC
Configurazione parametri per una rete CC-Link (serie A)	G.RLPA	RLPA_MD
	GP.RLPA	RLPA_P_MD
Configurazione parametri per una rete CC-Link e avviamento del collegamento dati (System Q)	G.RLPASET	RLPASET_MD
	GP.RLPASET	RLPASET_P_MD
Impostazione dei parametri di rinfresco automatico (serie A)	G.RRPA	RRPA_MD
	GP.RRPA	RRPA_P_MD
Lettura dal buffer di memoria di una stazione intelligente o dalla memoria operandi della CPU del PLC	G.RIRD	RIRD_MD
	GP.RIRD	RIRD_P_MD
Scrittura nel buffer di memoria di una stazione intelligente o nella memoria operandi della CPU del PLC	G.RIWT	RIWT_MD
	GP.RIWT	RIWT_P_MD
Lettura dal buffer di memoria di una stazione intelligente (con handshake)	G.RIRCV	RIRCV_MD
	GP.RIRCV	RIRCV_P_MD
Scrittura nel buffer di memoria di una stazione intelligente (con handshake)	G.RISEND	RISEND_MD
	GP.RISEND	RISEND_P_MD
Scrittura nel buffer di memoria ad aggiornamento automatico	G.RITO	RITO_MD
	GP.RITO	RITO_P_MD
Lettura da un buffer di memoria ad aggiornamento automatico	G.RIFR	RIFR_MD
	GP.RIFR	RIFR_P_MD

11.5.1 RLPA (serie A)

CPU

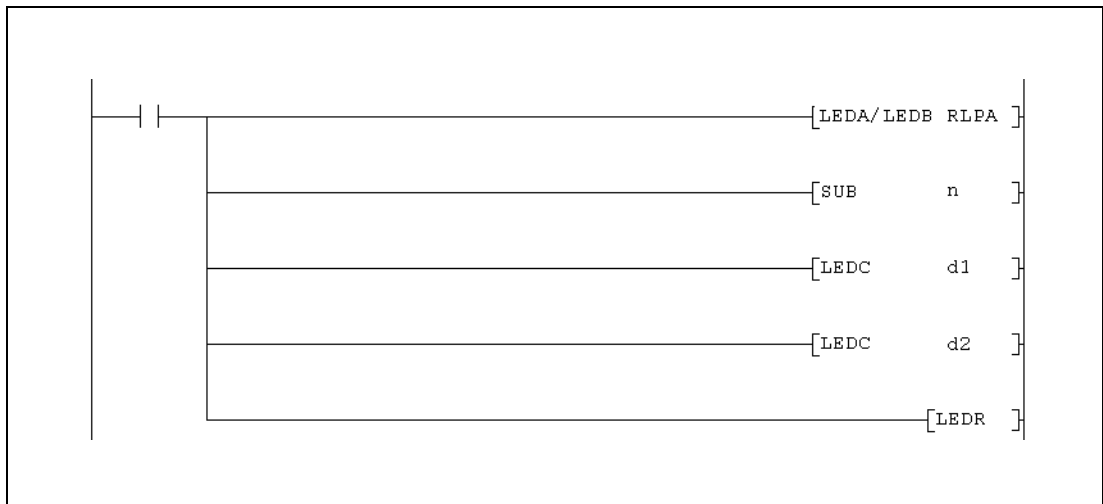
AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●		

Operandi
MELSEC A

	Operandi utilizzabili																Lunghezza blocco	Numero di passi	Indice	Rapporto	Errore			
	Operandi a bit						Operandi a word (16-bit)						Costanti		Puntatore							Livello		
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V						K	H (16#)	P
n1																●	●							
d1							●	●	●	●	●													●
d2	●	●	●	●	●																			

GX IEC
Developer

Lista istruzioni MELSEC	Schema a contatti	Lista istruzioni IEC												
<div style="border: 1px solid black; padding: 5px; width: fit-content;">MELSEC</div> <table style="border-collapse: collapse; margin-top: 10px;"> <tr><td>LEDA/LEDB</td><td>RLPA</td></tr> <tr><td>SUB</td><td>n</td></tr> <tr><td>LEDC</td><td>d1</td></tr> <tr><td>LEDC</td><td>d2</td></tr> <tr><td>LEDR</td><td></td></tr> </table>	LEDA/LEDB	RLPA	SUB	n	LEDC	d1	LEDC	d2	LEDR			<table style="border-collapse: collapse; margin-top: 10px;"> <tr><td>RLPA_MD</td><td>n, d1, d2</td></tr> </table>	RLPA_MD	n, d1, d2
LEDA/LEDB	RLPA													
SUB	n													
LEDC	d1													
LEDC	d2													
LEDR														
RLPA_MD	n, d1, d2													

GX
Developer

Per ulteriori informazioni sulla programmazione delle istruzioni dedicate con gli editor MELSEC, fare riferimento al capitolo 3.3 di questo manuale.

Variabili

Dati impostati	Significato		Campo	Contenuto inserito da	Tipo dati	
n	Indirizzo iniziale di I/O del modulo master CC-Link (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come 10 _H)		da 0 a FE _H	Utente	BIN 16-bit	
d1	Indirizzo iniziale degli operandi che contengono i parametri del collegamento					
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da	
	(d1)+0	Modo sincrono	<ul style="list-style-type: none"> ● 0: Modo sincrono non attivo ● 1: Modo sincrono 	0 o 1	Utente	
	(d1)+1	Numero delle stazioni collegate	Imposta il numero delle stazioni slave collegate al modulo master di CC-Link	da 1 a 64	Utente	
	(d1)+2	Configurazione stazione slave (prima stazione)	Vedi tabella a pagina seguente.			
	(d1)+3	Configurazione stazione slave (seconda stazione)				
	•	•	•	•		
	•	•	•	•		
		Configurazione stazione slave (ultima stazione)	Vedi tabella a pagina seguente.			
		P r i m a	Dimensione buffer di invio	Numero di operandi scambiati fra stazione master e locale o fra stazioni intelligenti.		Funzione del modulo usato
		s t a z.	Dimensione buffer di ricezione			
		L o c a l e	Dimensione buffer ad aggiornamento automatico			
	•	•	•	•		•
•	•	•	•	•		
(d1)+(n-2)	U l t i m a	Dimensione buffer di invio	Come per stazione 1	Funzione del modulo usato		
(d1)+(n-1)		Dimensione buffer di ricezione				
(d1)+n		Dimensione buffer ad aggiornamento automatico				
d2	Operando a bit attivato per una scansione al termine dell'istruzione RLPA.		0 o 1	Sistema	Bit	

Numero punti necessari per d1:

Due punti sono usati per la selezione del modo sincrono in (d1)+0 e l'indicazione del numero di stazioni collegate in (d1)+1. **Un** punto è necessario per ciascuna stazione per la configurazione di stazione. Inoltre, **tre** punti vengono usati per ciascuna stazione locale o intelligente, per impostare le dimensioni dei buffer.

Configurazione stazione slave

Per ciascuna stazione, un operando a word ((d1)+2, (d1)+6, (d1)+10, ...) viene riservato per contenere la configurazione della stazione:

Significato	Descrizione	Campo valori			
Configurazione stazione slave	<div style="text-align: center;"> <p style="font-size: small;">b15 ----- b12 b11 ----- b8 b7 ----- b0</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="width: 40px; text-align: center;">Tipo stazione</td> <td style="width: 40px; text-align: center;">Stazioni occupate</td> <td style="width: 80px; text-align: center;">Numero stazione</td> </tr> </table> </div> <p>0: Stazione I/O remoti 1: Stazione remota 2: Stazione intelligente (compreso stazioni locali e stazione master in standby)</p> <p style="text-align: center;">Numero di stazioni occupate dal modulo CC-Link:</p> <p>1: 1 stazione occupata 2: 2 stazioni occupate 3: 3 stazioni occupate 4: 4 stazioni occupate</p> <p style="text-align: right;">Imposta il numero di stazione nel campo da 1 a 64.</p>	Tipo stazione	Stazioni occupate	Numero stazione	da b0 a b7: 1 – 64 (01 _H – 40 _H) da b8 a b11: 1 – 4 da b12 a b15: 0 – 2
Tipo stazione	Stazioni occupate	Numero stazione			

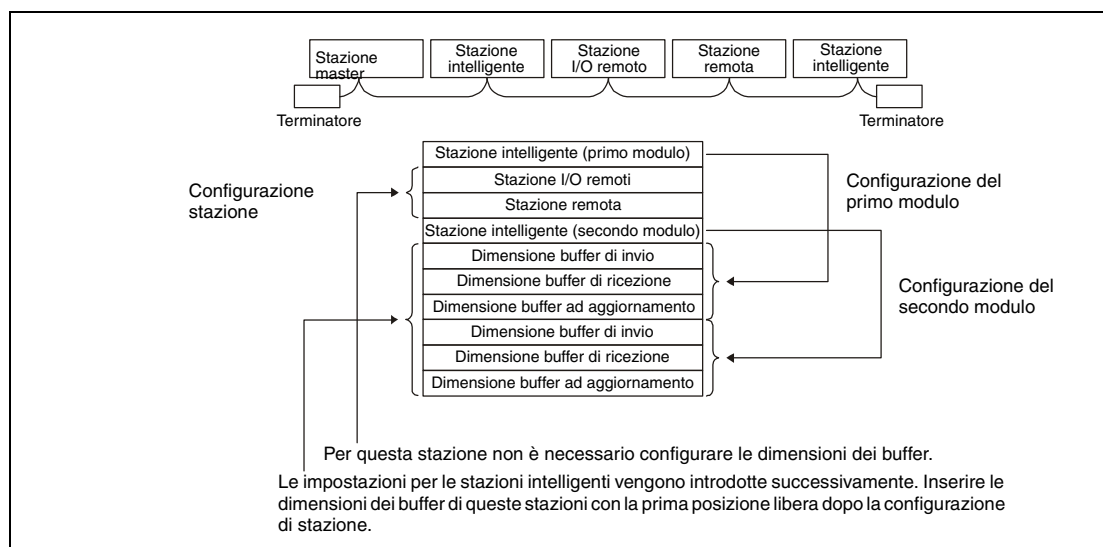
Funzioni Configurazione parametri per una rete CC-Link

RLPA Istruzione di parametrizzazione

Quando viene eseguita l'istruzione RLPA, l'insieme dei dati relativi ai parametri di rete contenuti negli operandi a partire da quello indicato da (d1) viene inviato verso il modulo master specificato da (n).

NOTA Usare l'istruzione RLPA solo per configurare il modo sincrono, il numero di stazioni collegate, le configurazioni di stazione e le dimensioni dei buffer. Per tutti gli altri parametri vengono utilizzati dei valori di default. Se i parametri vengono impostati sia con l'istruzione RLPA che con l'istruzione TO, i parametri impostati con l'istruzione TO vengono ignorati.

Se il tipo di stazione slave specificato è una stazione locale/intelligente, è necessario definire la "dimensione buffer di trasmissione", "dimensione buffer di ricezione" e "dimensione buffer ad aggiornamento automatico". Se il tipo di stazione slave è una stazione di I/O remoto o una stazione remota, questi dati non sono necessari. Un esempio viene mostrato nella figura seguente:



NOTE Per la dimensione dei buffer di invio e ricezione, considerare 7 word oltre alla dimensione effettivamente necessaria per la comunicazione.

Per la dimensione di buffer ad aggiornamento automatico, allocare lo spazio necessario per la singola stazione intelligente.

Nelle stazioni intelligenti, impostare 0 come dimensione del buffer ad aggiornamento automatico nelle stazioni in cui la funzione di aggiornamento automatico non è fornita o non usata.

Se l'istruzione RLPA viene utilizzata un'altra volta in modo RUN per modificare i parametri di rete, i nuovi dati non vengono usati per la comunicazione con la stazione slave. Dopo che la CPU serie A è stata commutata su STOP/PAUSE e poi su RUN, i nuovi parametri di rete verranno utilizzati per la comunicazione con le stazioni slave.

L'esecuzione dell'istruzione RLPA avvia automaticamente il collegamento dati.

Quando viene eseguita l'istruzione RLPA, si deve prevedere un interblocco usando il segnale di errore unità (Xn0) ed il segnale di unità pronta (XnF) che indicano se l'unità CC-Link è pronta.

Condizione di esecuzione

Se viene usata l'istruzione LEDA, l'istruzione RLPA viene eseguita ad ogni scansione in cui il comando di scrittura è ON.

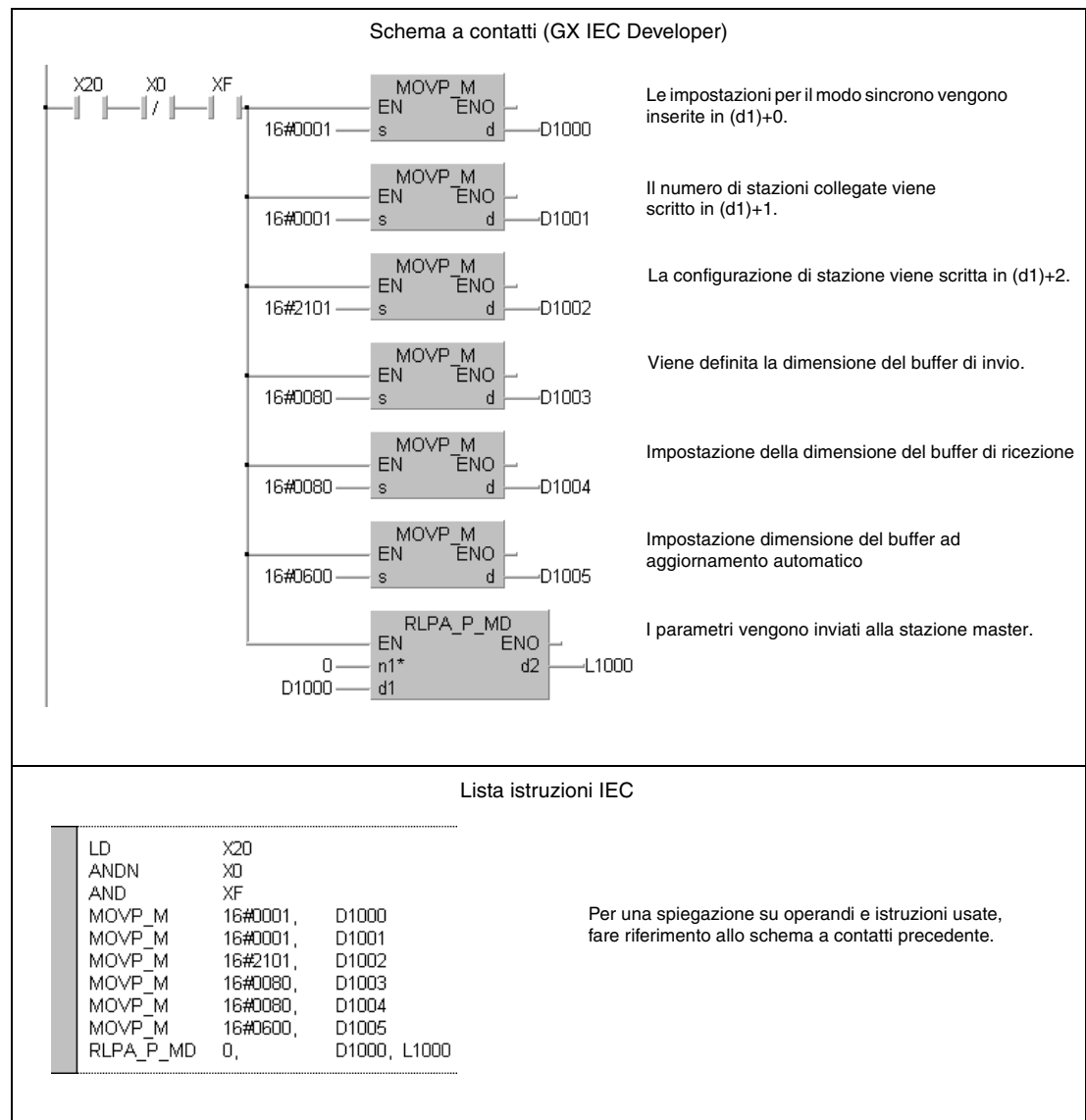
Se viene usata l'istruzione LEDB, l'istruzione RLPA viene eseguita per una scansione sul fronte positivo (OFF -> ON) del comando di scrittura.

Esempio di programma RLPA

Questo programma imposta i seguenti parametri di rete nel modulo master CC-Link con indirizzo iniziale di I/O X/Y000.

Parametro	Impostazione	Valore	Operando per memorizzazione dati
Modo sincrono	Valido	1	D1000
Numero delle stazioni collegate	1 modulo	1	D1001
Configurazione stazione	Tipo stazione slave	Stazione intelligente	D1002
	Numero stazioni occupate	1 stazione	
	Numero stazione	Numero 1	
Dimensione buffer di invio	128 word	80 _H	D1003
Dimensione buffer di ricezione	128 word	80 _H	D1004
Buffer ad aggiornamento automatico	960 word	600 _H	D1005

- Editor IEC (questo programma di esempio è mostrato a pagina seguente con lista istruzioni MELSEC e schema a contatti di GX Developer).



- Lista istruzioni MELSEC e schema a contatti di GX Developer
 Per una spiegazione degli operandi e istruzioni utilizzate, vedere il programma di esempio per lo schema a contatti di GX IEC Developer nella pagina precedente.

Schema a contatti (GX Developer)

Lista istruzioni MELSEC

	LD	X20	
	ANI	X0	
	AND	X0F	
MELSEC	MOV P	H0001	D1000
	MOV P	H0001	D1001
	MOV P	H2101	D1002
	MOV P	H0080	D1003
	MOV P	H0080	D1004
	MOV P	H0600	D1005
	LEDB	RLPA	
	SUB	K0	
	LEDC	D1000	
	LEDC	L1000	
	LEDR		

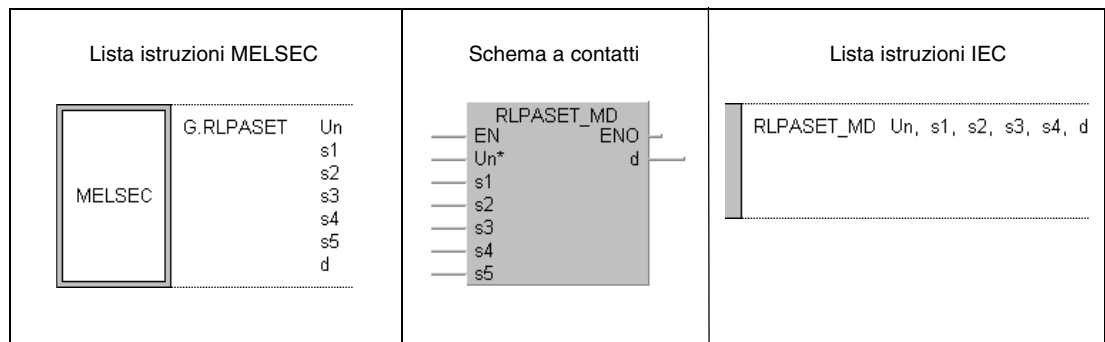
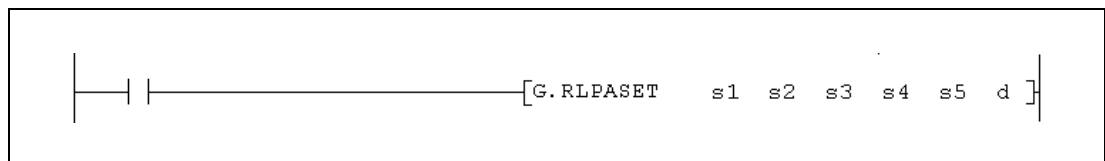
11.5.2 RLPASET(System Q)

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
					●

Operandi
MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	—	SM0	
s2	—	●	●	—	—	—	—	—	—		
s3	—	●	●	—	—	—	—	—	—		
s4	—	●	●	—	—	—	—	—	—		
s5	—	●	●	—	—	—	—	—	—		
d	●	●	●	—	—	—	—	—	—		

GX IEC
DeveloperGX
Developer

Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati		
Un	Indirizzo iniziale di I/O del modulo master CC-Link (vengono inserite solo le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come 10 _H)	da 0 a FE _H	Utente	BIN 16-bit		
s1	Indirizzo iniziale dell'area operandi in cui sono contenuti i dati di controllo dell'istruzione.					
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da	
	(s1)+0	Risultato dell'esecuzione dell'istruzione	Indica se si è verificato un errore durante l'esecuzione dell'istruzione: 0000 _H : Nessun errore Qualsiasi valore diverso da 0000 _H : Codice di errore	—	Sistema	
	(s1)+1	Validazione della configurazione	I primi quattro bit vengono usati per specificare se i dati da s2 a s5 sono validi o meno: Bit 0 = 1: Configurazione stazione slave (s2) Bit 1 = 1: Configurazione stazione riservata (s3) Bit 2 = 1: Ignora stazione in errore (s4) Bit 3 = 1: Configurazione buffer invio, ricezione e ad aggiornamento variabile (s5) Per i dati di configurazione non validi, vengono assunti valori di default.	da 0 a F	Utente	
	(s1)+2	Numero di moduli connessi	Imposta il numero di stazioni slave collegate (comprese le stazioni riservate)	da 1 a 64		Indirizzo
	(s1)+3	Numero tentativi	Imposta il numero di tentativi per la comunicazione con una stazione in avaria.	da 1 a 7		
	(s1)+4	Numero moduli a ritorno automatico	Imposta il numero di moduli slave che vengono reinseriti automaticamente sulla rete dopo un guasto, in una scansione del link.	da 1 a 10		
	(s1)+5	Funzionamento in caso di STOP della CPU del PLC	Specifica lo stato della rete in caso di errore sulla CPU del PLC della stazione master. 0: Stop 1: Continua	0 o 1		
	(s1)+6	Scelta modo scansione	Sceglie fra modalità sincrona e asincrona 0: La rete dati è sincrona con la scansione della sequenza di programma 1: La rete dati è asincrona con la scansione della sequenza di programma	0 o 1		
(s1)+7	Impostazione tempo di ritardo	Intervallo di scansione del link (unità: 50 µs)	da 0 a 100			

Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati	
s2	Indirizzo iniziale dell'area operandi contenente la configurazione della stazione slave.			Indirizzo	
	Dati impostati	Significato	Descrizione		Contenuto inserito da
	(s2)+0	Configurazione Stazione N.1	Vedi tabella a pagina 93 Inserire le configurazioni per tutti i moduli specificati in (s1)+2 come numero di moduli collegati.		Utente
	(s2)+1	Configurazione Stazione N.2			
	•	•			
	•	•			
(s2)+62	Configurazione Stazione N.63				
(s2)+63	Configurazione Stazione N.64				
s3	Indirizzo iniziale dell'area operandi contenente la configurazione delle stazioni riservate. Eseguire la configurazione di tutte le stazioni fino al numero massimo impostato in s2.			Indirizzo	
	Dati impostati	Significato	Descrizione		Contenuto inserito da
	(s3)+0	Configurazione stazioni 1 – 16	Specificare una stazione riservata, impostando a 1 il bit corrispondente al numero di stazione (vedi tabella a pagina 93). Specificare solo il numero della prima stazione in un modulo che occupa due o più stazioni. I parametri di default non definiscono nessuna stazione come riservata.		Utente
	(s3)+1	Configurazione stazioni 17 – 32			
	(s3)+2	Configurazione stazioni 33 – 48			
(s3)+3	Configurazione stazioni 49 – 64				
s4	Indirizzo iniziale dell'area operandi contenente la configurazione delle stazioni in avaria da ignorare. Eseguire la configurazione di tutte le stazioni fino al numero massimo impostato in s2.			Indirizzo	
	Dati impostati	Significato	Descrizione		Contenuto inserito da
	(s4)+0	Configurazione stazioni 1 – 16	Se si deve ignorare l'avaria di una stazione, attivare il bit corrispondente al numero di stazione (vedi tabella a pagina 93). Nei moduli che occupano due o più stazioni, impostare solo il primo numero di stazione.		Utente
	(s4)+1	Configurazione stazioni 17 – 32			
	(s4)+2	Configurazione stazioni 33 – 48	Se una stazione viene configurata per ignorarne l'avaria e contemporaneamente come riservata, l'impostazione di stazione riservata ha precedenza.		
(s4)+3	Configurazione stazioni 49 – 64	I parametri di default non ignorano gli errori di nessuna stazione.			

Variabili

Dati impostati	Significato		Campo	Contenuto inserito da	Tipo dati	
s5	Indirizzo iniziale dell'area operandi contenente le impostazioni sulle dimensioni dei buffer di memoria. Eseguire la configurazione per le stazioni definite in s2 come locali o stazioni intelligenti. Iniziare con la stazione di numero inferiore.					
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da	
	(s5)+0	P r i m o d u l o	Dimensione buffer di invio	Specificare la dimensione del buffer necessario per la comunicazione fra la stazione master e le stazioni locali o intelligenti. La dimensione massima complessiva dei buffer di invio e ricezione è 4096 word (1000 _H). Per la dimensione dei buffer di invio e ricezione, considerare 7 word oltre alla dimensione effettivamente necessaria per la comunicazione.	0 _H : Nessun buffer 40 _H - 1000 _H (da 64 a 4096 word) Impostazione di default: 40 _H	Utente
	(s5)+1		Dimensione buffer di ricezione	Numero di punti del buffer ad aggiornamento automatico usato per la comunicazione fra stazione master e stazioni locali o stazioni intelligenti. La dimensione del buffer ad aggiornamento automatico deve essere uguale o maggiore della dimensione necessaria per le singole stazioni intelligenti.	0 _H : Nessun buffer 40 _H - 1000 _H (da 64 a 4096 word) Impostazione di default: 40 _H	
	(s5)+2		Dimensione buffer a rinfresco automatico	Numero di punti del buffer ad aggiornamento automatico usato per la comunicazione fra stazione master e stazioni locali o stazioni intelligenti. La dimensione del buffer ad aggiornamento automatico deve essere uguale o maggiore della dimensione necessaria per le singole stazioni intelligenti.	0 _H : Nessun buffer 80 _H - 1000 _H (da 128 a 4096 word) Impostazione di default: 80 _H	
	•	•	•	•	•	
	(s5)+75	M o d u l o	Dimensione buffer di invio	Come per modulo 1.		
	(s5)+76		Dimensione buffer di ricezione			
(s5)+77	Dimensione buffer a rinfresco automatico					
d	Operando a bit attivato per una scansione al termine dell'istruzione RLPASET. (d)+1 indica se si è verificato un errore durante l'esecuzione dell'istruzione.					
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da	
	(d)+0:	Istruzione completata	Indica lo stato di completamento dell'istruzione RLPASET ON: Istruzione completata OFF: Istruzione non completata	0 o 1	Sistema	
(d)+1:	Istruzione completata con errore	Indica l'occorrenza di un errore durante l'elaborazione dell'istruzione RLPASET ON: Completamento anomalo OFF: Completamento normale	0 o 1			

Configurazione stazione slave

Per ciascuna stazione, un operando a word (da (s2)+0 a (s2)+63) viene riservato per contenere la configurazione della stazione:

Significato	Descrizione	Campo
Impostazioni per i moduli da 1 a 64	<div style="text-align: center;"> </div> <p>0: Stazione I/O remoti 1: Stazione remota 2: Stazione intelligente (compreso stazioni locali e stazione master in standby)</p> <p>Imposta il numero di stazione nel campo da 1 a 64.</p> <p>Numero di stazioni occupate dal modulo CC-Link: 1: 1 stazione occupata 2: 2 stazioni occupate 3: 3 stazioni occupate 4: 4 stazioni occupate</p>	da b0 a b7: 1 – 64 (01 _H – 40 _H) da b8 a b11: 1 – 4 da b12 a b15: 0 – 2

Le impostazioni di default da (s2)+0 a (s2)+63 sono da „0101_H“ a „0140_H“. (numero di stazione da 1 a 64, una stazione occupata, stazione di I/O remoto).

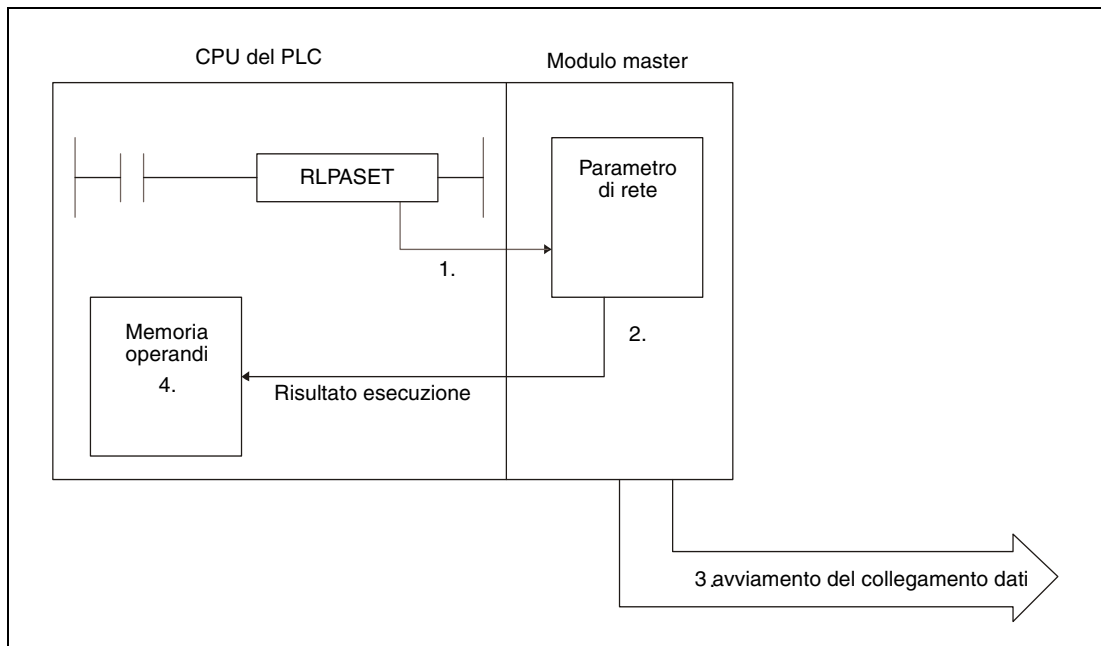
Indicazione del numero stazione in s3 e s4

Ciascun bit dei quattro operandi a word usati per s3 e s4 rappresenta una stazione:

Dati impostati	Bit															
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
(s3)+0 (s4)+0	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
(s3)+1 (s4)+1	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17
(s3)+2 (s4)+2	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33
(s3)+3 (s4)+3	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49

I numeri da 1 a 64 nella tabella indicano i numeri di stazione. Un bit a 1 indica che la stazione corrispondente è selezionata.

Funzioni **Configurazione parametri per una rete CC-Link e avviamento del collegamento dati**
RLPASET **Istruzione di parametrizzazione**



1. I parametri di rete contenuti da (s1) a (s5) vengono inviati al modulo master di CC-Link indicato da Un, per mezzo dell'istruzione RLPASET.
 2. I parametri ricevuti vengono controllati dal modulo master.
 3. Se i dati sono corretti, la comunicazione viene avviata.
 4. L'operando indicato da (d) viene attivato.
- È possibile attivare una sola istruzione RLPASET alla volta.

Numero operandi richiesti

Per l'istruzione RLPASET sono necessari i seguenti operandi:

- s1: 8 operandi a word
- s2: 64 operandi a word
- s3: 4 operandi a word
- s4: 4 operandi a word
- s5: 78 operandi a word

Considerare i requisiti di memoria da (s1) a (s5) durante la programmazione.

Un esempio:

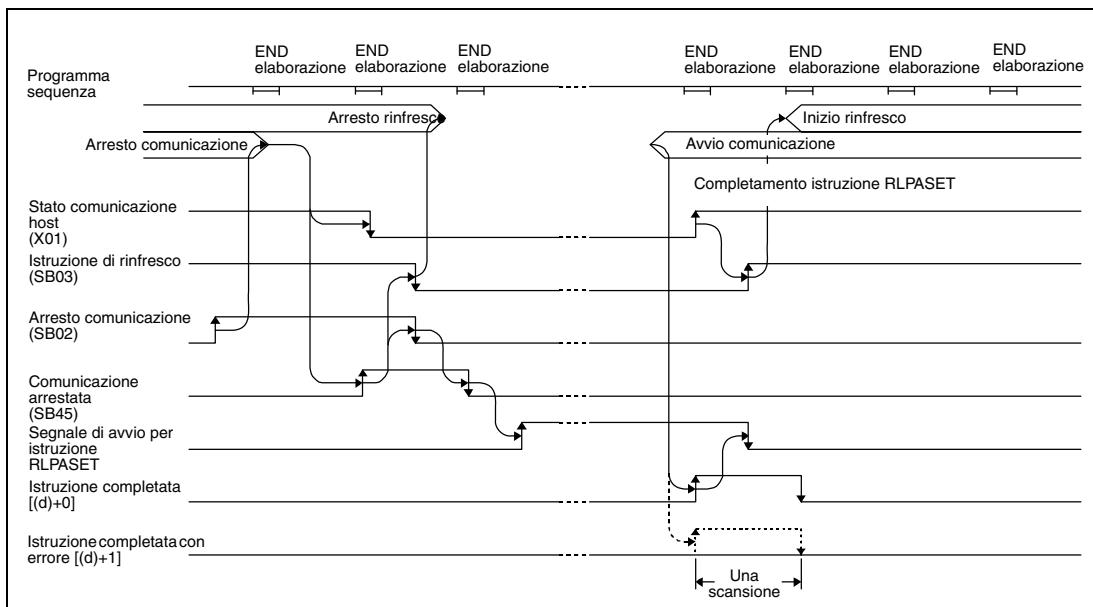
Quattro stazioni slave sono collegate ad un modulo master. Nella Q02CPU inserita nel PLC della stazione master sono disponibili i registri di comunicazione da D0 a D12287. Se D12284 viene specificato come primo operando per (s2) perché esistono solo quattro stazioni slave, l'esecuzione dell'istruzione RLPASET termina con il codice di errore 4101. Questo è dovuto al fatto che la CPU del PLC controlla sempre l'area necessaria per 64 stazioni (da D12284 a D12347 in questo esempio) e in questo caso il campo operando viene superato.

Lo stato di esecuzione dell'istruzione RLPASET può essere controllato con gli operandi (d1)+0 e (d1)+1:

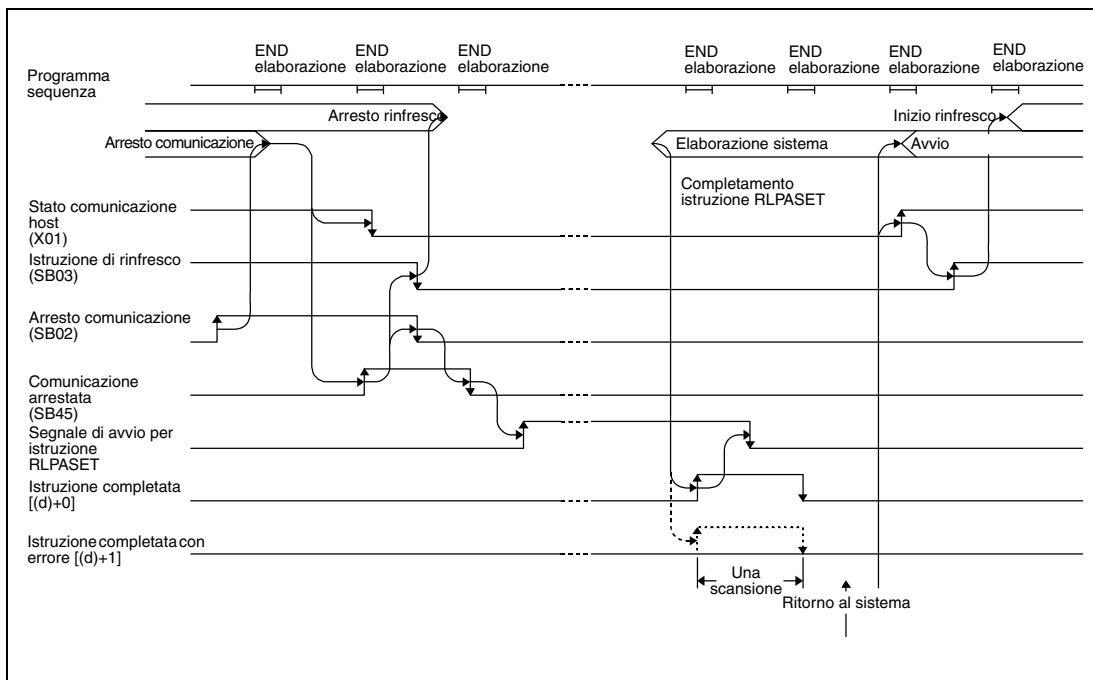
- L'operando a bit (d1)+0 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione RLPASET, e si disattiva con l'elaborazione di END della scansione successiva.

- L'operando a bit (d1)+1 segnala la presenza di un errore durante l'esecuzione di una istruzione RLPASET. Se l'istruzione termina normalmente, l'operando rimane OFF. Se si verifica un errore durante l'esecuzione dell'istruzione RLPASET, (d1)+1 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione RLPASET, e si disattiva con l'elaborazione di END della scansione successiva.

La figura seguente mostra la temporizzazione durante l'esecuzione dell'istruzione RLPASET con tutte stazioni normali:



Di seguito viene mostrata la temporizzazione dell'istruzione RLPASET in caso di stazione in avaria:



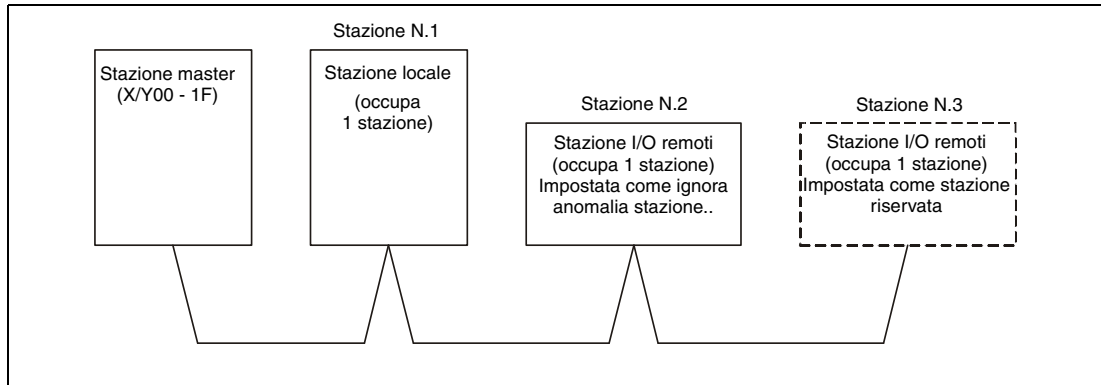
Errori di esecuzione

Nei casi seguenti, si verifica un errore di funzionamento, viene attivato il flag di errore SM0, ed un codice di errore viene memorizzato in SD0:

- Se il modulo indicato da (Un) non è un modulo funzione speciale (codice di errore: 2112).
- Se si cerca di eseguire una istruzione non supportata (codice di errore: 4002).
- Se il numero di operandi dell'istruzione non è corretto. (codice di errore: 4003)
- Se l'istruzione specifica un operando che non si può usare. (codice di errore: 4004)
- Se l'istruzione contiene dati che non si possono usare. (codice di errore: 4100)
- Se il numero di punti per i dati usati dall'istruzione supera il campo ammesso, oppure l'area di memoria e le costanti di un operando usato dall'istruzione superano il campo ammesso (compresi gli operandi dummy). (codice di errore: 4101)

Esempio di programma RLPASET

Questo programma trasferisce i parametri di rete alla stazione master con indirizzo iniziale di I/O X/Y000. La rete CC-Link è composta da tre stazioni:

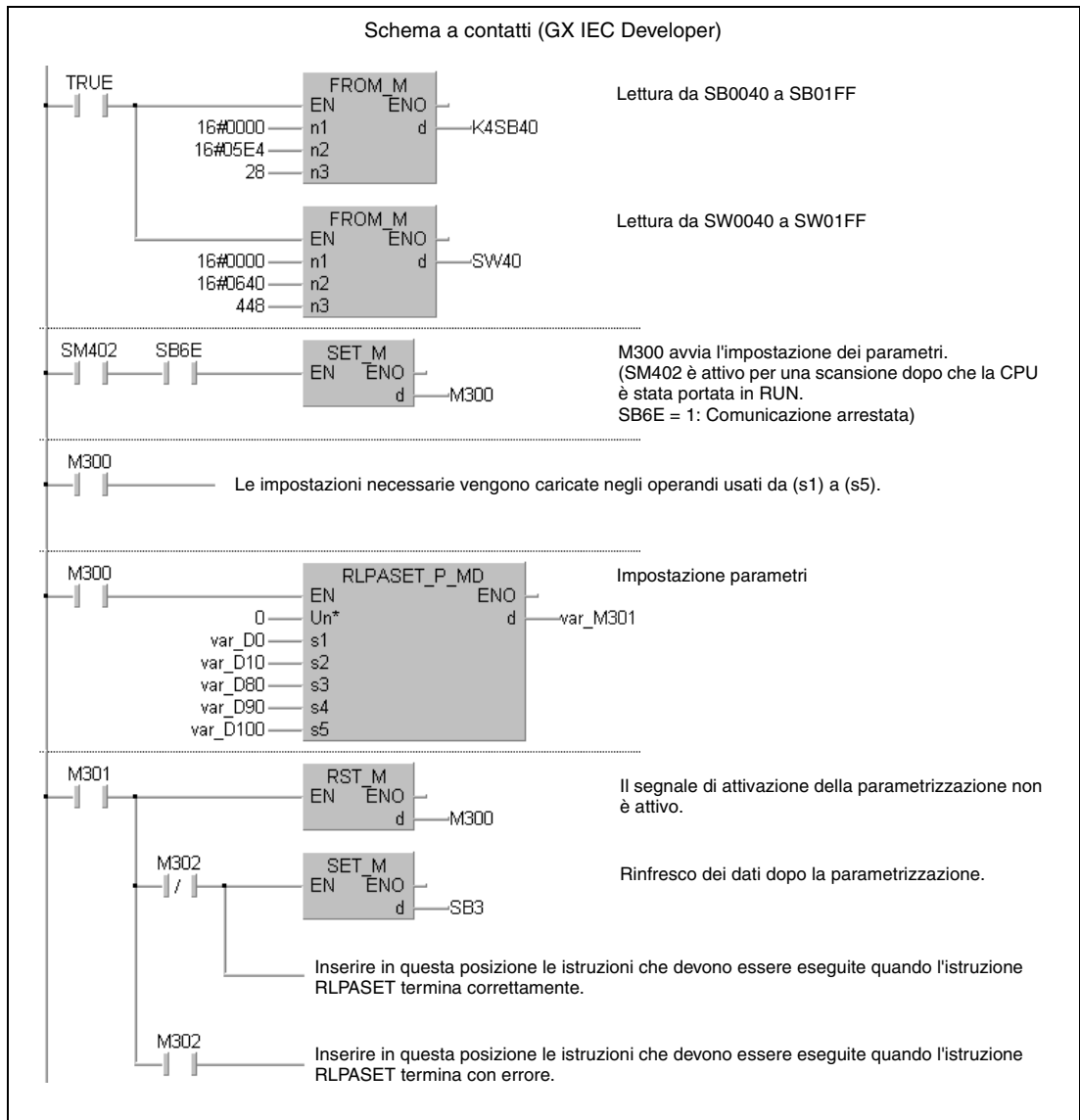


Gli operandi indicati da (s1) a (s5) contengono i valori seguenti:

Parametro		Impostazione	Impostazione valore	Operando allocato	
Dati di controllo per l'esecuzione dell'istruzione	(s1)+1	Validazione della configurazione	Tutte le impostazioni sono valide.	15	D1
	(s1)+2	Numero di moduli connessi	3 moduli slave	3	D2
	(s1)+3	Numero tentativi	3 volte	3	D3
	(s1)+4	Numero moduli a ritorno automatico	1 modulo	1	D4
	(s1)+5	Funzionamento in caso di STOP della CPU del PLC	Stop	0	D5
	(s1)+6	Scelta modo scansione	Asincrono	0	D6
	(s1)+7	Impostazione ritardo	0 μ s	0	D7
Configurazione stazioni slave	(s2)+0	Configurazione prima stazione	Stazione locale, occupa 1 stazione, Stazione N.1	2101 _H	D10
	(s2)+1	Configurazione seconda stazione	Stazione I/O remoti, occupa 1 stazione, Stazione N.2	102 _H	D11
	(s2)+2	Configurazione terza stazione	Stazione I/O remoti, occupa 1 stazione, Stazione N.3	103 _H	D12
Stazioni riservate	(s3)+0	Scelta delle stazioni riservate	La stazione 3 è riservata (bit 2 = 1)	4	D80
	(s3)+1			0	D81
	(s3)+2			0	D82
	(s3)+3			0	D83
Ignora errore stazione in avaria	(s4)+0	Indica le stazioni su cui si devono ignorare gli errori di avaria	Stazione N.2 (bit 1 attivo)	2	D90
	(s4)+1			0	D91
	(s4)+2			0	D92
	(s4)+3			0	D93
Dimensioni buffer	(s5)+0	Buffer di invio della prima stazione locale (stazione 1)	100 word	64 _H	D100
	(s5)+1	Buffer di ricezione della prima stazione locale (stazione 1)	100 word	64 _H	D101
	(s5)+2	Buffer a rinfresco automatico della prima stazione locale (stazione 1)	Non utilizzato	0 _H	D102

Il contenuto dei registri dati da D1 a D102 deve essere impostato secondo la tabella precedente, prima di richiamare l'istruzione RLPA.

- Editor IEC (questo programma di esempio è mostrato a pagina seguente con lista istruzioni MELSEC e schema a contatti di GX Developer).

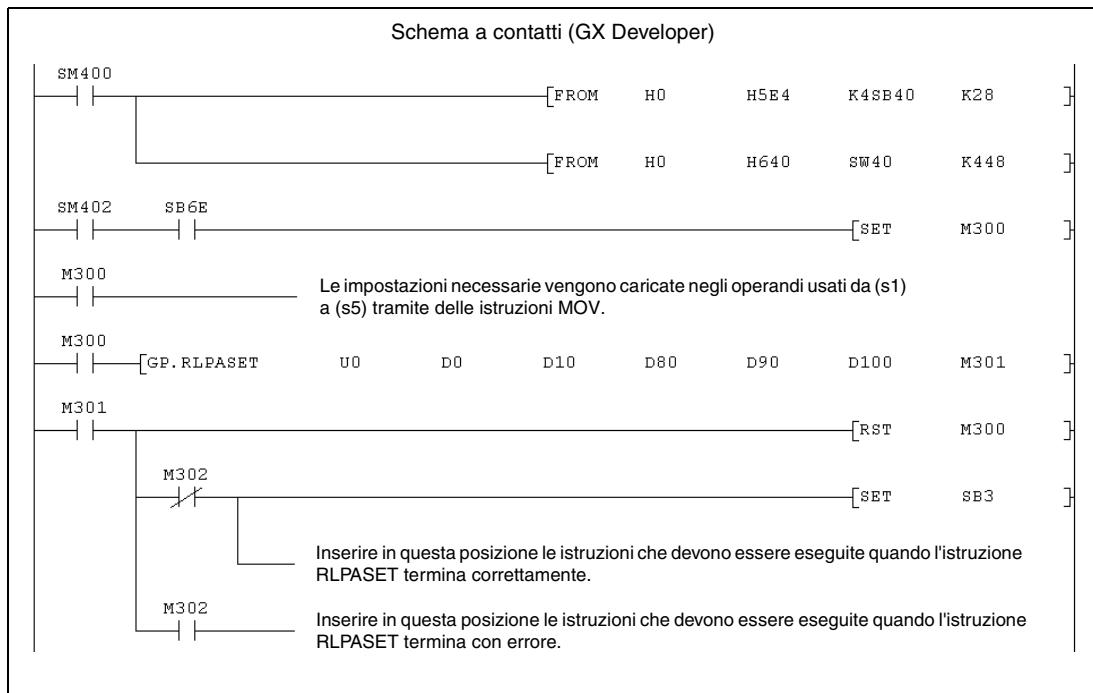


Lista istruzioni IEC					
LD	TRUE				
FROM_M	16#0000,	16#05E4,	28,	K4SB40	
FROM_M	16#0000,	16#0640,	448,	SW40	
.....					
LD	SM402				
AND	SB6E				Per una spiegazione su operandi e istruzioni usati, consultare
SET_M	M300				il programma di esempio a pagina precedente.
.....					
LD	M300				
Le impostazioni necessarie vengono caricate negli operandi usati da (s1) a (s5) tramite delle istruzioni MOV.					
.....					
LD	M300				
RLPASET_P_MD	0, var_D0, var_D10, var_D80, var_D90, var_D100, var_M301				
.....					
LD	M301				
RST_M	M300				
LD	M301				
ANDN	M302				
SET_M	SB3				
Una istruzione in questa posizione viene eseguita quando l'istruzione RLPASET è stata completata normalmente.					
LD	M301				
AND	M302				
Una istruzione in questa posizione viene eseguita quando l'istruzione RLPASET è stata completata con errore.					

NOTA

Negli editor IEC è necessario definire le variabili nello header della unità organizzativa di programma (POU) La mancata definizione delle variabili provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

- Lista istruzioni MELSEC e schema a contatti di GX Developer
Per una spiegazione degli operandi e istruzioni utilizzate, vedere il programma di esempio per lo schema a contatti di GX IEC Developer.



Lista istruzioni MELSEC

MELSEC	LD	SM400
	FROM	H0 H5E4 K4SB40 K28
MELSEC	FROM	H0 H640 SW40 K448
	LD	SM402
MELSEC	AND	SB6E
	SET	M300
MELSEC	LD	M300
	Le impostazioni necessarie vengono caricate negli operandi usati da (s1) a (s5) tramite delle istruzioni MOV.	
MELSEC	LD	M300
	GP.RLPASET	U0 D0 D10 D80 D90 D100 M301
	LD	M301
	RST	M300
	MPS	
	ANI	M302
	SET	SB3
	MPP	
MELSEC	AND	M302
	Una istruzione in questa posizione viene eseguita quando l'istruzione RLPASET è stata completata con errore.	

11.5.3 RRP A (serie A)

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●		

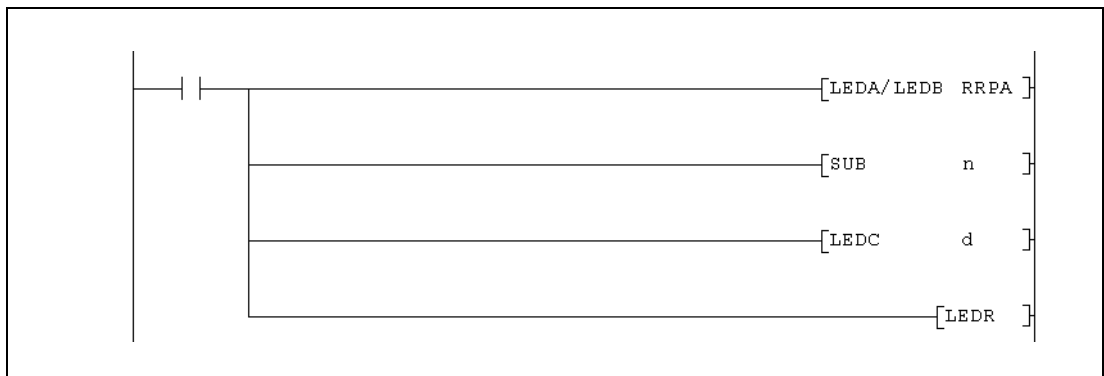
Operandi MELSEC A

	Operandi utilizzabili														Definizione cifra	Numero di passi	Indice	Rapporto	Errore										
	Operandi a bit						Operandi a word (16-bit)						Costanti							Puntatore		Livello							
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1						Z	V	K	H (16#)	P	I	N			
n																●	●												
d							●	●	●	●	●																		●

GX IEC Developer

<p style="text-align: center;">Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p>MELSEC</p> </div> <table style="margin-left: 20px;"> <tr> <td>LEDA/LEDB</td> <td>RRPA</td> </tr> <tr> <td>SUB</td> <td>n</td> </tr> <tr> <td>LEDC</td> <td>d</td> </tr> <tr> <td>LEDR</td> <td></td> </tr> </table>	LEDA/LEDB	RRPA	SUB	n	LEDC	d	LEDR		<p style="text-align: center;">Schema a contatti</p>	<p style="text-align: center;">Lista istruzioni IEC</p> <table style="margin-left: 20px;"> <tr> <td>RRPA_MD</td> <td>n, d</td> </tr> </table>	RRPA_MD	n, d
LEDA/LEDB	RRPA											
SUB	n											
LEDC	d											
LEDR												
RRPA_MD	n, d											

GX Developer



Per ulteriori informazioni sulla programmazione delle istruzioni dedicate con gli editor MELSEC, fare riferimento al capitolo 3.3 di questo manuale.

Variabili

Dati impostati	Significato		Campo	Contenuto inserito da	Tipo dati	
n	Indirizzo iniziale di I/O del modulo master CC-Link (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come 10 _H)		da 0 a FE _H	Utente	BIN 16-bit	
d	Indirizzo iniziale dell'area operandi contenenti i parametri di rinfresco					
	Dati impostati	Significato	Descrizione	Contenuto inserito da	BIN 16-bit	
	d+0	R X	Indirizzo iniziale	Indirizzo iniziale degli ingressi remoti (RX) nel modulo master o locale		Sistema
	d+1		Operando nella CPU del PLC	Definisce gli operandi della CPU con rinfresco automatico, tramite il codice operando riportato nella tabella seguente.		Utente
	d+2		Indirizzo iniziale degli operandi lato CPU	Imposta l'indirizzo del primo operando lato CPU*		
	d+3		Numero punti aggiornati	Numero di operandi scambiati fra CPU e modulo CC-Link*		
	d+4	R Y	Indirizzo iniziale	Indirizzo iniziale delle uscite remote (RY) nel modulo master o locale		Utente
	d+5		Operando nella CPU del PLC	Definisce gli operandi della CPU con rinfresco automatico, tramite il codice operando riportato nella tabella seguente.		
	d+6		Indirizzo iniziale degli operandi lato CPU	Imposta l'indirizzo del primo operando lato CPU*		
	d+7		Numero punti aggiornati	Numero di operandi scambiati fra CPU e modulo CC-Link*		
	d+8	R W	Indirizzo iniziale	Indirizzo iniziale dei registri remoti (RW) nel modulo master o locale		RW: sistema
	d+9		Operando nella CPU del PLC	Definisce gli operandi della CPU con rinfresco automatico, tramite il codice operando riportato nella tabella seguente.		RW: Utente
	d+10		Indirizzo iniziale degli operandi lato CPU	Imposta l'indirizzo del primo operando lato CPU*		Utente
	d+11		Numero punti aggiornati	Numero di operandi scambiati fra CPU e modulo CC-Link*		
	d+12	S B	Indirizzo iniziale	Indirizzo iniziale dei relé speciali (SB) nel modulo master o locale		Sistema
	d+13		Operando nella CPU del PLC	Definisce gli operandi della CPU con rinfresco automatico, tramite il codice operando riportato nella tabella seguente.		Utente
	d+14		Indirizzo iniziale degli operandi lato CPU	Imposta l'indirizzo del primo operando lato CPU*		
	d+15		Numero punti aggiornati	Numero di operandi scambiati fra CPU e modulo CC-Link*		
	d+16	S W	Indirizzo iniziale	Indirizzo iniziale dei registri speciali di comunicazione (SW) nel modulo master o locale		Sistema
	d+17		Operando nella CPU del PLC	Definisce gli operandi della CPU con rinfresco automatico, tramite il codice operando riportato nella tabella seguente.		Utente
d+18	Indirizzo iniziale degli operandi lato CPU		Imposta l'indirizzo del primo operando lato CPU*			
d+19	Numero punti aggiornati		Numero di operandi scambiati fra CPU e modulo CC-Link*			

* Inserire „0“ o un multiplo di „16“ come indirizzo operando degli operandi a bit (X, Y, M, B) e il numero dei punti a rinfresco automatico. Altrimenti viene generato un errore. (tuttavia se viene impostato „0“ come numero di punti di rinfresco, l'operando corrispondente non viene rinfrescato).

In d+5, d+13 ecc. viene attivato l'operando della CPU del PLC che corrisponde a un operando del modulo CC-Link. Ad esempio, si possono utilizzare i relé interni (M) per indicare lo stato degli ingressi remoti (RX).

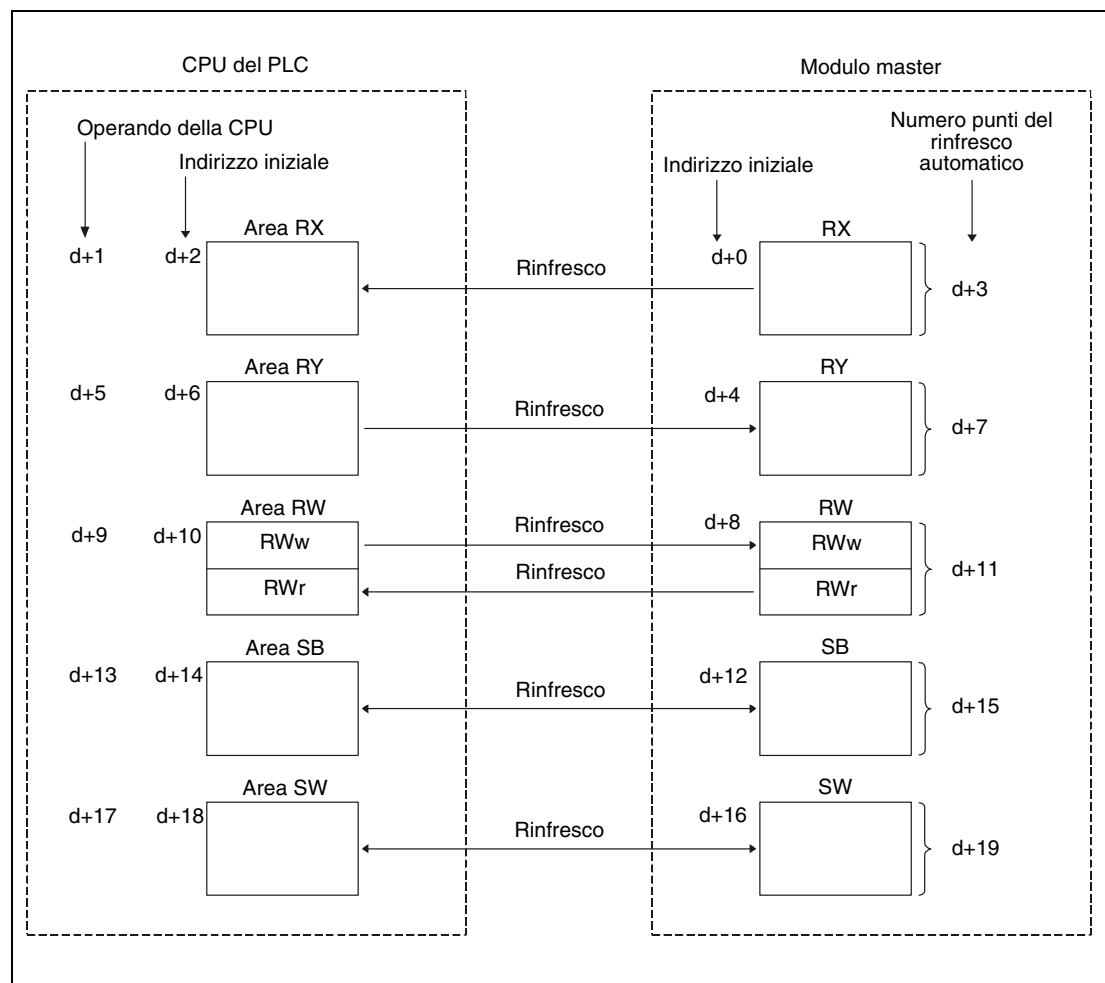
Codice	Operando	Codice	Operando
0	—	5	T
1	X	6	C
2	Y	7	D
3	M	8	W
4	B	9	R

Funzioni Impostazione dei parametri di rinfresco automatico

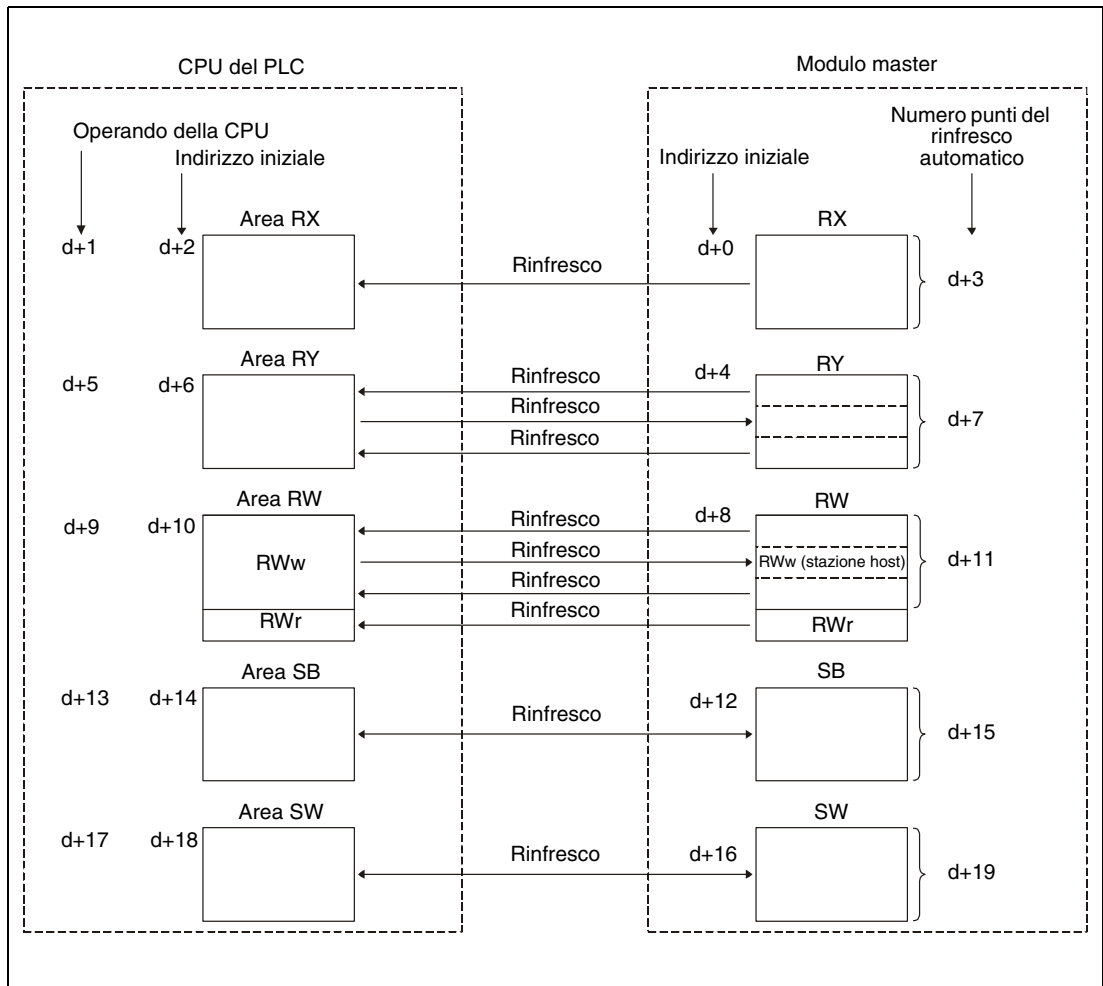
RRPA Istruzione di parametrizzazione

L'istruzione RRPA definisce gli operandi e il numero di punti su cui viene eseguito il rinfresco automatico fra CPU e modulo master/locale. Se vengono usate le istruzioni FROM/TO per leggere o scrivere dati da e per un modulo master/locale, l'esecuzione dell'istruzione RRPA non è necessaria.

- Scambio dati fra CPU del PLC e stazione master



● Scambio dati fra CPU del PLC e stazione locale



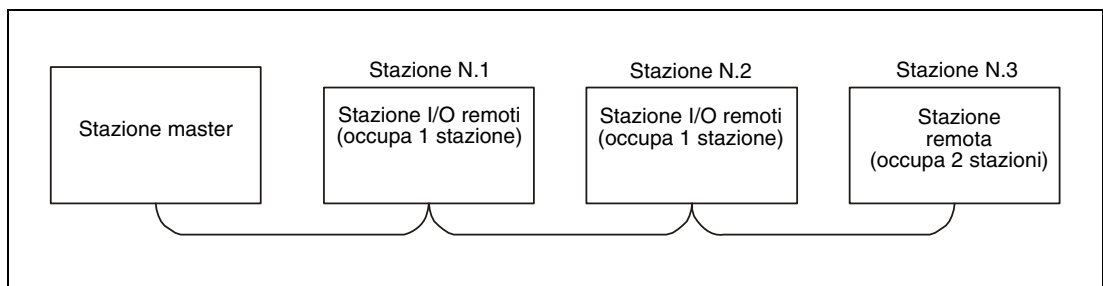
Quando viene eseguita l'istruzione RRPA, i dati di rinfresco automatico vengono registrati nella CPU e viene avviato il rinfresco automatico fra CPU e modulo master/locale.

L'istruzione RRPA viene eseguita una sola volta all'ingresso della CPU in modo RUN. Se vengono programmate diverse istruzioni RRPA per lo stesso modulo, sono valide solo le configurazioni della prima istruzione. Se si vogliono modificare le impostazioni, eseguire l'istruzione RRPA con i nuovi parametri. Dopo che la CPU è stata commutata su STOP/PAUSE e poi di nuovo su RUN, i nuovi parametri di rinfresco verranno considerati.

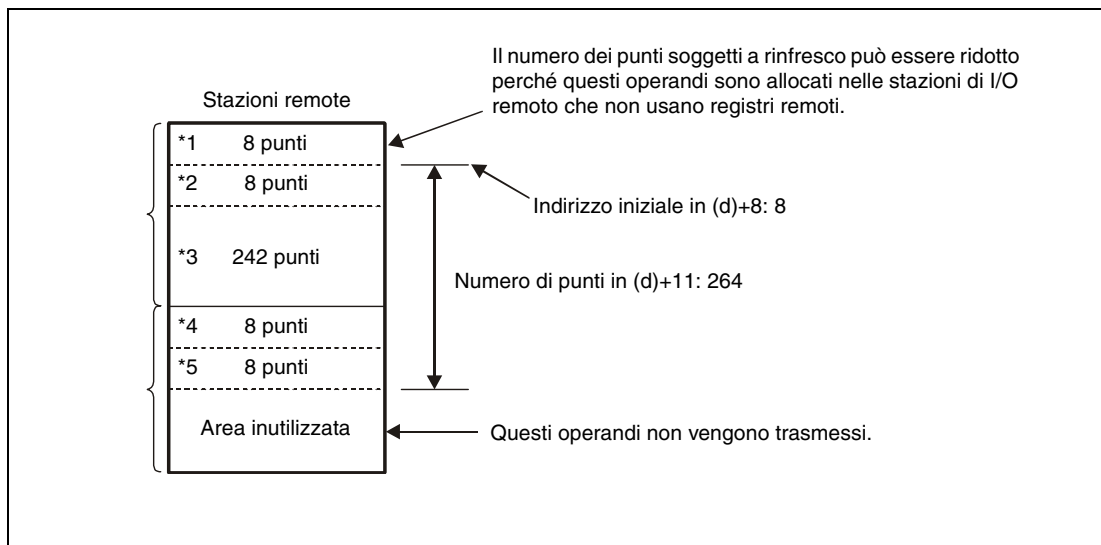
Per rinfrescare tutte le aree dei registri remoti (RWw e RWr), scrivere „0“ come indirizzo iniziale in d+8 e 512 come numero di punti in d+11.

NOTA

La configurazione di sistema seguente viene utilizzata per spiegare il rinfresco dei registri remoti:



Tutte le 256 word (per 64 stazioni) di RWw nell'area RW vengono occupate anche se il numero totale di stazioni è inferiore a 64. L'inizio dell'area RWr parte quindi dopo questi 256 punti RWw.



*1: Area RWw (8 punti) delle stazioni 1 e 2 (stazioni I/O remoto)

*2: Area RWw (8 punti) della stazione 3 (stazione I/O remoto)

*3: 242 punti dell'area RWw vengono occupati automaticamente dal sistema

*4: Area RWr (8 punti) delle stazioni 1 e 2 (stazioni I/O remoto)

*5: Area RWr (8 punti) della stazione 3 (stazione I/O remoto)

Configurazione degli operandi SB e SW con rinfresco:

- alloca gli operandi della CPU del PLC rinfrescati ai relé speciali (SB) e ai registri speciali (SW). Notare la direzione del rinfresco: i relé da SB0000 a SB003F vengono rinfrescati dalla CPU verso il modulo master, mentre da SB0040 a SB00FF vengono rinfrescati dal modulo master verso la CPU.
- Il file registri (R) non può essere specificato come operando rinfrescato per SB e SW. Se si specificano registri su file per SB e SW e impostati nella CPU, viene generato un codice di errore e la CPU cessa di funzionare.
- Il campo operandi per gli operandi rinfrescati in SB o SW non deve comprendere zone retentive. Se si specificano operandi per SB e SW nel campo retentivo, si possono avere anomalie di comportamento a causa dello stato indefinito dei dati all'avviamento/reset.
- I campi di rinfresco per SB e SW definiti all'accensione con l'istruzione RRP A, non possono essere modificati.

Condizione di esecuzione

Se viene usata l'istruzione LEDA, l'istruzione RRP A viene eseguita ad ogni scansione in cui il comando di scrittura è ON.

Se viene usata l'istruzione LEDB, l'istruzione RRP A viene eseguita per una scansione sul fronte positivo (OFF -> ON) del comando di scrittura.

Errori di esecuzione

Nei casi seguenti, si verifica un errore di funzionamento, viene attivato il flag di errore M9011, ed il codice di errore „50“ viene memorizzato in D908: (il codice di errore „503“ è scritto in D9001 se si usa una CPU AnU e in D9092 se si usa una CPU AnSH).

- Il codice operando è „0“ o diverso da 1 a 9.
- L'indirizzo iniziale di un operando a bit non è „0“ o multiplo di 16.
- Il numero dei punti rinfrescati non è multiplo di 16.

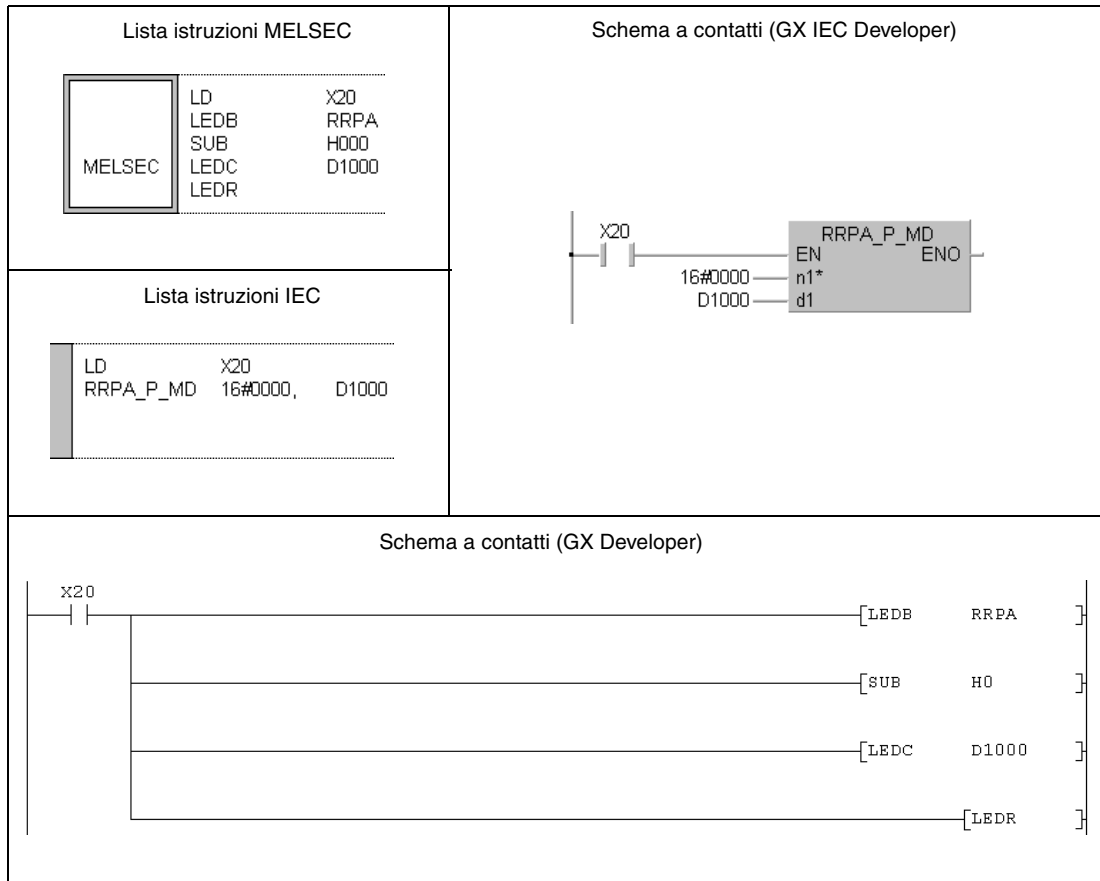
Esempio di programma

RRPA

Il programma seguente imposta i parametri di rinfresco automatico per il modulo master corrispondente agli indirizzi di I/O da X/Y000 a X/Y01F. Le impostazioni sono contenute a partire dal registro D1000:

Parametro		Impostazione	Dati impostati	Dispositivo di memoria
RX	Indirizzo iniziale	0	0 _H	D1000
	Operando nella CPU del PLC	X (codice: 1)	1 _H	D1001
	Indirizzo iniziale degli operandi lato CPU	XA0	A0 _H	D1002
	Numero punti aggiornati	32	32	D1003
RY	Indirizzo iniziale	0	0 _H	D1004
	Operando nella CPU del PLC	Y (codice: 2)	2 _H	D1005
	Indirizzo iniziale degli operandi lato CPU	YA0	A0 _H	D1006
	Numero punti aggiornati	48	48	D1007
RW	Indirizzo iniziale	0	0 _H	D1008
	Operando nella CPU del PLC	D (codice: 7)	7 _H	D1009
	Indirizzo iniziale degli operandi lato CPU	D160	160	D1010
	Numero punti aggiornati	272	272	D1011
SB	Indirizzo iniziale	0	0 _H	D1012
	Operando nella CPU del PLC	M (codice: 3)	3 _H	D1013
	Indirizzo iniziale degli operandi lato CPU	M160	160	D1014
	Numero punti aggiornati	256	256	D1015
SW	Indirizzo iniziale	0	0 _H	D1016
	Operando nella CPU del PLC	W (codice: 8)	8 _H	D1017
	Indirizzo iniziale degli operandi lato CPU	WA0	A0 _H	D1018
	Numero punti aggiornati	256	256	D1019

Il contenuto dei registri dati da D1000 a D1019 deve essere impostato secondo la tabella precedente, prima di richiamare l'istruzione RRP A.



11.5.4 RIRD (serie A)

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●		

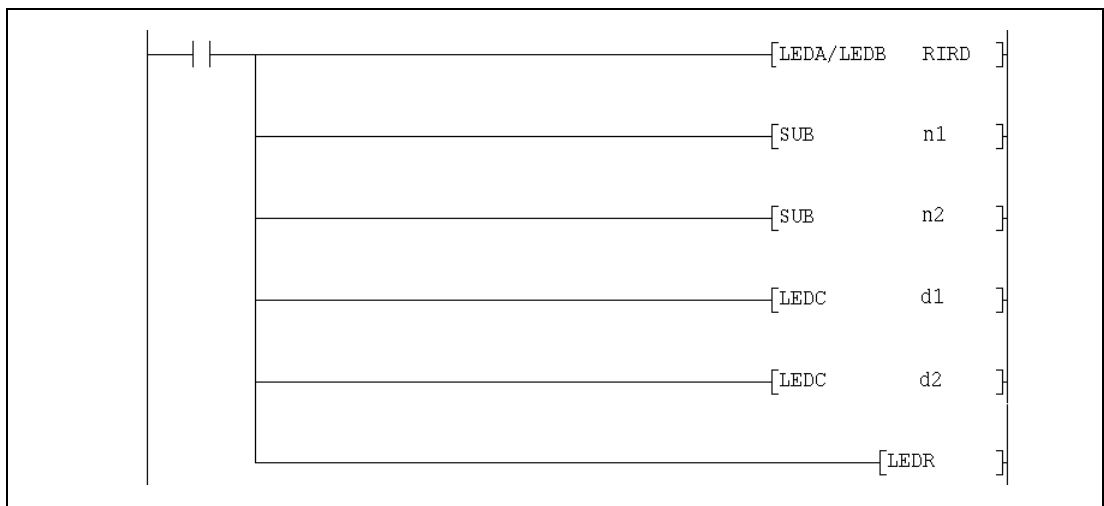
Operandi MELSEC A

	Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Rapporto	Errore			
	Operandi a bit						Operandi a word (16-bit)						Costanti		Puntatore							Livello		
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V						K	H (16#)	P
n1																	●	●						
n2																	●	●						
d1							●	●	●	●	●													
d2	●	●	●	●	●																		●	

GX IEC Developer

<p style="text-align: center;">Lista istruzioni MELSEC</p> <table border="1" style="width: 100%;"> <tr> <td style="text-align: center;">MELSEC</td> <td>LEDA/LEDB</td> <td>RIRD</td> </tr> <tr> <td></td> <td>SUB</td> <td>n1</td> </tr> <tr> <td></td> <td>SUB</td> <td>n2</td> </tr> <tr> <td></td> <td>LEDC</td> <td>d1</td> </tr> <tr> <td></td> <td>LEDC</td> <td>d2</td> </tr> <tr> <td></td> <td>LEDR</td> <td></td> </tr> </table>	MELSEC	LEDA/LEDB	RIRD		SUB	n1		SUB	n2		LEDC	d1		LEDC	d2		LEDR		<p style="text-align: center;">Schema a contatti</p>	<p style="text-align: center;">Lista istruzioni IEC</p> <table border="1" style="width: 100%;"> <tr> <td style="text-align: center;">RIRD_MD</td> <td>n1, n2, d1, d2</td> </tr> </table>	RIRD_MD	n1, n2, d1, d2
MELSEC	LEDA/LEDB	RIRD																				
	SUB	n1																				
	SUB	n2																				
	LEDC	d1																				
	LEDC	d2																				
	LEDR																					
RIRD_MD	n1, n2, d1, d2																					

GX Developer



Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati	
n1	Indirizzo iniziale di I/O del modulo master/locale CC-Link (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come 10 _H)	da 0 a FE _H	Utente	BIN 16-bit	
n2	Numero di stazione della stazione remota, da cui si leggono i dati Campo: Se l'istruzione RIRD viene eseguita nella stazione master: da 1 a 64 Se l'istruzione RIRD viene eseguita in una stazione locale: da 0 a 64		Utente	BIN 16-bit	
d1	Indirizzo iniziale dell'area operandi in cui sono contenuti i dati di controllo dell'istruzione e vengono memorizzati i dati letti.				
	Operando	Significato	Descrizione	Campo	
	(d1)+0	Risultato dell'esecuzione dell'istruzione	Indica se si è verificato un errore durante l'esecuzione dell'istruzione: 0000 _H : Nessun errore Qualsiasi valore diverso da 0000 _H : Si è verificato un errore. Il valore memorizzato è un codice di errore descritto nel manuale utente del modulo CC-Link.	—	Sistema
	(d1)+1	Numero di punti da leggere	Specificare il numero di dati (unità: word) da leggere. Questo numero dipende dal tipo del modulo CPU montato sulla stazione da cui vengono letti i dati: AnU, serie QnA, System Q: max. 480 word Tutte le altre CPU: max. 32 word	da 1 a 480 da 1 a 32	
	(d1)+2	Codice di accesso	<ul style="list-style-type: none"> Per moduli master con versione software da A a H. Impostare „0004_H“ per accedere al buffer di memoria di una stazione intelligente. Impostare „2004_H“ per accedere al buffer di memoria ad accesso casuale di una stazione locale. 	0004 _H o 2004 _H	
	(d1)+3	Codice operando e codice di accesso	<ul style="list-style-type: none"> Per moduli master con versione software J o superiore Gli 8 bit pesanti di questo operando contengono un codice operando. Il codice di accesso, che specifica se l'accesso è per il buffer di memoria di un modulo CC-Link (04_H) o per una CPU (05_H), è inserito negli 8 bit più leggeri. 	Byte più pesante: vedi tabella seguente Byte più leggero: 04 _H o 05 _H	Utente
	(d1)+3	Indirizzo iniziale	<ul style="list-style-type: none"> Per moduli master con versione software da A a H. Indirizzo iniziale del buffer di memoria Per moduli master con versione software J o superiore Indirizzo iniziale del buffer di memoria o indirizzo del primo operando 	Dipende dalla stazione indirizzata	
	(d1)+4 a (d1)+n	Area di memorizzazione dei dati letti	La dimensione di questa area è determinata dal numero di punti da leggere contenuto in (d1)+1.	—	Sistema

Variabili

Dati impostati	Significato			Campo	Contenuto inserito da	Tipo dati
d2	Operando a bit attivato per una scansione al termine dell'istruzione RIRD. (d2)+1 indica se si è verificato un errore durante l'esecuzione dell'istruzione.					Bit
	Operando	Significato	Descrizione	Campo	Contenuto inserito da	
	(d2)+0	Istruzione completata	Indica lo stato di completamento dell'istruzione RIRD ON: Istruzione completata OFF: Istruzione non completata	—	Sistema	
(d2)+1	Istruzione completata con errore	Indica l'occorrenza di un errore durante l'elaborazione dell'istruzione RIRD ON: Completamento anomalo OFF: Completamento normale	—			

A partire dalla versione J del software del modulo master, vengono utilizzati due codici (entrambi contenuti in (d1)+2) per specificare i dati da leggere. Il **codice di accesso** determina se l'accesso viene fatto al buffer di memoria di un modulo CC-Link oppure alla memoria operandi del modulo CPU. Il **codice operando** indica l'area del buffer di memoria o degli operandi:

- Accesso al buffer di memoria di un modulo CC-Link (codice di accesso: 04_H)

Accesso a	Codice operando	
Buffer di memoria di una stazione intelligente	00 _H	
Buffer di memoria di una stazione master o locale	Buffer ad accesso casuale	20 _H
	Ingressi remoti	21 _H
	Uscite remote	22 _H
	Registri remoti	24 _H
	Relé speciali di comunicazione	63 _H
	Registri speciali di comunicazione	64 _H

- Accesso alla memoria operandi di un modulo CPU (codice di accesso: 05_H)
Gli operandi non indicati nella tabella seguente non sono accessibili. Per accedere ad un operando a bit, specificare „0“ o un multiplo di „16“ come operando iniziale. Altrimenti viene generato un errore.

Operandi		Tipo operando		Codice operando
Nome	Simbolo	Bit	Word	
Ingressi	X	●		00 _H
Uscite	Y	●		02 _H
Relé interni	M	●		03 _H
Relé retentivi	L	●		83 _H
Relé di comunicazione	B	●		23 _H
Timer (contatto)	T	●		09 _H
Timer (bobina)		●		0A _H
Timer (valore attuale)			●	
Contatore (contatto)	C	●		11 _H
Contatore (bobina)		●		12 _H
Contatore (valore attuale)			●	
Registro dati	D		●	04 _H
Registro di comunicazione	W		●	24 _H
File registri	R		●	84 _H

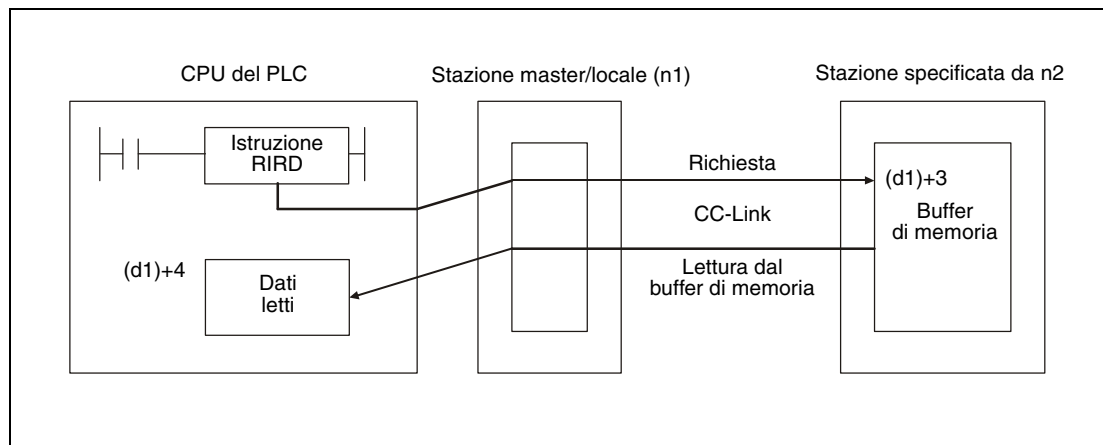
Funzioni **Letture dal buffer di memoria di una stazione intelligente o dalla memoria operandi della CPU del PLC.**

RIRD Lettura dati

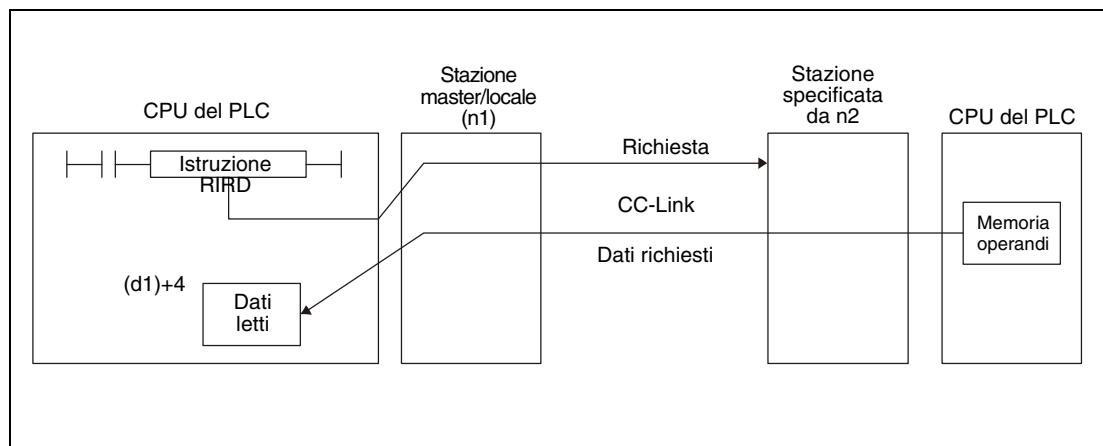
L'istruzione RIRD legge dati dal buffer di memoria di una stazione intelligente collegata al CC-Link. Se viene usato un modulo master con versione software J o superiore, è anche possibile accedere alla memoria operandi della CPU di un PLC montato nell'altra stazione.

L'indirizzo iniziale del buffer di memoria o il primo operando, viene specificato da $(d1)+3$. Il numero di stazione dell'altra stazione è indicato da $n2$. Questa stazione è collegata alla stazione master/locale specificata da $n1$. I dati letti vengono memorizzati nella CPU che esegue l'istruzione RIRD, negli operandi a partire da $(d1)+4$. Il numero di dati da leggere è specificato da $(d1)+1$.

- Funzionamento con versioni software da A a H:



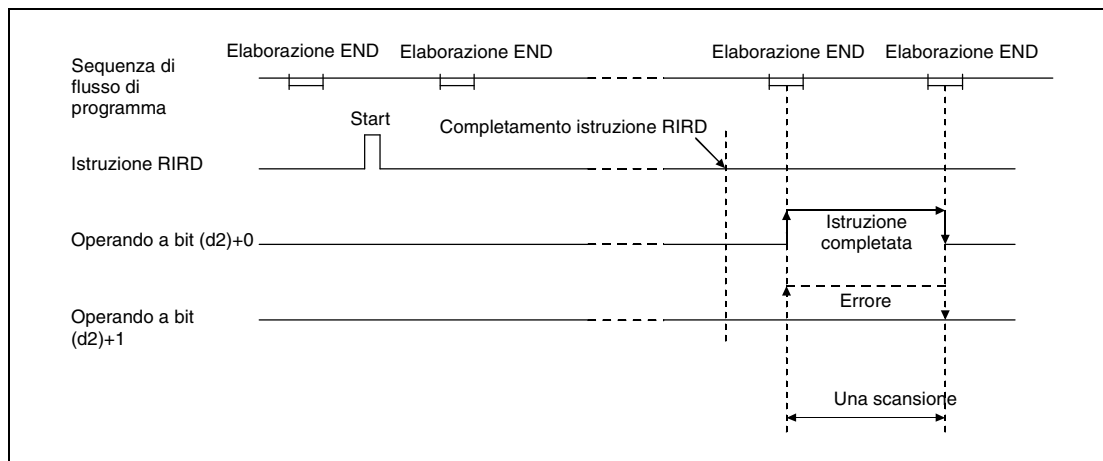
- Ulteriori funzioni con versione software J o superiore:



Lo stato di esecuzione dell'istruzione RIRD può essere controllato con gli operandi $(d2)+0$ e $(d2)+1$:

- L'operando a bit $(d2)+0$ si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione RIRD, e si disattiva con l'elaborazione di END della scansione successiva.
- L'operando a bit $(d2)+1$ segnala la presenza di un errore durante l'esecuzione di una istruzione RIRD. Se l'istruzione termina normalmente, l'operando rimane OFF. Se si verifica un errore durante l'esecuzione dell'istruzione RIRD, $(d2)+1$ si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione RIRD, e si disattiva con l'elaborazione di END della scansione successiva.

La figura seguente mostra la temporizzazione durante l'esecuzione di una istruzione RIRD:



È possibile eseguire contemporaneamente istruzioni RIRD su più stazioni, ma la stessa stazione intelligente o locale non può essere utilizzata simultaneamente da più di una stazione.

Impostare i parametri di rete eseguendo l'istruzione RLPA prima dell'istruzione RIRD.

Se viene inserito il valore „0“ o un valore fuori del campo da 1 a 480 come numero dei dati da leggere in (d1)+1, l'operando (d2)+1 viene attivato al termine dell'istruzione RIRD per indicare un errore.

Condizione di esecuzione

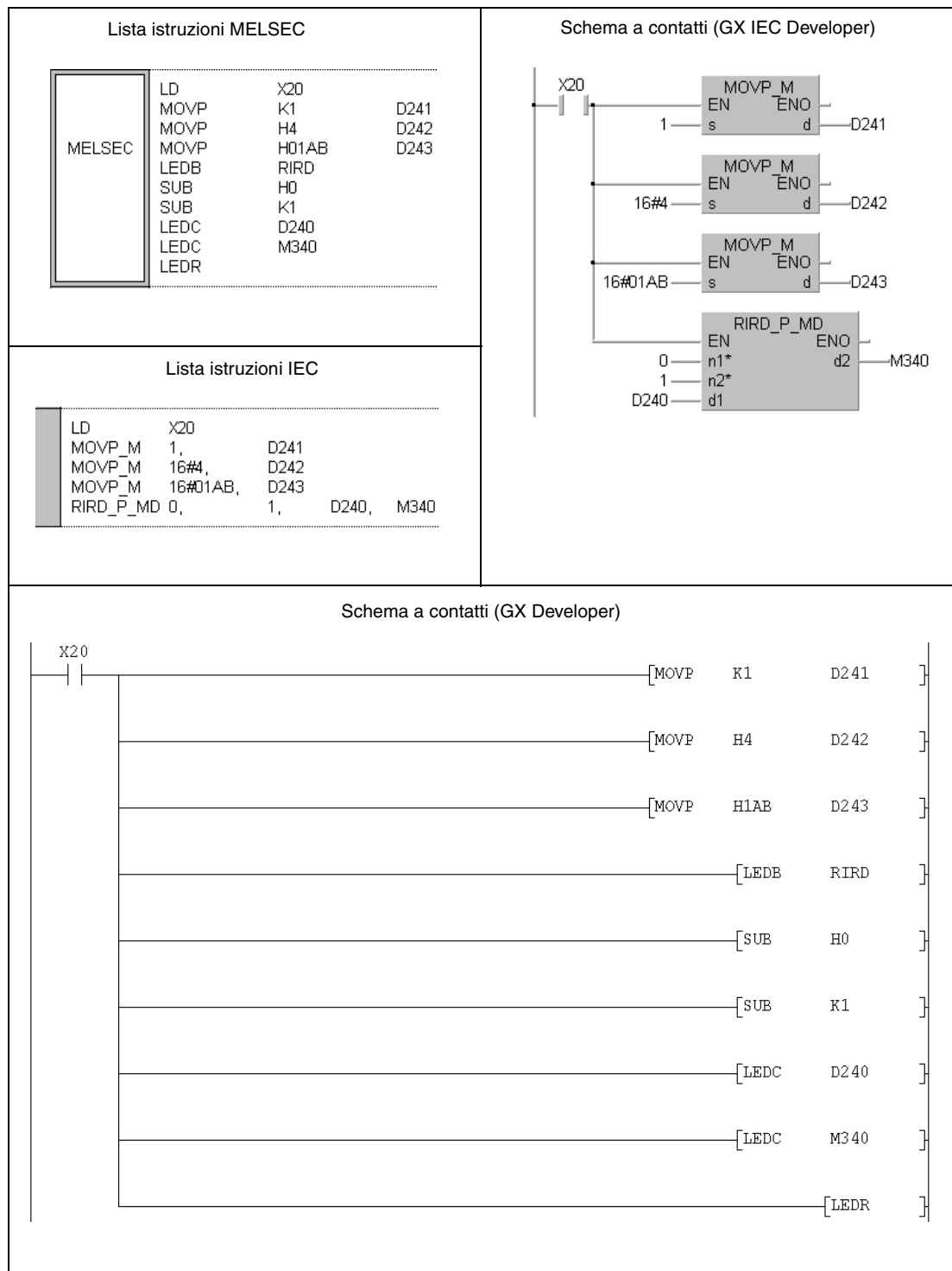
Se viene usata l'istruzione LEDA, l'istruzione RIRD viene eseguita ad ogni scansione in cui il comando di scrittura è ON.

Se viene usata l'istruzione LEDB, l'istruzione RIRD viene eseguita per una scansione sul fronte positivo (OFF -> ON) del comando di scrittura.

Notare che il processo di lettura seguito dall'istruzione RIRD richiede diverse scansioni prima del suo completamento. Eseguire quindi la successiva istruzione RIRD solo dopo che l'operando di completamento (d2)+0 viene attivato.

Esempio di programma RIRD

Il programma seguente viene eseguito nella CPU del PLC della stazione master. Legge il contenuto del buffer di memoria all'indirizzo 1A8_H da una stazione intelligente con numero di stazione 1. Il modulo master di CC-Link occupa gli indirizzi di I/O da X/Y000 a X/Y01F.



Per ulteriori informazioni sulla programmazione delle istruzioni dedicate con gli editor MELSEC, fare riferimento al capitolo 3.3 di questo manuale.

11.5.5 RIRD (serie QnA, System Q)

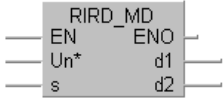
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

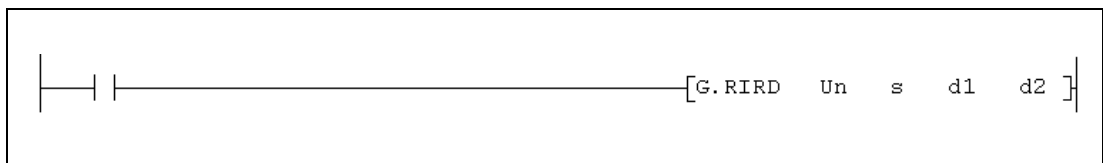
Operandi MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	—	—	SM0	8
d1	—	●	●	—	—	—	—	—	—		
d2	●	●	●	—	—	—	—	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p>MELSEC</p> </div> <p>G.RIRD Un s d1 d2</p>	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p>RIRD_MD</p> </div> <p>RIRD_MD Un, s, d1, d2</p>
--	--	--

GX Developer



Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati	
Un	Indirizzo iniziale di I/O del modulo master/locale CC-Link (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come 10 _H)	da 0 a FE _H	Utente	BIN 16-bit	
s	Indirizzo iniziale dell'area operandi in cui sono contenuti i dati di controllo dell'istruzione.				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(s)+0	Risultato dell'esecuzione dell'istruzione	Indica se si è verificato un errore durante l'esecuzione dell'istruzione: 0000 _H : Nessun errore Qualsiasi valore diverso da 0000 _H : Si è verificato un errore. Il valore memorizzato è un codice di errore descritto nel manuale utente del modulo CC-Link.	—	Sistema
	(s)+1	Numero stazione	Numero di stazione della stazione remota, da cui si leggono i dati	da 0 a 64	Utente
	(s)+2	Codice di accesso	<ul style="list-style-type: none"> Per moduli master serie A/Q con versione software da A a H. Impostare „0004_H“ per accedere al buffer di memoria di una stazione intelligente. Impostare „2004_H“ per accedere al buffer di memoria di una stazione locale. 	0004 _H o 2004 _H	
		Codice operando e codice di accesso	<ul style="list-style-type: none"> Per un modulo master con versione software J o superiore, o moduli System Q Gli 8 bit pesanti di questo operando contengono un codice operando. Il codice di accesso, che specifica se l'accesso è per il buffer di memoria di un modulo CC-Link (04 _H) o per una CPU (05 _H), è inserito negli 8 bit più leggeri.	Byte più pesante: vedi tabella seguente Byte più leggero: 04 _H o 05 _H	
(s)+3	Indirizzo iniziale	<ul style="list-style-type: none"> Per moduli master serie A/Q con versione software da A a H. Indirizzo iniziale del buffer di memoria Per un modulo master con versione software J o superiore, o moduli System Q Indirizzo iniziale del buffer di memoria o indirizzo del primo operando	Dipende dalla stazione indirizzata		
(s)+4	Numero di punti da leggere	Specificare il numero di dati (unità: word) da leggere. Questo numero dipende dal tipo del modulo CPU montato sulla stazione da cui vengono letti i dati: AnU, serie QnA, System Q: max. 480 word Tutte le altre CPU: max. 32 word	da 1 a 480 da 1 a 32		
d1	Indirizzo iniziale dell'area operandi che memorizza i dati letti.		Utente	BIN 16-bit	

Variabili

Dati impostati	Significato			Campo	Contenuto inserito da	Tipo dati
d2	Operando a bit attivato per una scansione al termine dell'istruzione RIRD. (d2)+1 indica se si è verificato un errore durante l'esecuzione dell'istruzione.					Bit
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da	
	(d2)+0	Istruzione completata	Indica lo stato di completamento dell'istruzione RIRD ON: Istruzione completata OFF: Istruzione non completata	—	Sistema	
(d2)+1	Istruzione completata con errore	Indica l'occorrenza di un errore durante l'elaborazione dell'istruzione RIRD ON: Completamento anomalo OFF: Completamento normale	—			

A partire dalla versione J del software del modulo master, vengono utilizzati due codici (entrambi contenuti in s+2) per specificare i dati da leggere: Il **codice di accesso** determina se l'accesso viene fatto al buffer di memoria di un modulo CC-Link oppure alla memoria operandi del modulo CPU. Il **codice operando** indica l'area del buffer di memoria o degli operandi:

- Accesso al buffer di memoria di un modulo CC-Link (codice di accesso: 04_H)

Accesso a		Codice operando
Buffer di memoria di una stazione intelligente		00 _H
Buffer di memoria di una stazione master o locale	Buffer ad accesso casuale	20 _H
	Ingressi remoti	21 _H
	Uscite remote	22 _H
	Registri remoti	24 _H
	Relé speciali di comunicazione	63 _H
	Registri speciali di comunicazione	64 _H

- Accesso alla memoria operandi di un modulo CPU (codice di accesso: 05_H)
Gli operandi non indicati nella tabella seguente non sono accessibili. Per accedere ad un operando a bit, specificare „0“ o un multiplo di „16“ come operando iniziale. Altrimenti viene generato un errore.

Operando		Tipo operando		Unità	Codice operando
Nome	Simbolo	Bit	Word		
Ingressi	X	●		Esadecimale	00 _H
Uscite	Y	●			02 _H
Relé interni	M	●		Decimale	03 _H
Relé retentivi	L	●			83 _H
Relé di comunicazione	B	●		Esadecimale	23 _H
Timer (contatto)	T	●		Decimale	09 _H
Timer (bobina)		●			0A _H
Timer (valore attuale)			●		0C _H
Timer retentivo (contatto)	ST	●			89 _H
Timer retentivo (bobina)		●			8A _H
Timer retentivo (valore attuale)			●		8C _H

Operando		Tipo operando		Unità	Codice operando
Nome	Simbolo	Bit	Word		
Contatore (contatto)	C	●		Decimale	11 _H
Contatore (bobina)		●			12 _H
Contatore (valore attuale)			●		14 _H
Registro dati	D		●		04 _H
Registro di comunicazione	W		●	Esadecimale	24 _H
File registri	R		●	Decimale	84 _H
Relé speciali di comunicazione	SB	●		Esadecimale	63 _H
Registri speciali di comunicazione	SW		●		64 _H
Relé speciale	SM	●		Decimale	43 _H
Registro speciale	SD		●		44 _H

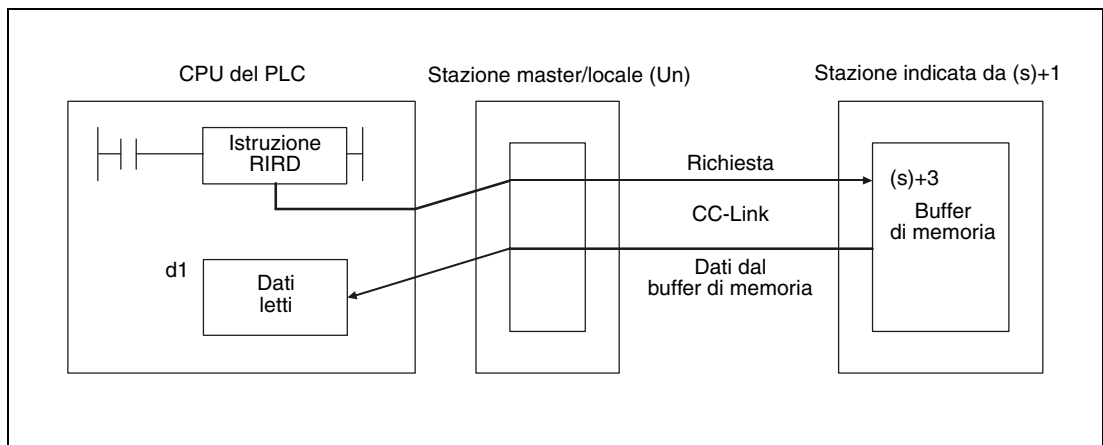
Funzioni **Letture dal buffer di memoria di una stazione intelligente o dalla memoria operandi della CPU del PLC**

RIRD Lettura dati

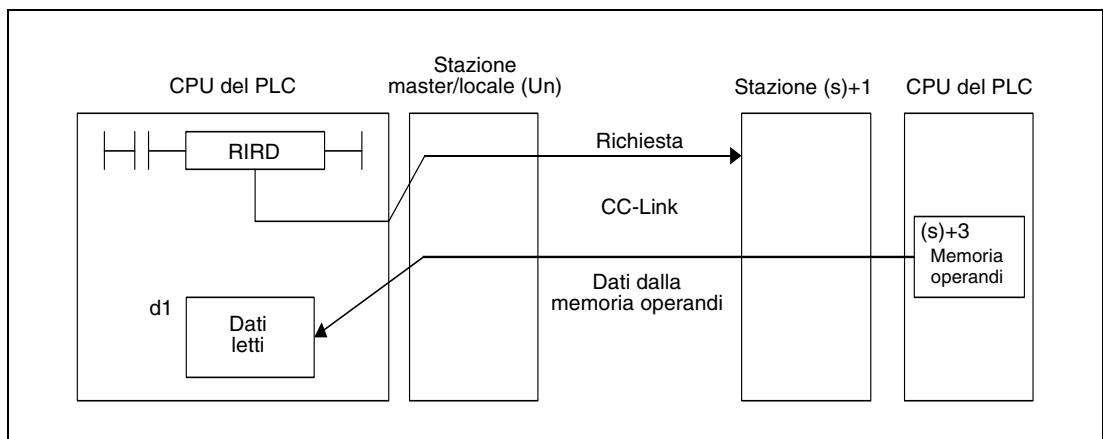
L'istruzione RIRD legge dati dal buffer di memoria di un dispositivo intelligente collegato al CC-Link. Se viene usato un modulo master con versione software J o superiore, oppure un modulo CC-Link del System Q MELSEC, è possibile anche accedere alla memoria operandi della CPU del PLC di un'altra stazione collegata alla rete CC-Link.

L'indirizzo iniziale del buffer di memoria o il primo operando è indicato da (s)+3. Il numero dell'altra stazione è indicato da (s)+1. Questa stazione è collegata alla stazione master/locale specificata da Un. I dati letti vengono memorizzati nella CPU che esegue l'istruzione RIRD, nell'area operandi a partire da d1. Il numero di dati da leggere è indicato da (s)+4.

- Accesso al buffer di memoria di un modulo CC-Link



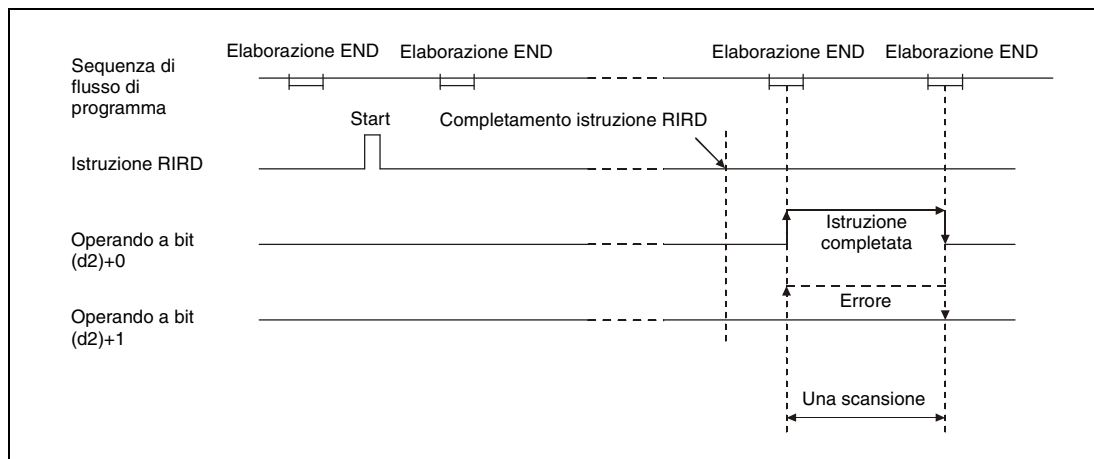
- Accesso alla memoria operandi della CPU di un'altra stazione CC-Link



Lo stato di esecuzione dell'istruzione RIRD può essere controllato con gli operandi (d2)+0 e (d2)+1:

- L'operando a bit (d2)+0 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione RIRD, e si disattiva con l'elaborazione di END della scansione successiva.
- L'operando a bit (d2)+1 segnala la presenza di un errore durante l'esecuzione di una istruzione RIRD. Se l'istruzione termina normalmente, l'operando rimane OFF. Se si verifica un errore durante l'esecuzione dell'istruzione RIRD, (d2)+1 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione RIRD, e si disattiva con l'elaborazione di END della scansione successiva.

La figura seguente mostra la temporizzazione durante l'esecuzione di una istruzione RIRD:



È possibile eseguire contemporaneamente istruzioni RIRD su più stazioni, ma la stessa stazione intelligente o locale non può essere utilizzata simultaneamente da più di una stazione.

Errori di esecuzione

Nei casi seguenti, si verifica un errore di funzionamento, viene attivato il flag di errore SM0, ed un codice di errore viene memorizzato in SD0:

- Se il modulo specificato da Un non è un modulo intelligente o un modulo funzione speciale. (codice di errore: serie QnA 2110, System Q 2112)
- Se si cerca di eseguire una istruzione non supportata (codice di errore: 4002).
- Se il numero di operandi dell'istruzione non è corretto. (codice di errore: 4003)
- Se l'istruzione specifica un operando che non si può usare. (codice di errore: 4004)
- Se l'area specificata da s contiene dati che non si possono usare. (codice di errore: 4100)
- Se il numero di dati impostato supera il campo ammesso. (codice di errore: 4101)
- Se l'area di memoria o le costanti degli operandi specificati con l'istruzione superano il campo ammesso (codice di errore: 4101)
- Solo serie QnA: Numero eccessivo di istruzioni speciali per CC-Link (codice di errore: 4107).
- Solo serie QnA: Parametri CC-Link non impostati. (codice di errore: 4108)

Esempio di programma

RIRD

Il programma seguente viene eseguito nella CPU del PLC della stazione master. Quando si attiva l'ingresso X0, il contenuto di 10 locazioni del buffer di memoria viene letto dalla stazione intelligente con il numero di stazione 1, a partire dall'indirizzo 100_H del buffer di memoria. I dati letti vengono memorizzati nella CPU del PLC a partire dal registro dati D0. L'indirizzo iniziale di I/O del modulo master CC-Link è X/Y40.

- Editor IEC (questo programma di esempio è mostrato a pagina seguente con lista istruzioni MELSEC e schema a contatti di GX Developer).

Schema a contatti (GX IEC Developer)

Lista istruzioni IEC

LD	X0			
ANDN	M100			
MOVP_M	1,	D101		
MOVP_M	16#4,	D102		
MOVP_M	16#100,	D103		
MOVP_M	10,	D104		
RIRD_P_MD	4,	var_D100,	D0,	var_MD
SET_M	M100			

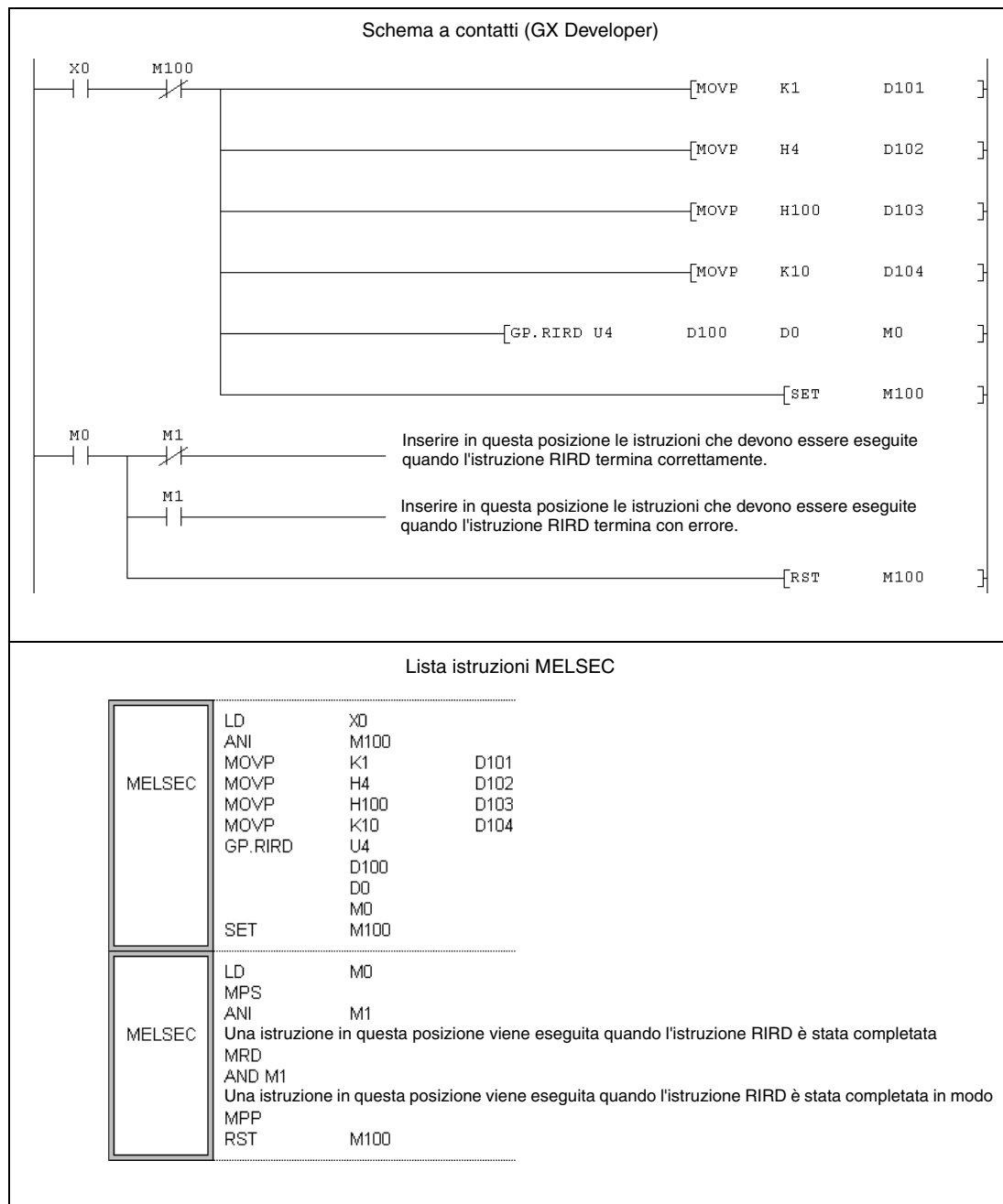
Per una spiegazione su operandi e istruzioni usate, fare riferimento al programma precedente.

LD	M0			
ANDN	M1			
Una istruzione in questa posizione viene eseguita quando l'istruzione RIRD è stata completata normalmente.				
LD	M0			
AND	M1			
Una istruzione in questa posizione viene eseguita quando l'istruzione RIRD è stata completata in modo anomalo.				
LD	M0			
RST_M	M100			

NOTA

Negli editor IEC è necessario definire le variabili nello header della unità organizzativa di programma (POU) La mancata definizione delle variabili provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

- Lista istruzioni MELSEC e schema a contatti di GX Developer
Per una spiegazione degli operandi e istruzioni utilizzate, vedere il programma di esempio per lo schema a contatti di GX IEC Developer nella pagina precedente.



11.5.6 RIWT (serie A)

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●		

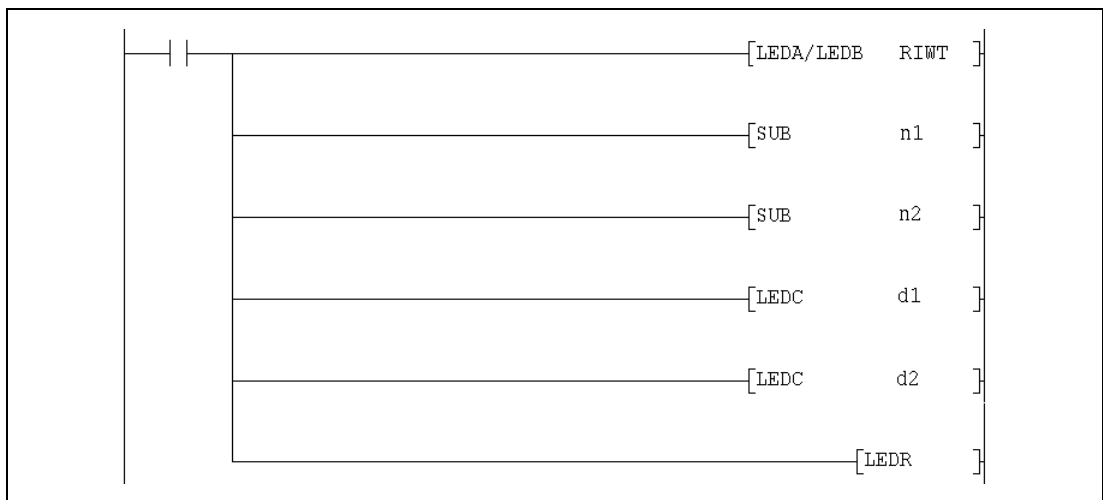
Operandi MELSEC A

	Operandi utilizzabili														Definizione cifra	Numero di passi	Indice	Rapporto	Errore				
	Operandi a bit						Operandi a word (16-bit)						Costanti							Puntatore		Livello	
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1						Z	V	K	H (16#)
n1															●	●					26		
n2														●	●								
d1							●	●	●	●	●												
d2	●	●	●	●	●																		

GX IEC Developer

<p style="text-align: center;">Lista istruzioni MELSEC</p> <table border="1" style="width: 100%;"> <tr> <td style="text-align: center;">MELSEC</td> <td>LEDA/LEDB</td> <td>RIWT</td> </tr> <tr> <td></td> <td>SUB</td> <td>n1</td> </tr> <tr> <td></td> <td>SUB</td> <td>n2</td> </tr> <tr> <td></td> <td>LEDC</td> <td>d1</td> </tr> <tr> <td></td> <td>LEDC</td> <td>d2</td> </tr> <tr> <td></td> <td>LEDR</td> <td></td> </tr> </table>	MELSEC	LEDA/LEDB	RIWT		SUB	n1		SUB	n2		LEDC	d1		LEDC	d2		LEDR		<p style="text-align: center;">Schema a contatti</p>	<p style="text-align: center;">Lista istruzioni IEC</p> <table border="1" style="width: 100%;"> <tr> <td style="text-align: center;">RIWT_MD</td> <td>n1, n2, d1, d2</td> </tr> </table>	RIWT_MD	n1, n2, d1, d2
MELSEC	LEDA/LEDB	RIWT																				
	SUB	n1																				
	SUB	n2																				
	LEDC	d1																				
	LEDC	d2																				
	LEDR																					
RIWT_MD	n1, n2, d1, d2																					

GX Developer



Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati
n1	Indirizzo iniziale di I/O del modulo master/locale CC-Link (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come 10 _H)	da 0 a FE _H	Utente	BIN 16-bit
n2	Numero di stazione della stazione remota, in cui si scrivono i dati Campo: Se l'istruzione RIWT viene eseguita nella stazione master: da 1 a 64 Se l'istruzione RIWT viene eseguita in una stazione locale: da 0 a 64		Utente	BIN 16-bit

Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati		
d1	Indirizzo iniziale dell'area operandi in cui sono contenuti i dati di controllo dell'istruzione e vengono memorizzati i dati letti.				BIN 16-bit	
	Dati impostati	Significato	Descrizione	Campo		Contenuto inserito da
	(d1)+0	Risultato dell'esecuzione dell'istruzione	Indica se si è verificato un errore durante l'esecuzione dell'istruzione: 0000 _H : Nessun errore Qualsiasi valore diverso da 0000 _H : Si è verificato un errore. Il valore memorizzato è un codice di errore descritto nel manuale utente del modulo CC-Link.	—		Sistema
	(d1)+1	Numero di punti da scrivere	Specificare il numero di dati (unità: word) da scrivere. Questo numero dipende dal tipo del modulo CPU montato sulla stazione in cui vengono scritti i dati: AnU, serie QnA, System Q: max. 480 word Tutte le altre CPU: max. 10 word	da 1 a 480 da 1 a 10		Utente
	(d1)+2	Codice di accesso	<ul style="list-style-type: none"> Per moduli master con versione software da A a H. Impostare „0004_H“ per scrivere nel buffer di memoria di una stazione intelligente. Impostare „2004_H“ per scrivere nel buffer di memoria di una stazione locale. 	0004 _H o 2004 _H		
		Codice operando e codice di accesso	<ul style="list-style-type: none"> Per moduli master con versione software J o superiore Gli 8 bit pesanti di questo operando contengono un codice operando. Il codice di accesso, che specifica se l'accesso è per il buffer di memoria di un modulo CC-Link (04 _H) o per una CPU (05 _H), è inserito negli 8 bit più leggeri.	Byte più pesante: vedi tabella seguente Byte più leggero: 04 _H o 05 _H		
	(d1)+3	Indirizzo iniziale	<ul style="list-style-type: none"> Per moduli master con versione software da A a H. Indirizzo iniziale del buffer di memoria	Dipende dalla stazione indirizzata		
			<ul style="list-style-type: none"> Per moduli master con versione software J o superiore Indirizzo iniziale del buffer di memoria o indirizzo del primo operando			
(d1)+4 a (d1)+n	Area contenente i dati da scrivere	Specificare la dimensione di questa area in (d1)+1	—	Utente		

Variabili

Dati impostati	Significato			Campo	Contenuto inserito da	Tipo dati
d2	Operando a bit attivato per una scansione al termine dell'istruzione RIWT. (d2)+1 indica se si è verificato un errore durante l'esecuzione dell'istruzione.					Bit
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da	
	(d2)+0	Istruzione completata	Indica lo stato di completamento dell'istruzione RIWT ON: Istruzione completata OFF: Istruzione non completata	—	Sistema	
(d2)+1	Istruzione completata con errore	Indica l'occorrenza di un errore durante l'elaborazione dell'istruzione RIWT ON: Completamento anomalo OFF: Completamento normale	—			

A partire dalla versione J del software del modulo master, vengono utilizzati due codici (entrambi contenuti in (d1)+2) per specificare la destinazione dei dati. Il **codice di accesso** determina se i dati vengono scritti nel buffer di memoria di un modulo CC-Link oppure nella memoria del modulo CPU. Il **codice operando** indica l'area del buffer di memoria o degli operandi che vengono sovrascritti:

- Scrittura nel buffer di memoria di un modulo CC-Link (codice di accesso: 04_H)

Accesso a	Codice operando	
Buffer di memoria di una stazione intelligente	00 _H	
Buffer di memoria di una stazione master o locale	Buffer ad accesso casuale	20 _H
	Ingressi remoti	21 _H
	Uscite remote	22 _H
	Registri remoti	24 _H
	Relé speciali di comunicazione	63 _H
	Registri speciali di comunicazione	64 _H

- Accesso alla memoria di un modulo CPU (codice di accesso: 05_H)

Gli operandi non indicati nella tabella seguente non sono accessibili. Per accedere ad un operando a bit, specificare „0“ o un multiplo di „16“ come operando iniziale. Altrimenti viene generato un errore.

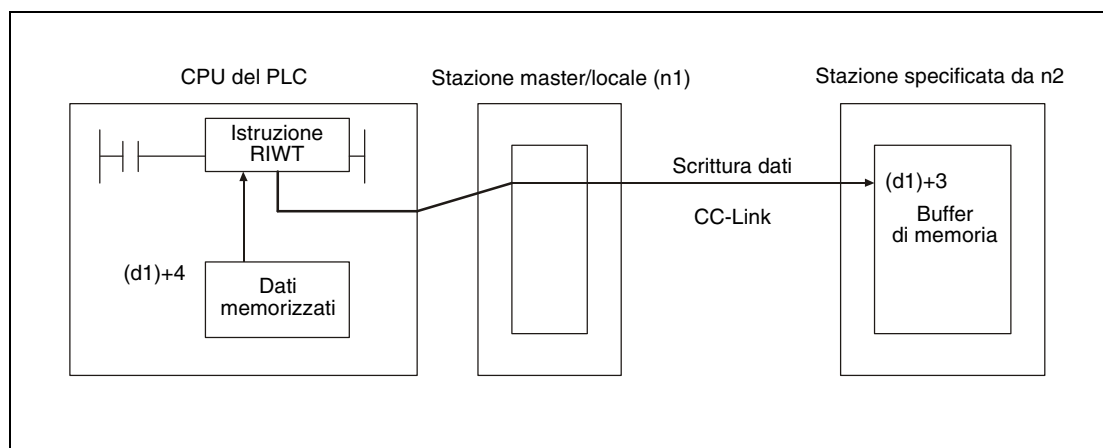
Operandi		Tipo operando		Codice operando
Nome	Simbolo	Bit	Word	
Ingressi	X	●		00 _H
Uscite	Y	●		02 _H
Relé interni	M	●		03 _H
Relé retentivi	L	●		83 _H
Relé di comunicazione	B	●		23 _H
Timer (contatto)	T	●		09 _H
Timer (bobina)		●		0A _H
Timer (valore attuale)			●	
Contatore (contatto)	C	●		11 _H
Contatore (bobina)		●		12 _H
Contatore (valore attuale)			●	
Registro dati	D		●	04 _H
Registro di comunicazione	W		●	24 _H
File registri	R		●	84 _H

Funzioni Scrittura nel buffer di memoria di una stazione intelligente o nella memoria della CPU del PLC**RIWT Scrittura dati**

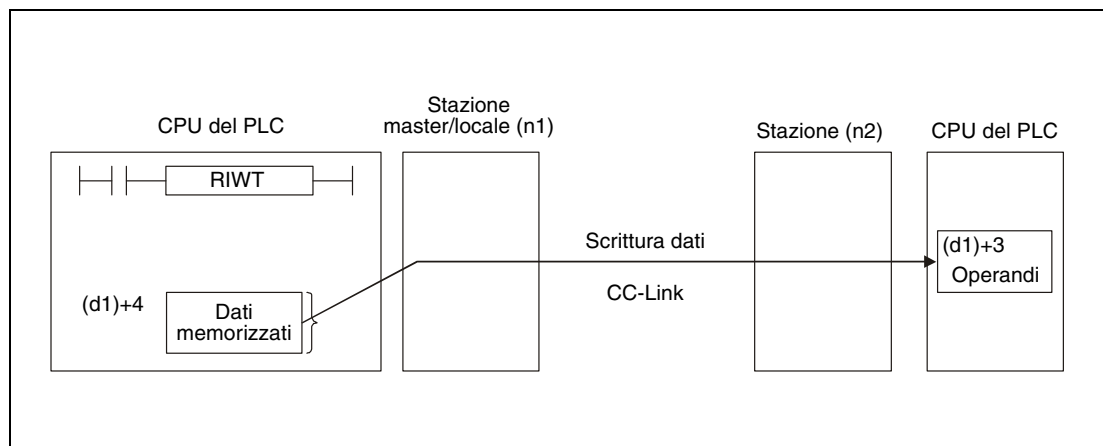
L'istruzione RIWT scrive dati nel buffer di memoria di una stazione intelligente collegata al CC-Link. Se viene usato un modulo master con versione software J o superiore, è anche possibile scrivere nella memoria della CPU di un PLC montato nell'altra stazione.

Il numero dell'altra stazione è specificato da n2. Questa stazione è collegata alla stazione master/locale specificata da n1. I dati per questa stazione sono contenuti nella CPU che esegue l'istruzione RIWT, nell'area operandi a partire da (d1)+4. Il numero di dati da scrivere è indicato da (d1)+1. L'indirizzo iniziale del buffer di memoria o l'indirizzo iniziale dell'area operandi è specificato da (d1)+3.

- Funzionamento con versioni software da A a H:



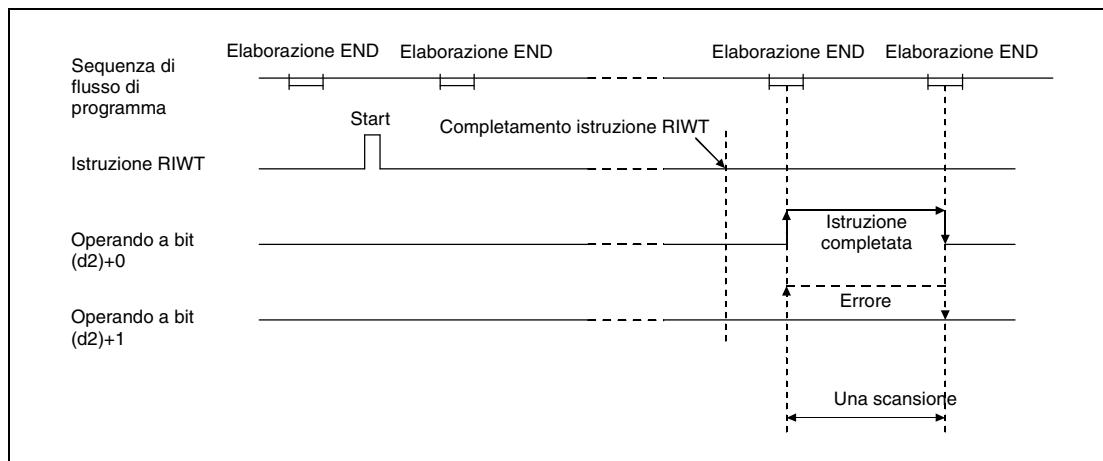
- Ulteriori funzioni con versione software J o superiore:



Lo stato di esecuzione dell'istruzione RIWT può essere controllato con gli operandi (d2)+0 e (d2)+1:

- L'operando a bit (d2)+0 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione RIWT, e si disattiva con l'elaborazione di END della scansione successiva.
- L'operando a bit (d2)+1 segnala la presenza di un errore durante l'esecuzione di una istruzione RIWT. Se l'istruzione termina normalmente, l'operando rimane OFF, ma se si verifica un errore durante l'esecuzione dell'istruzione RIWT, (d2)+1 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione RIWT, e si disattiva con l'elaborazione di END della scansione successiva.

La figura seguente mostra la temporizzazione durante l'esecuzione di una istruzione RIWT:



È possibile eseguire contemporaneamente istruzioni RIWT su più stazioni, ma la stessa stazione intelligente o locale non può essere utilizzata simultaneamente da più di una stazione.

Impostare i parametri di rete eseguendo l'istruzione RLPA prima dell'istruzione RIWT.

Se viene inserito il valore „0“ o un valore fuori del campo da 1 a 480 come numero dei dati da scrivere in (d1)+1, l'operando (d2)+1 viene attivato al termine dell'istruzione RIWT per indicare un errore.

Condizione di esecuzione

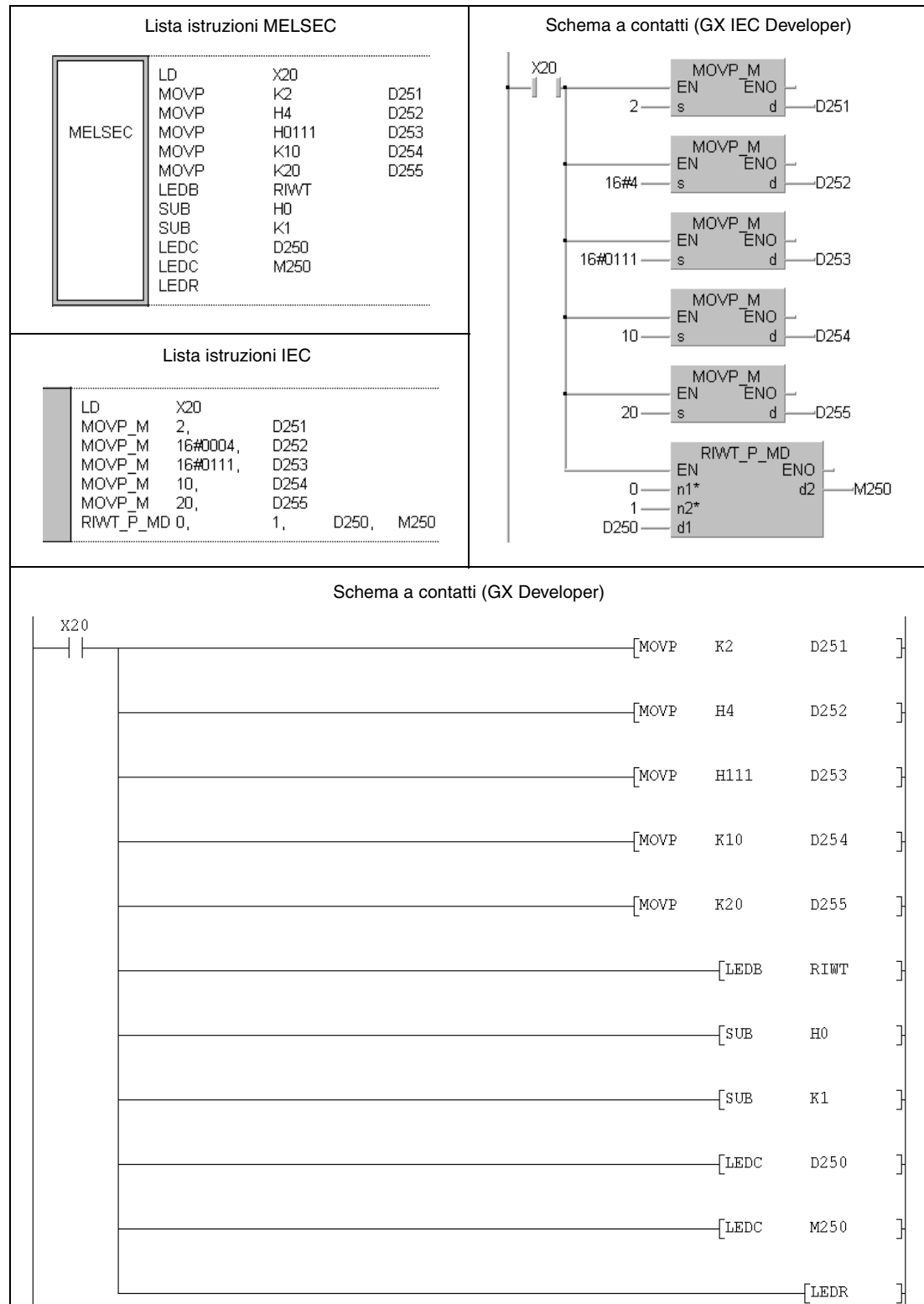
Se viene usata l'istruzione LEDA, l'istruzione RIWT viene eseguita ad ogni scansione in cui il comando di scrittura è ON.

Se viene usata l'istruzione LEDB, l'istruzione RIWT viene eseguita per una scansione sul fronte positivo (OFF -> ON) del comando di scrittura.

Notare che il processo di scrittura seguito dall'istruzione RIWT richiede diverse scansioni prima del suo completamento. Eseguire quindi la successiva istruzione RIWT solo dopo che l'operando di completamento (d2)+0 viene attivato.

Esempio di programma RIWT

Il programma seguente, eseguita dalla CPU del PLC della stazione master, scrive il valore „10“ all'indirizzo 111_H e il valore „20“ all'indirizzo 112_H del buffer di memoria di una stazione intelligente con numero di stazione 1. Il modulo master di CC-Link occupa gli indirizzi di I/O da X/Y000 a X/Y01F.



Per ulteriori informazioni sulla programmazione delle istruzioni dedicate con gli editor MELSEC, fare riferimento al capitolo 3.3 di questo manuale.

11.5.7 RIWT (serie QnA, System Q)

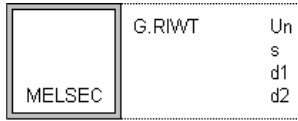
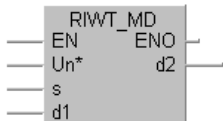
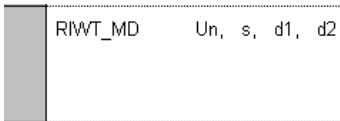
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

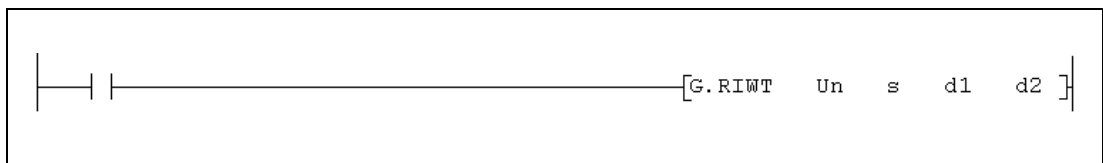
Operandi MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s	—	●	●	—	—	—	—	—	—	SM0	8
d1	—	●	●	—	—	—	—	—	—		
d2	●	●	●	—	—	—	—	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> 	<p>Schema a contatti</p> 	<p>Lista istruzioni IEC</p> 
---	--	--

GX Developer



Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati	
Un	Indirizzo iniziale di I/O del modulo master/locale CC-Link (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come 10 _H)	da 0 a FE _H	Utente	BIN 16-bit	
s	Indirizzo iniziale dell'area operandi in cui sono contenuti i dati di controllo dell'istruzione.				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(s)+0	Risultato dell'esecuzione dell'istruzione	Indica se si è verificato un errore durante l'esecuzione dell'istruzione: 0000 _H : Nessun errore Qualsiasi valore diverso da 0000 _H : Si è verificato un errore. Il valore memorizzato è un codice di errore descritto nel manuale utente del modulo CC-Link.	—	Sistema
	(s)+1	Numero stazione	Numero di stazione della stazione remota, in cui si scrivono i dati.	da 0 a 64	Utente
	(s)+2	Codice di accesso	<ul style="list-style-type: none"> Per moduli master serie A/Q con versione software da A a H. Impostare „0004_H“ per scrivere nel buffer di memoria di una stazione intelligente. Impostare „2004_H“ per scrivere nel buffer di memoria di una stazione locale. 	0004 _H o 2004 _H	
		Codice operando e codice di accesso	<ul style="list-style-type: none"> Per un modulo master con versione software J o superiore, o moduli System Q Gli 8 bit pesanti di questo operando contengono un codice operando. Il codice di accesso, che specifica se l'accesso è per il buffer di memoria di un modulo CC-Link (04 _H) o per una CPU (05 _H), è inserito negli 8 bit più leggeri.	Byte più pesante: vedi tabella seguente Byte più leggero: 04 _H o 05 _H	
	(s)+3	Indirizzo iniziale	<ul style="list-style-type: none"> Per moduli master serie A/Q con versione software da A a H. Indirizzo iniziale del buffer di memoria 	Dipende dalla stazione indirizzata	
<ul style="list-style-type: none"> Per un modulo master con versione software J o superiore, o moduli System Q Indirizzo iniziale del buffer di memoria o primo operando					
(s)+4	Lunghezza dati	Specificare il numero di dati (unità: word) da scrivere. Questo numero dipende dal tipo del modulo CPU montato sulla stazione in cui vengono scritti i dati: AnU, serie QnA, System Q: max. 480 word Tutte le altre CPU: max. 32 word	da 1 a 480 da 1 a 10		
d1	Indirizzo iniziale dell'area operandi dove vengono scritti i dati.		Utente	BIN 16-bit	

Variabili

Dati impostati	Significato			Campo	Contenuto inserito da	Tipo dati
d2	Operando a bit attivato per una scansione al termine dell'istruzione RIWT. (d2)+1 indica se si è verificato un errore durante l'esecuzione dell'istruzione.					Bit
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da	
	(d2)+0	Istruzione completata	Indica lo stato di completamento dell'istruzione RIWT ON: Istruzione completata OFF: Istruzione non completata	—	Sistema	
(d2)+1	Istruzione completata con errore	Indica l'occorrenza di un errore durante l'elaborazione dell'istruzione RIWT ON: Completamento anomalo OFF: Completamento normale	—			

A partire dalla versione J del software del modulo master, vengono utilizzati due codici (entrambi contenuti in (d1)+2) per specificare la destinazione dei dati. Il **codice di accesso** determina se i dati vengono scritti nel buffer di memoria di un modulo CC-Link oppure nella memoria del modulo CPU. Il **codice operando** indica l'area del buffer di memoria o degli operandi che vengono sovrascritti:

- Accesso al buffer di memoria di un modulo CC-Link (codice di accesso: 04_H)

Accesso a		Codice operando
Buffer di memoria di una stazione intelligente		00 _H
Buffer di memoria di una stazione master o locale	Buffer ad accesso casuale	20 _H
	Ingressi remoti	21 _H
	Uscite remote	22 _H
	Registri remoti	24 _H
	Relé speciali di comunicazione	63 _H
	Registri speciali di comunicazione	64 _H

- Accesso alla memoria di un modulo CPU (codice di accesso: 05_H)

Gli operandi non indicati nella tabella seguente non sono accessibili. Per accedere ad un operando a bit, specificare „0“ o un multiplo di „16“ come operando iniziale. Altrimenti viene generato un errore.

Operando		Tipo operando		Unità	Codice operando
Nome	Simbolo	Bit	Word		
Ingressi	X	●		Esadecimale	00 _H
Uscite	Y	●			02 _H
Relé interni	M	●		Decimale	03 _H
Relé retentivi	L	●			83 _H
Relé di comunicazione	B	●		Esadecimale	23 _H
Timer (contatto)	T	●		Decimale	09 _H
Timer (bobina)		●			0A _H
Timer (valore attuale)			●		0C _H
Timer retentivo (contatto)	ST	●			89 _H
Timer retentivo (bobina)		●			8A _H
Timer retentivo (valore attuale)			●		8C _H

Operando		Tipo operando		Unità	Codice operando
Nome	Simbolo	Bit	Word		
Contatore (contatto)	C	●		Decimale	11 _H
Contatore (bobina)		●			12 _H
Contatore (valore attuale)			●		14 _H
Registro dati	D		●		04 _H
Registro di comunicazione	W		●	Esadecimale	24 _H
File registri	R		●	Decimale	84 _H
Relé speciali di comunicazione	SB	●		Esadecimale	63 _H
Registri speciali di comunicazione	SW		●		64 _H
Relé speciale	SM	●		Decimale	43 _H
Registro speciale	SD		●		44 _H

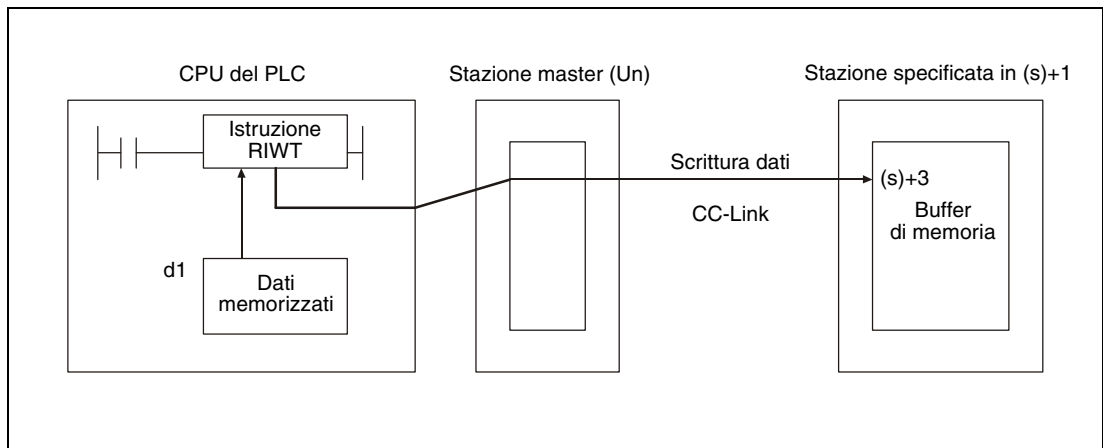
Funzioni **Scrittura nel buffer di memoria di una stazione intelligente o nella memoria della CPU del PLC**

RIWT Scrittura dati

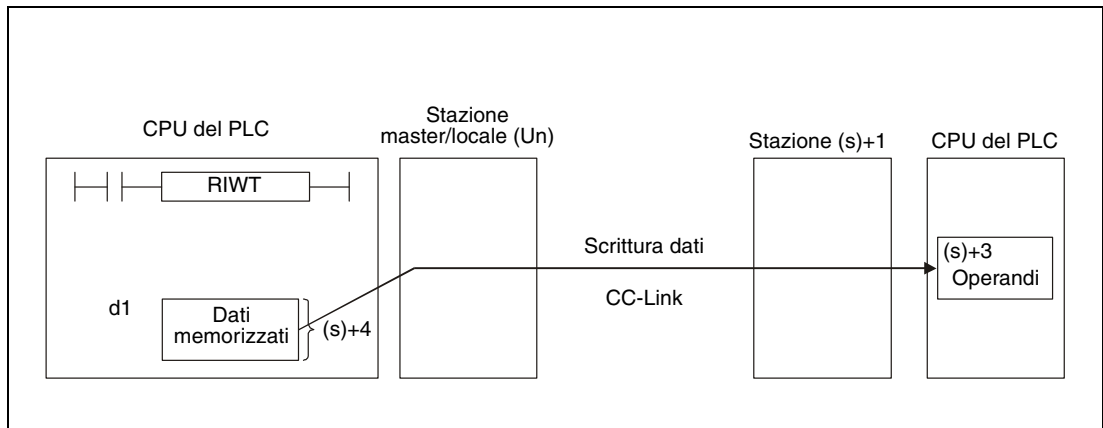
L'istruzione RIWT scrive dati nel buffer di memoria di una stazione intelligente collegata al CC-Link. Se viene usato un modulo master con versione software J o superiore, oppure un modulo CC-Link del System Q MELSEC, è possibile anche scrivere nella memoria della CPU del PLC di un'altra stazione collegata alla rete CC-Link.

Il numero di stazione della stazione partner è indicato da (s)+1. Questa stazione è collegata con la stazione master/locale specificata da Un. La destinazione dei dati è indicata da d1. L'operando (s)+2 contiene un codice che specifica se la scrittura interessa un buffer di memoria o la memoria operandi di un modulo CPU. L'indirizzo iniziale del buffer di memoria o il primo operando, viene specificato da (s)+3. Il numero dei dati da scrivere è indicato da (s)+4.

- Accesso al buffer di memoria di un modulo CC-Link



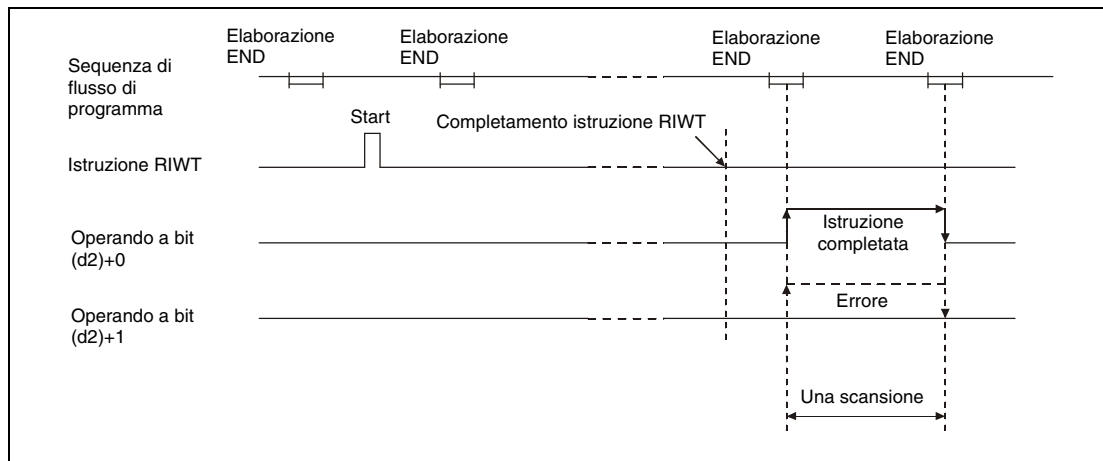
- Accesso alla memoria della CPU di un'altra stazione CC-Link



Lo stato di esecuzione dell'istruzione RIWT può essere controllato con gli operandi (d2)+0 e (d2)+1:

- L'operando a bit (d2)+0 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione RIWT, e si disattiva con l'elaborazione di END della scansione successiva.
- L'operando a bit (d2)+1 segnala la presenza di un errore durante l'esecuzione di una istruzione RIWT. Se l'istruzione termina normalmente, l'operando rimane OFF, ma se si verifica un errore durante l'esecuzione dell'istruzione RIWT, (d2)+1 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione RIWT, e si disattiva con l'elaborazione di END della scansione successiva.

La figura seguente mostra la temporizzazione durante l'esecuzione di una istruzione RIWT:



Notare che è possibile eseguire contemporaneamente istruzioni RIWT su più stazioni, ma la stessa stazione intelligente o locale non può essere utilizzata simultaneamente da più di una stazione.

Errori di esecuzione

Nei casi seguenti, si verifica un errore di funzionamento, viene attivato il flag di errore SM0, ed un codice di errore viene memorizzato in SD0:

- Se il modulo specificato da Un non è un modulo intelligente o un modulo funzione speciale. (codice di errore: 2112)
- Se si cerca di eseguire una istruzione non supportata (codice di errore: 4002). 4002)
- Se il numero di operandi dell'istruzione non è corretto. (codice di errore: 4003)
- Se l'istruzione specifica un operando che non si può usare. (error code: 4004)
- Se l'area specificata da s contiene dati che non si possono usare. (codice di errore: 4100)
- Se il numero di dati impostato supera il campo ammesso. (codice di errore: 4101)
- Se l'area di memoria o le costanti degli operandi specificati con l'istruzione superano il campo ammesso (codice di errore: 4101)

Esempio di programma RIWT

Il programma seguente viene eseguito nella CPU del PLC della stazione master. Quando l'ingresso X0 è attivo, il contenuto dei registri dati da D0 a D9 viene inviato verso la stazione intelligente numero 1 e memorizzato nel suo buffer di memoria, negli indirizzi da 100_H a 109_H. L'indirizzo iniziale di I/O del modulo master CC-Link è X/Y40.

- Editor IEC
(questo programma di esempio è mostrato a pagina seguente con lista istruzioni MELSEC e schema a contatti di GX Developer).

Schema a contatti (GX IEC Developer)

1 — s — EN — MOV P_M — ENO — d — D101

16#4 — s — EN — MOV P_M — ENO — d — D102

16#100 — s — EN — MOV P_M — ENO — d — D103

10 — s — EN — MOV P_M — ENO — d — D104

4 — Un* — EN — RIWT_P_MD — ENO — d2 — var_MD

var_D100 — s — D0 — d1

EN — SET_M — ENO — d — M100

Il numero di stazioni viene scritto in (s)+1

Il codice 4_H viene scritto in (s)+2 per accedere al buffer di memoria di una stazione intelligente.

L'indirizzo iniziale del buffer di memoria viene memorizzato in (s)+3.

Numero di punti da scrivere

Scrittura nel buffer di memoria

M100 viene attivato per indicare scrittura in corso.

M0 — LD — ANDN — M1 — / — EN — RST_M — ENO — d — M100

Inserire in questa posizione le istruzioni che devono essere eseguite quando l'istruzione RIWT termina correttamente.

Inserire in questa posizione le istruzioni che devono essere eseguite quando l'istruzione RIWT termina con errore.

Al termine della scrittura dei dati, M100 viene azzerato.

Lista istruzioni IEC

LD	X0		
ANDN	M100		
MOV P_M	1,	D101	
MOV P_M	16#4,	D102	
MOV P_M	16#100,	D103	
MOV P_M	10,	D104	
RIWT_P_MD	4,	var_D100, D0,	var_MD
SET_M	M100		

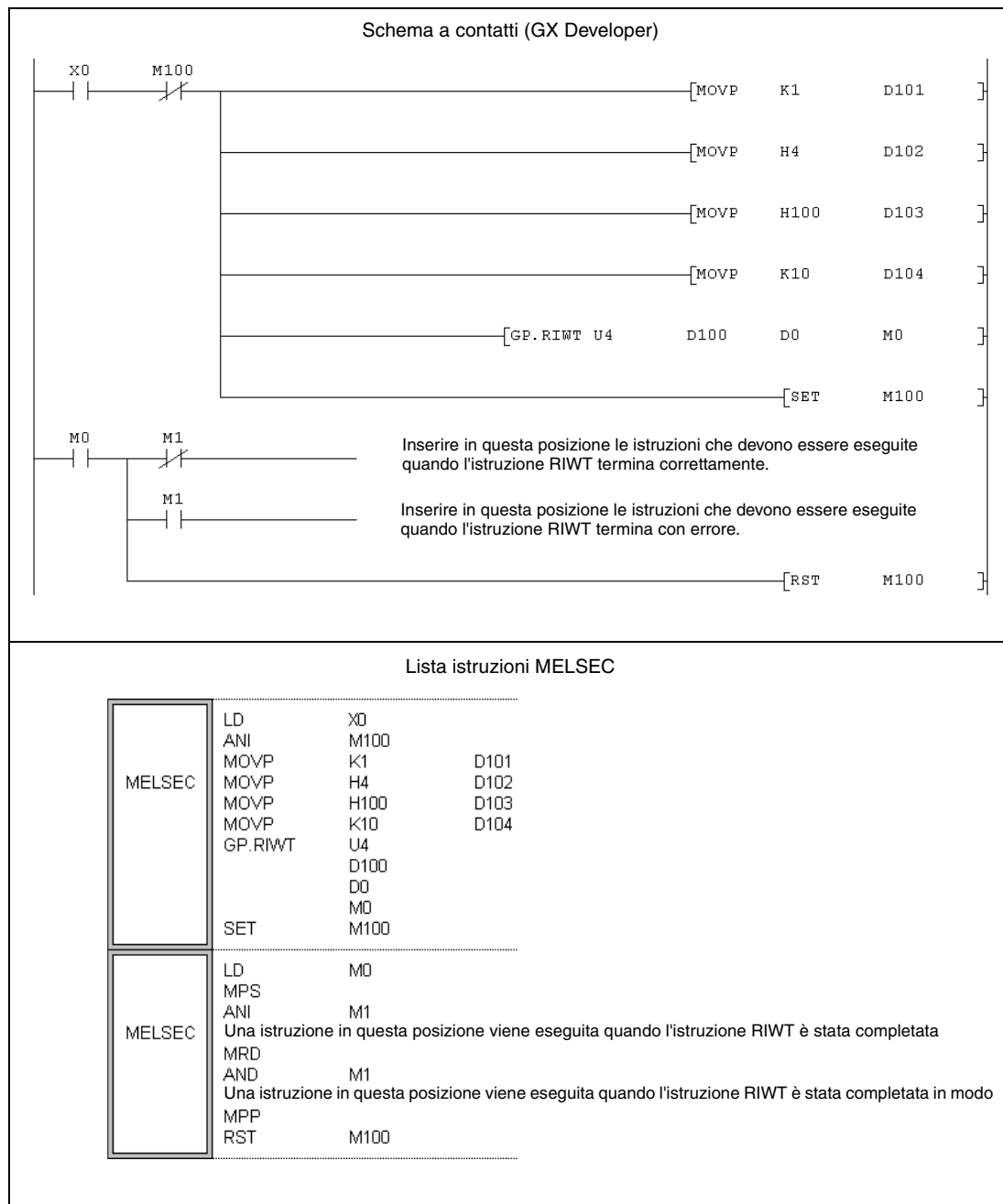
Per una spiegazione su operandi e istruzioni usate, fare riferimento al programma precedente.

LD	M0		
ANDN	M1		
Una istruzione in questa posizione viene eseguita quando l'istruzione RIWT è stata completata normalmente.			
LD	M0		
AND	M1		
Una istruzione in questa posizione viene eseguita quando l'istruzione RIWT è stata completata in modo anomalo.			
LD	M0		
RST_M	M100		

NOTA

Negli editor IEC è necessario definire le variabili nello header della unità organizzativa di programma (POU) La mancata definizione delle variabili provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

- Lista istruzioni MELSEC e schema a contatti di GX Developer
Per una spiegazione degli operandi e istruzioni utilizzate, vedere il programma di esempio per lo schema a contatti di GX IEC Developer nella pagina precedente.



11.5.8 RIRCV (serie A)

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●		

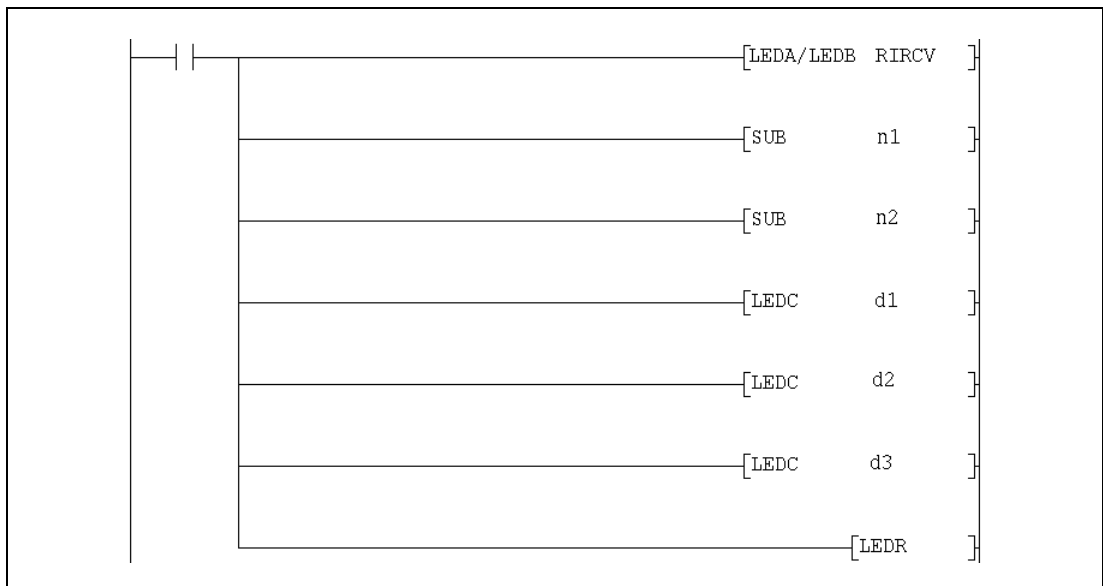
Operandi MELSEC A

	Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Rapporto	Errore			
	Operandi a bit						Operandi a word (16-bit)						Costanti		Puntatore							Livello		
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V						K	H (16#)	P
n1																●	●							
n2																●	●							
d1							●	●	●	●	●													●
d2							●	●	●	●	●													
d3	●	●	●	●	●																			

GX IEC Developer

<p style="text-align: center;">Lista istruzioni MELSEC</p> <table border="1" style="width: 100%;"> <tr> <td style="text-align: center;">MELSEC</td> <td>LEDA/LEDB</td> <td>RIRCV</td> </tr> <tr> <td></td> <td>SUB</td> <td>n1</td> </tr> <tr> <td></td> <td>SUB</td> <td>n2</td> </tr> <tr> <td></td> <td>LEDC</td> <td>d1</td> </tr> <tr> <td></td> <td>LEDC</td> <td>d2</td> </tr> <tr> <td></td> <td>LEDC</td> <td>d3</td> </tr> <tr> <td></td> <td>LEDR</td> <td></td> </tr> </table>	MELSEC	LEDA/LEDB	RIRCV		SUB	n1		SUB	n2		LEDC	d1		LEDC	d2		LEDC	d3		LEDR		<p style="text-align: center;">Schema a contatti</p>	<p style="text-align: center;">Lista istruzioni IEC</p> <table border="1" style="width: 100%;"> <tr> <td>RIRCV_MD</td> <td>n1, n2, d1, d2, d3</td> </tr> </table>	RIRCV_MD	n1, n2, d1, d2, d3
MELSEC	LEDA/LEDB	RIRCV																							
	SUB	n1																							
	SUB	n2																							
	LEDC	d1																							
	LEDC	d2																							
	LEDC	d3																							
	LEDR																								
RIRCV_MD	n1, n2, d1, d2, d3																								

GX Developer



Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati	
n1	Indirizzo iniziale di I/O del modulo master CC-Link (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come 10 _H)	da 0 a FE _H	Utente	BIN 16-bit	
n2	Numero di stazione della stazione intelligente da cui vengono letti i dati	da 1 a 64	Utente	BIN 16-bit	
d1	Indirizzo iniziale dell'area operandi in cui sono contenuti i dati di controllo dell'istruzione e vengono memorizzati i dati letti.				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(d1)+0	Risultato dell'esecuzione dell'istruzione	Indica se si è verificato un errore durante l'esecuzione dell'istruzione: 0000 _H : Nessun errore Qualsiasi valore diverso da 0000 _H : Si è verificato un errore. Il valore memorizzato è un codice di errore descritto nel manuale utente del modulo CC-Link.	—	Sistema
	(d1)+1	Numero di punti da leggere	Specifica quanti dati (in unità "word") devono essere letti dalla stazione intelligente. Impostare un valore compreso nei limiti di capacità del buffer di memoria della stazione intelligente e dell'area buffer per la ricezione dei parametri della stazione master.	da 1 a 480	Utente
	(d1)+2	Codice di accesso	Impostare il valore „0004 _H “ (lettura dal buffer di memoria di una stazione intelligente).	0004 _H	
	(d1)+3	Controllo errori	Specifica l'operando che indica la presenza di un errore durante l'esecuzione dell'istruzione RIRCV: 0: Operando d1 1: Operando RX+1	0 o 1	
	(d1)+4	Indirizzo iniziale	Indirizzo iniziale nel buffer di memoria (indirizzo del primo dato da leggere)	Dipende dalla stazione indirizzata	
(d1)+5 a (d1)+n	Area di memorizzazione dei dati letti	La dimensione di questa area è determinata dal numero di punti da leggere contenuto in (d1)+1.	—	Sistema	
d2	Operandi di comunicazione usati per lo handshaking				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(d2)+0	Ingressi remoti (RX) e uscite remote (RY)	<ul style="list-style-type: none"> Byte più pesante specifica un ingresso remoto (RX) della stazione intelligente Byte più leggero specifica un'uscita remota (RY) della stazione intelligente 	da 0 a 124 da 0 a 125	Utente (gli indirizzi RX, RY e RW _r vengono specificati dall'utente, ma vengono attivati e disattivati dal sistema).
(d2)+1	Registro remoto (RW _r)	Specifica un registro remoto (RW _r) della stazione intelligente	da 0 a 15 o FF (se impostato FF, nessun numero specificato).		

Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati		
d3	Operando a bit attivato per una scansione al termine dell'istruzione RIRCV. (d3)+1 indica se si è verificato un errore durante l'esecuzione dell'istruzione.				Bit	
	Dati impostati	Significato	Descrizione	Campo		Contenuto inserito da
	(d3)+0	Istruzione completata	Indica lo stato di completamento dell'istruzione RIRCV ON: Istruzione completata OFF: Istruzione non completata	—		Sistema
(d3)+1	Istruzione completata con errore	Indica l'occorrenza di un errore durante l'elaborazione dell'istruzione RIRCV ON: Completamento anomalo OFF: Completamento normale	—			

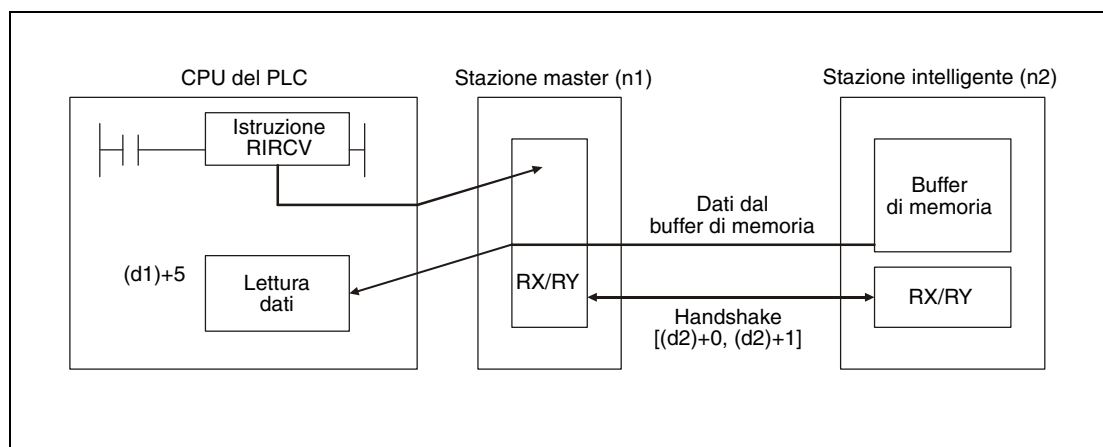
Funzioni

Letture dati dal buffer di memoria di una stazione intelligente (con handshake)**RIRCV Lettura dati (con handshake)**

L'esecuzione di una istruzione RIRCV è possibile solo sulla CPU del PLC di una stazione master. Questa istruzione viene usata per leggere dati dal buffer di memoria di una stazione intelligente. Lo scambio dati viene controllato da un operando di handshake.

Il numero di punti da leggere è contenuto in (d1)+1. L'indirizzo iniziale del buffer di memoria, specificato da (d1)+3, è il primo indirizzo da cui leggere. Il numero di stazione della stazione intelligente è specificato da n2. Questa stazione è collegata alla stazione master specificata da n1. I dati letti vengono memorizzati nella CPU che esegue l'istruzione RIRCV, nell'area operandi a partire da (d1)+5.

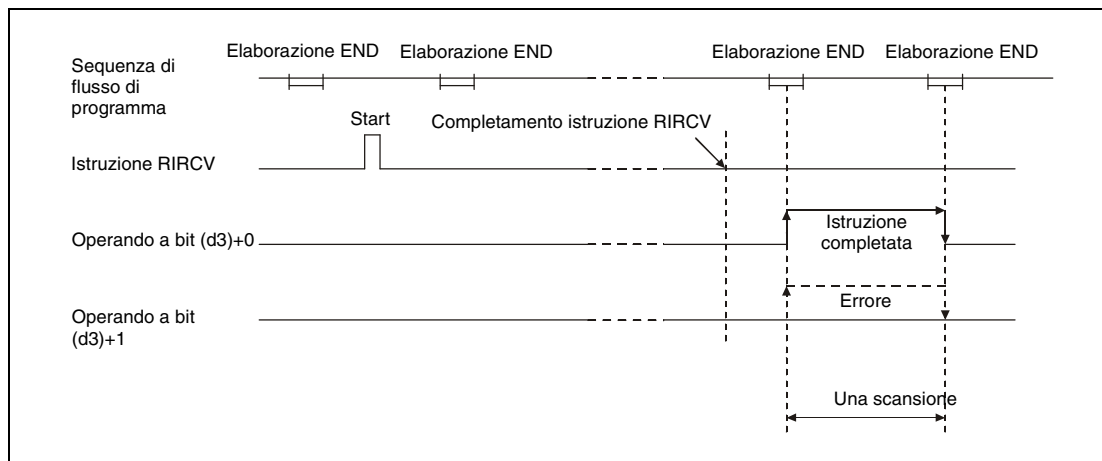
Funzionamento dell'istruzione RIRCV:



Lo stato di esecuzione dell'istruzione RIRCV può essere controllato con gli operandi (d3)+0 e (d3)+1:

- L'operando a bit (d3)+0 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione RIRCV, e si disattiva con l'elaborazione di END della scansione successiva.
- L'operando a bit (d3)+1 segnala la presenza di un errore durante l'esecuzione di una istruzione RIRCV. Se l'istruzione termina normalmente, l'operando rimane OFF. Se si verifica un errore durante l'esecuzione dell'istruzione RIRCV, (d3)+1 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione RIRCV, e si disattiva con l'elaborazione di END della scansione successiva.

La figura seguente mostra la temporizzazione durante l'esecuzione di una istruzione RIRCV:



Anche se è possibile eseguire contemporaneamente istruzioni RIRCV da più stazioni intelligenti, non è possibile accedere simultaneamente alla stessa stazione da più stazioni diverse.

Condizioni di esecuzione

Se viene usata l'istruzione LEDA, l'istruzione RIRCV viene eseguita ad ogni scansione in cui il comando di scrittura è ON.

Se viene usata l'istruzione LEDB, l'istruzione RIRCV viene eseguita per una scansione sul fronte positivo (OFF -> ON) del comando di scrittura.

Notare che il processo di lettura seguito dall'istruzione RIRCV richiede diverse scansioni prima del suo completamento. Eseguire quindi la successiva istruzione RIRCV solo dopo che l'operando di completamento (d3)+0 viene attivato. Una istruzione RIRCV non viene elaborata se l'esecuzione viene avviata mentre è in corso l'esecuzione di un'altra istruzione RIRCV.

Errori di esecuzione

Prima di eseguire l'istruzione RIRCV, impostare i parametri di rete con l'istruzione RLPA. Se viene eseguita l'istruzione RIRCV senza aver prima impostato i parametri, l'operando (d3)+1 viene attivato al termine dell'istruzione, mentre l'operando indicato da (d1)+0 contiene il codice di errore 4B00_H.

Se viene inserito „0“ o un valore non compreso fra 1 e 480 come numero di dati da leggere in (d1)+1, l'operando (d3)+1 viene attivato ed il codice di errore BB42_H viene scritto in (d1)+1 al completamento dell'istruzione RIRCV.

Esempio di programma RIRCV

Il programma seguente, eseguito nella CPU del PLC della stazione master, legge il contenuto delle locazioni da 400_H a 405_H del buffer di memoria di una stazione intelligente (stazione numero 1), mentre X20 è attivo. Lo handshake è realizzato con gli operandi RX2, RY2 e RWr2. Gli errori sono segnalati dall'operando indicato da (d1)+0. Il modulo master del CC-Link corrisponde all'indirizzo iniziale di I/O X/Y000.

- Editor IEC
(questo programma di esempio è mostrato a pagina seguente con lista istruzioni MELSEC e schema a contatti di GX Developer).

Schema a contatti (GX IEC Developer)

Il numero di punti da leggere (6 word) viene inserito in (d1)+1.

Il codice di accesso 4_H viene memorizzato in (d1)+2 per accedere al buffer di memoria di una stazione

Il codice „0” viene memorizzato in (d1)+3, mentre gli errori vengono segnalati dall'operando (d1)+0.

L'indirizzo iniziale nel buffer di memoria viene memorizzato in (d1)+4.

RX2 e RY2 vengono inseriti in (d2)+0.

RWr2 viene specificato in (d2)+1 come operando di handshake.

Letture del buffer di memoria

Il numero di punti da leggere (6 word) viene inserito in (d1)+1.

Il codice di accesso 4_H viene memorizzato in (d1)+2 per accedere al buffer di memoria di una stazione

Il codice „0” viene memorizzato in (d1)+3, mentre gli errori vengono segnalati dall'operando (d1)+0.

L'indirizzo iniziale nel buffer di memoria viene memorizzato in (d1)+4.

RX2 e RY2 vengono inseriti in (d2)+0.

RWr2 viene specificato in (d2)+1 come operando di handshake.

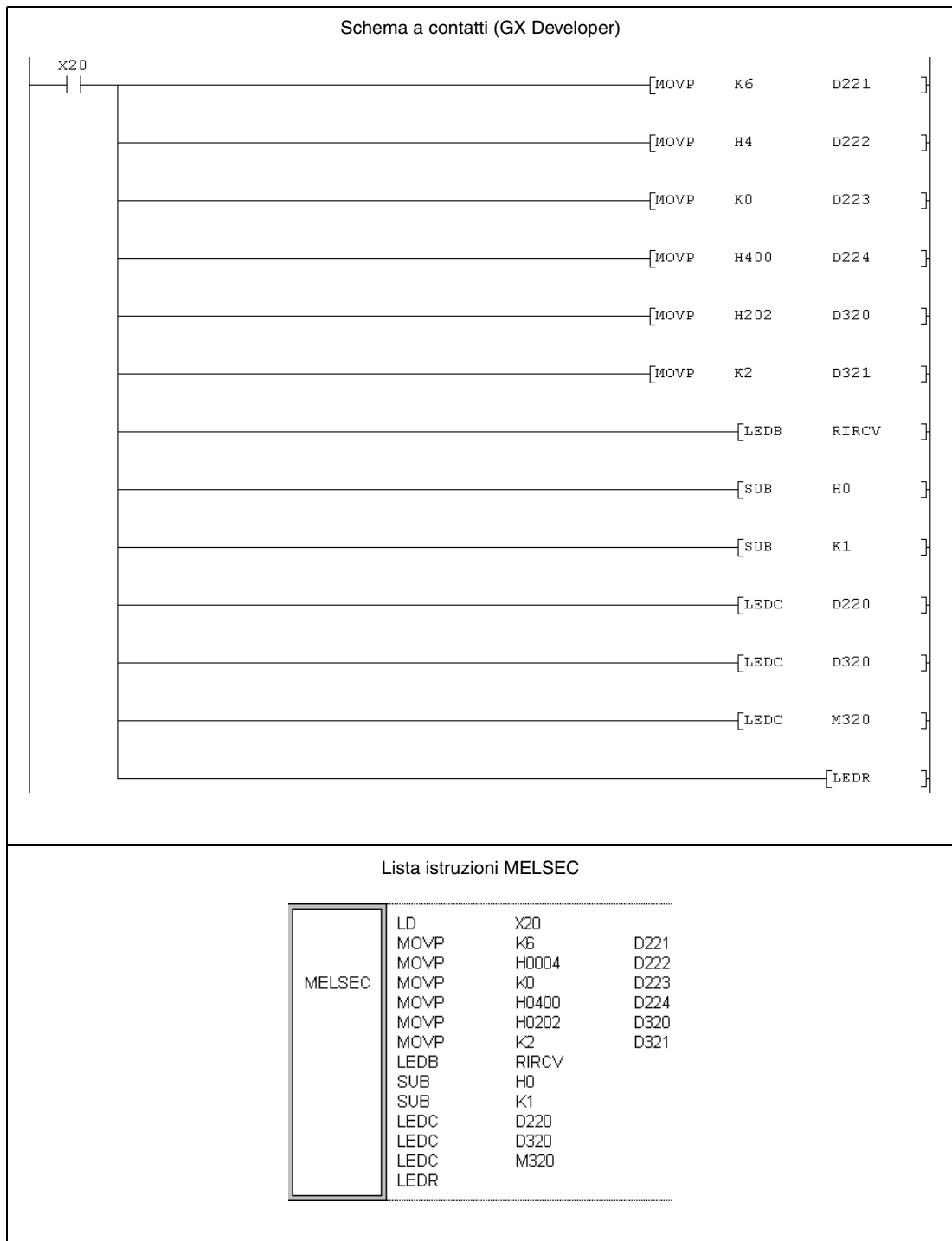
Letture del buffer di memoria

Lista istruzioni IEC

LD	X20								
MOV P_M	6,	D221							
MOV P_M	16#0004,	D222							
MOV P_M	0,	D223							
MOV P_M	16#0400,	D224							
MOV P_M	16#0202,	D320							
MOV P_M	2,	D321							
RIRCV_P_MD	0,	1,	D220,	D320,	M320				

Per una spiegazione su operandi e istruzioni usate, fare riferimento al programma precedente.

- Lista istruzioni MELSEC e schema a contatti di GX Developer
Per una spiegazione degli operandi e istruzioni utilizzate, vedere il programma di esempio per lo schema a contatti di GX IEC Developer nella pagina precedente.



Per ulteriori informazioni sulla programmazione delle istruzioni dedicate con gli editor MELSEC, fare riferimento al capitolo 3.3 di questo manuale.

11.5.9 RIRCV (serie QnA, System Q)

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

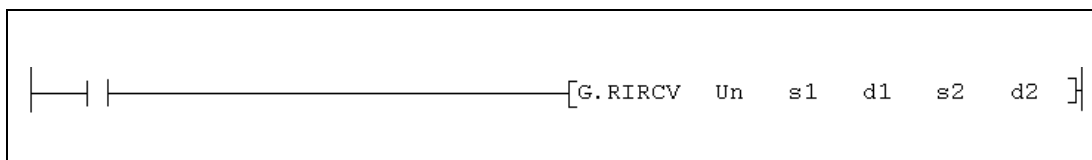
Operandi MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	—	SM0	10
s1	—	●	●	—	—	—	—	—	—		
d1	—	●	●	—	—	—	—	—	—		
d2	●	●	●	—	—	—	—	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> <table border="1" style="width: 100%;"> <tr> <td style="text-align: center; width: 20px; height: 50px;">MELSEC</td> <td style="padding-left: 10px;">G.RIRCV</td> <td style="padding-left: 20px;">Un s1 d1 s2 d2</td> </tr> </table>	MELSEC	G.RIRCV	Un s1 d1 s2 d2	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p> <table border="1" style="width: 100%;"> <tr> <td style="width: 20px; height: 40px;"></td> <td style="padding-left: 10px;">RIRCV_MD</td> <td style="padding-left: 20px;">Un, s1, s2, d1, d2</td> </tr> </table>		RIRCV_MD	Un, s1, s2, d1, d2
MELSEC	G.RIRCV	Un s1 d1 s2 d2						
	RIRCV_MD	Un, s1, s2, d1, d2						

GX Developer



Variabili

Dati impostati	Significato		Campo	Contenuto inserito da	Tipo dati
Un	Indirizzo iniziale di I/O del modulo master CC-Link (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come 10 _H)		da 0 a FE _H	Utente	BIN 16-bit
s1	Indirizzo iniziale dell'area operandi in cui sono contenuti i dati di controllo dell'istruzione.				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(s1)+0	Risultato dell'esecuzione dell'istruzione	Indica se si è verificato un errore durante l'esecuzione dell'istruzione: 0000 _H : Nessun errore Qualsiasi valore diverso da 0000 _H : Si è verificato un errore. Il valore memorizzato è un codice di errore descritto nel manuale utente del modulo CC-Link.	—	Sistema
	(s1)+1	Numero stazione	Numero di stazione della stazione intelligente da cui vengono letti i dati	da 0 a 64	Utente
	(s1)+2	Codice di accesso	Impostare il valore „0004 _H “ (lettura dal buffer di memoria di una stazione intelligente).	0004 _H	
(s1)+3	Indirizzo iniziale	Indirizzo iniziale nel buffer di memoria (indirizzo del primo dato da leggere)	Dipende dalla stazione indirizzata		
(s1)+4	Numero di punti da leggere	Specifica quanti dati (in unità "word") devono essere letti dalla stazione intelligente. Impostare un valore compreso nei limiti di capacità del buffer di memoria della stazione intelligente e dell'area buffer per la ricezione dei parametri della stazione master.	da 1 a 480		
s2	Operandi di comunicazione usati per lo handshaking				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(s2)+0	Uscita remota (RY) per richiesta dati	• Byte più pesante Imposta a „0“ gli 8 bit più significativi.	0	Utente
			• Byte più leggero Specifica un'uscita remota (RY) della stazione intelligente	da 0 a 127	
(s2)+1	Registro remoto (RW _r) usato come operando per la memorizzazione del codice di errore Ingresso remoto (RX) usato come operando di completamento.	• Byte più pesante Specifica un registro remoto (RW _r) della stazione intelligente, in cui viene inserito lo stesso codice di errore contenuto in (s1)+0.	da 0 a 15 o FF (se impostato FF, nessun numero specificato).		
		• Byte più leggero Specifica un ingresso remoto (RX) della stazione intelligente	da 0 a 127		
(s2)+2	Modo completamento	Specifica come deve essere indicato il completamento del processo di lettura: 0: tramite 1 operando (RX _n) 1: tramite 2 operandi (RX _n , RX _{n+1}) (RX _{n+1} viene attivato in caso di errore).	0 o 1		
d1	Indirizzo iniziale dell'area operandi che memorizza i dati letti.			Utente	BIN 16-bit

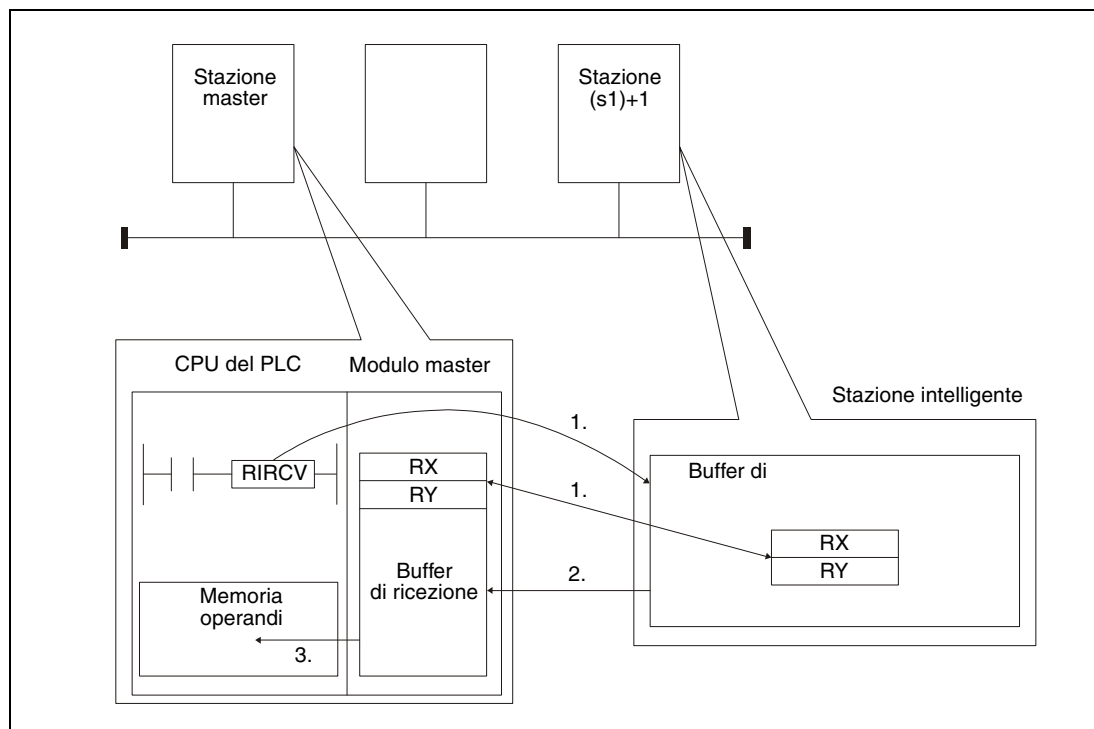
Variabili

Dati impostati	Significato			Campo	Contenuto inserito da	Tipo dati
d2	Operando a bit attivato per una scansione al termine dell'istruzione RIRCV. (d2)+1 indica se si è verificato un errore durante l'esecuzione dell'istruzione.					Bit
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da	
	(d2)+0	Istruzione completata	Indica lo stato di completamento dell'istruzione RIRCV ON: Istruzione completata OFF: Istruzione non completata	—	Sistema	
(d2)+1	Istruzione completata con errore	Indica l'occorrenza di un errore durante l'elaborazione dell'istruzione RIRCV ON: Completamento anomalo OFF: Completamento normale	—			

Funzioni

Letture dati dal buffer di memoria di una stazione intelligente (con handshake)**RIRCV Lettura dati (con handshake)**

L'esecuzione di una istruzione RIRCV è possibile solo sulla CPU del PLC di una stazione master. Questa istruzione viene usata per leggere dati dal buffer di memoria di una stazione intelligente. Lo scambio dati viene controllato da operandi di handshake.



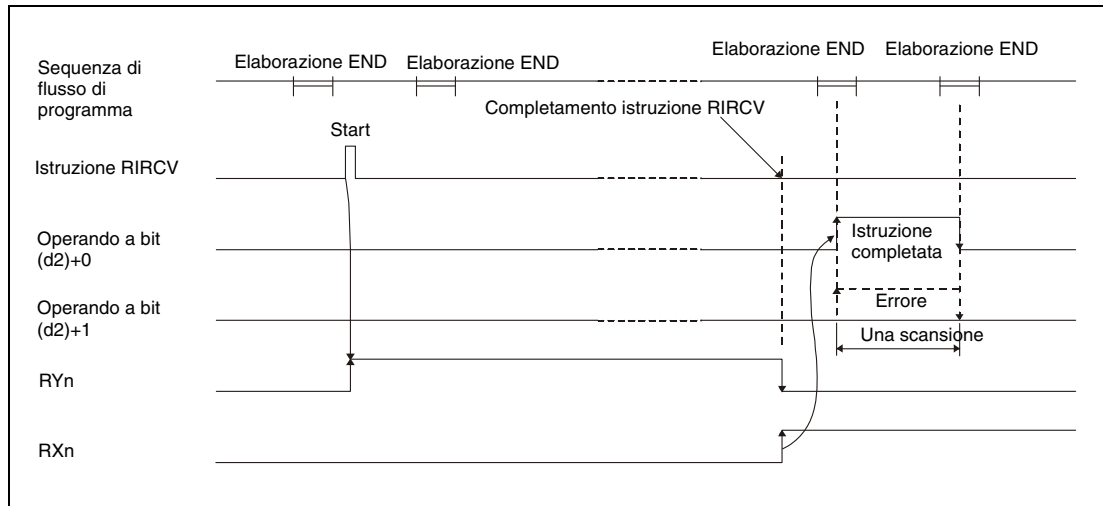
1. L'indirizzo del buffer di memoria specificato da (s1)+3 della stazione indicata da (s1)+1 viene aperto. Gli operandi specificati in s2 vengono utilizzati per lo handshake.
2. Il contenuto del numero di locazioni del buffer di memoria indicato in (s1)+4 viene letto nel buffer di ricezione del modulo master.
3. I dati letti vengono memorizzati nella CPU del PLC nell'area operandi a partire da d1. Successivamente, l'operando a bit specificato in (d2)+0 viene attivato per una scansione.

Lo stato di esecuzione dell'istruzione RIRCV può essere controllato con gli operandi (d2)+0 e (d2)+1:

- L'operando a bit (d2)+0 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione RIRCV, e si disattiva con l'elaborazione di END della scansione successiva.

- L'operando a bit (d2)+1 segnala la presenza di un errore durante l'esecuzione della istruzione RIRCV. Se l'istruzione termina normalmente, l'operando rimane OFF. Se si verifica un errore durante l'esecuzione dell'istruzione RIRCV, (d2)+1 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione RIRCV, e si disattiva con l'elaborazione di END della scansione successiva.

La figura seguente mostra la temporizzazione durante l'esecuzione di una istruzione RIRCV:



Anche se è possibile eseguire contemporaneamente istruzioni RIRCV da più stazioni intelligenti, non è possibile accedere simultaneamente alla stessa stazione da più stazioni diverse.

Errori di esecuzione

Nei casi seguenti, si verifica un errore di funzionamento, viene attivato il flag di errore SM0, ed un codice di errore viene memorizzato in SD0:

- Se il modulo specificato da Un non è un modulo intelligente o un modulo funzione speciale. (codice di errore: 2112)
- Se si cerca di eseguire una istruzione non supportata (codice di errore: 4002). 4002)
- Se il numero di operandi dell'istruzione non è corretto. (codice di errore: 4003)
- Se l'istruzione specifica un operando che non si può usare. (codice di errore: 4004)
- Se l'area specificata da s contiene dati che non si possono usare. (codice di errore: 4100)
- Se il numero di dati impostato supera il campo ammesso. (codice di errore: 4101)
- Se l'area di memoria o le costanti degli operandi specificati con l'istruzione superano il campo ammesso (codice di errore: 4101)

Esempio di programma

RIRCV

Il programma seguente viene eseguito nella CPU del PLC della stazione master. Quando si attiva M1, il contenuto di 11 locazioni del buffer di memoria viene letto dalla stazione intelligente con il numero di stazione 63, a partire dall'indirizzo 400_H del buffer di memoria. I dati vengono memorizzati nel buffer di memoria a partire dal registro dati D40. Il modulo master CC-Link corrisponde all'indirizzo iniziale di I/O X/Y00. Lo handshake è realizzato con gli operandi remoti RX2, RY2 e RWr2. Il completamento della lettura è indicato da due operandi. ((s2)+2 è impostato a „1“.)

● Editor IEC

(questo programma di esempio è mostrato a pagina seguente con lista istruzioni MELSEC e schema a contatti di GX Developer).

Schema a contatti (GX IEC Developer)

Lista istruzioni IEC

LD	M1			
MOV P_M	63,	D11		
MOV P_M	16#0004,	D12		
MOV P_M	16#0400,	D13		
MOV P_M	11,	D14		
MOV P_M	16#0002,	D15		
MOV P_M	16#0202,	D16		
MOV P_M	16#0001,	D17		
RIRCV_P_MD	0,	var_D10, var_d15,	D40, var_M20	

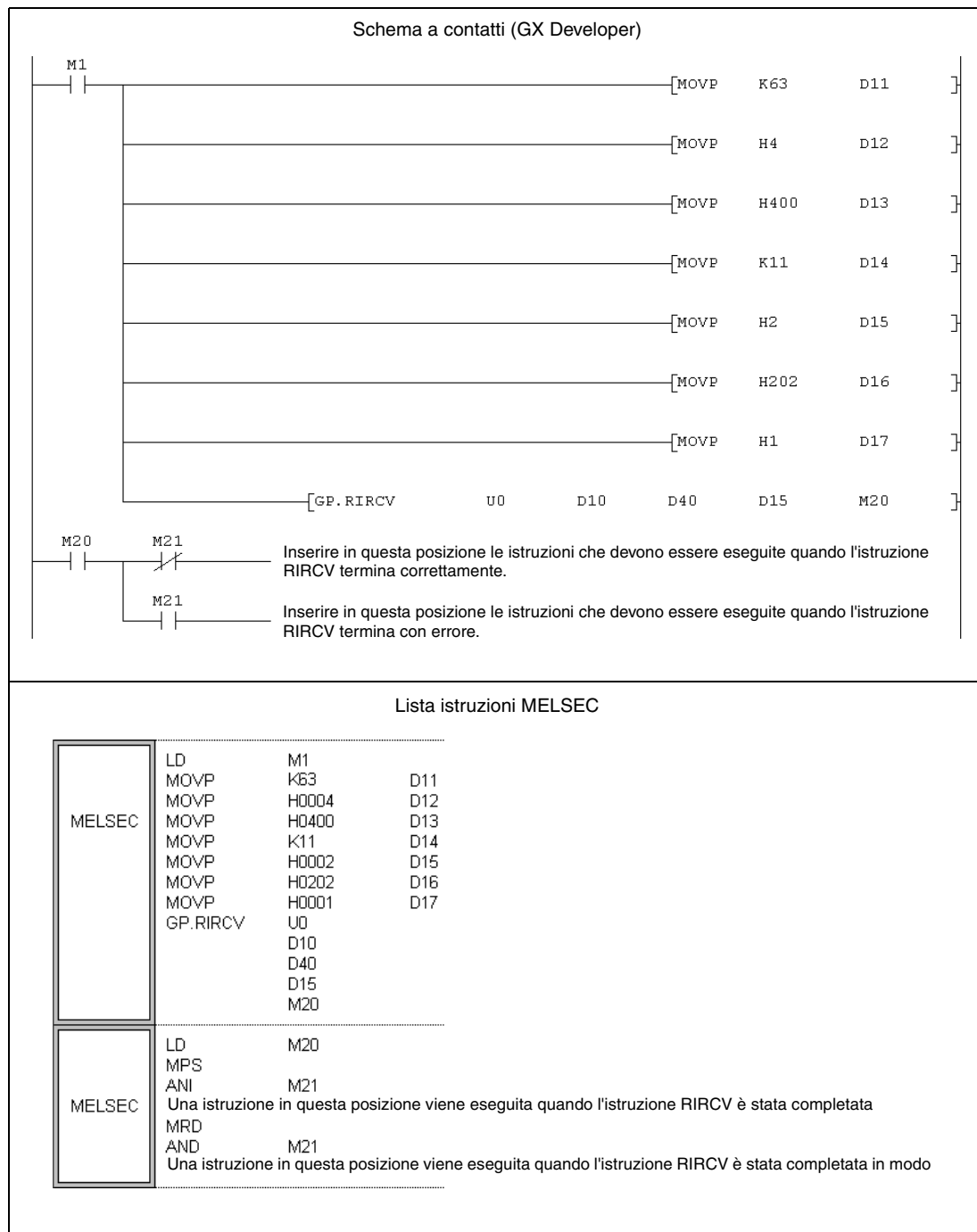
Per una spiegazione su operandi e istruzioni usate, fare riferimento al programma precedente.

LD	M20		
ANDN	M21		
Una istruzione in questa posizione viene eseguita quando l'istruzione RIRCV è stata completata normalmente.			
LD	M20		
AND	M21		
Una istruzione in questa posizione viene eseguita quando l'istruzione RIRCV è stata completata in modo anomalo.			

NOTA

Negli editor IEC è necessario definire le variabili nello header della unità organizzativa di programma (POU) La mancata definizione delle variabili provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

- Lista istruzioni MELSEC e schema a contatti di GX Developer
Per una spiegazione degli operandi e istruzioni utilizzate, vedere il programma di esempio per lo schema a contatti di GX IEC Developer nella pagina precedente.



11.5.10 RISEND (serie A)

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●		

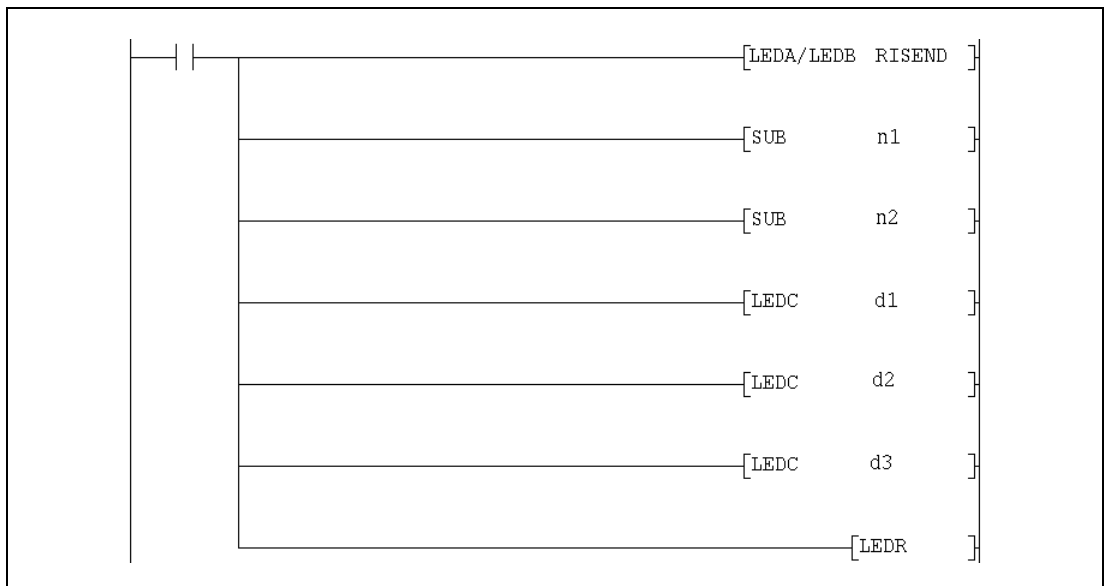
Operandi MELSEC A

	Operandi utilizzabili																Definizione cifra	Numero di passi	Indice	Rapporto	Errore			
	Operandi a bit						Operandi a word (16-bit)						Costanti		Puntatore							Livello		
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V						K	H (16#)	P
n1																●	●							
n2																●	●							
d1							●	●	●	●	●											29		●
d2							●	●	●	●	●													
d3	●	●	●	●	●																			

GX IEC Developer

<p style="text-align: center;">Lista istruzioni MELSEC</p> <table border="1" style="width: 100%;"> <tr> <td style="text-align: center; width: 15%;">MELSEC</td> <td style="width: 35%;">LEDA/LEDB</td> <td style="width: 50%;">RISEND</td> </tr> <tr> <td></td> <td>SUB</td> <td>n1</td> </tr> <tr> <td></td> <td>SUB</td> <td>n2</td> </tr> <tr> <td></td> <td>LEDC</td> <td>d1</td> </tr> <tr> <td></td> <td>LEDC</td> <td>d2</td> </tr> <tr> <td></td> <td>LEDC</td> <td>d3</td> </tr> <tr> <td></td> <td>LEDR</td> <td></td> </tr> </table>	MELSEC	LEDA/LEDB	RISEND		SUB	n1		SUB	n2		LEDC	d1		LEDC	d2		LEDC	d3		LEDR		<p style="text-align: center;">Schema a contatti</p>	<p style="text-align: center;">Lista istruzioni IEC</p> <table border="1" style="width: 100%;"> <tr> <td style="width: 30%;">RISEND_MD</td> <td style="width: 70%;">n1, n2, d1, d2, d3</td> </tr> </table>	RISEND_MD	n1, n2, d1, d2, d3
MELSEC	LEDA/LEDB	RISEND																							
	SUB	n1																							
	SUB	n2																							
	LEDC	d1																							
	LEDC	d2																							
	LEDC	d3																							
	LEDR																								
RISEND_MD	n1, n2, d1, d2, d3																								

GX Developer



Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati	
n1	Indirizzo iniziale di I/O del modulo master CC-Link (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come 10 _H)	da 0 a FE _H	Utente	BIN 16-bit	
n2	Numero di stazione della stazione intelligente in cui vengono scritti i dati	da 1 a 64	Utente	BIN 16-bit	
d1	Indirizzo iniziale dell'area operandi in cui sono contenuti i dati di controllo dell'istruzione e i dati da scrivere.				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(d1)+0	Risultato dell'esecuzione dell'istruzione	Indica se si è verificato un errore durante l'esecuzione dell'istruzione: 0000 _H : Nessun errore Qualsiasi valore diverso da 0000 _H : Si è verificato un errore. Il valore memorizzato è un codice di errore descritto nel manuale utente del modulo CC-Link.	—	Sistema
	(d1)+1	Numero di punti da scrivere	Specifica la quantità dei dati (unità: word) che devono essere scritte nella stazione intelligente.	da 1 a 480	Utente
	(d1)+2	Codice di accesso	Impostare il valore „0004 _H “ (scrittura nel buffer di memoria di una stazione intelligente).	0004 _H	
	(d1)+3	Controllo errori	Specifica l'operando che indica la presenza di un errore durante l'esecuzione dell'istruzione RISEND: 0: Operando d1 1: Operando RX+1	0 o 1	
	(d1)+4	Indirizzo iniziale	Indirizzo iniziale nel buffer di memoria (primo indirizzo in cui vengono scritti i dati).	Dipende dalla stazione indirizzata	
(d1)+5 a (d1)+n	Area contenente i dati da scrivere	La dimensione di questa area è determinata dal numero di punti da scrivere contenuto in (d1)+1.	—		
d2	Operandi di comunicazione usati per lo handshaking				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(d2)+0	Ingressi remoti (RX) e uscite remote (RY)	<ul style="list-style-type: none"> Byte più pesante specifica un ingresso remoto (RX) della stazione intelligente Byte più leggero specifica un'uscita remota (RY) della stazione intelligente 	da 0 a 127	Utente (gli indirizzi RX, RY e RW _r vengono specificati dall'utente, ma vengono attivati e disattivati dal sistema).
(d2)+1	Registro remoto (RW _r)	Specifica un registro remoto (RW _r) della stazione intelligente	da 0 a 15 o FF (se impostato FF, nessun numero specificato).		

Variabili

Dati impostati	Significato			Campo	Contenuto inserito da	Tipo dati
d3	Operando a bit attivato per una scansione al termine dell'istruzione RISEND. (d3)+1 indica se si è verificato un errore durante l'esecuzione dell'istruzione.					Bit
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da	
	(d3)+0	Istruzione completata	Indica lo stato di completamento dell'istruzione RISEND ON: Istruzione completata OFF: Istruzione non completata	—	Sistema	
(d3)+1	Istruzione completata con errore	Indica l'occorrenza di un errore durante l'elaborazione dell'istruzione RISEND ON: Completamento anomalo OFF: Completamento normale	—			

Funzioni

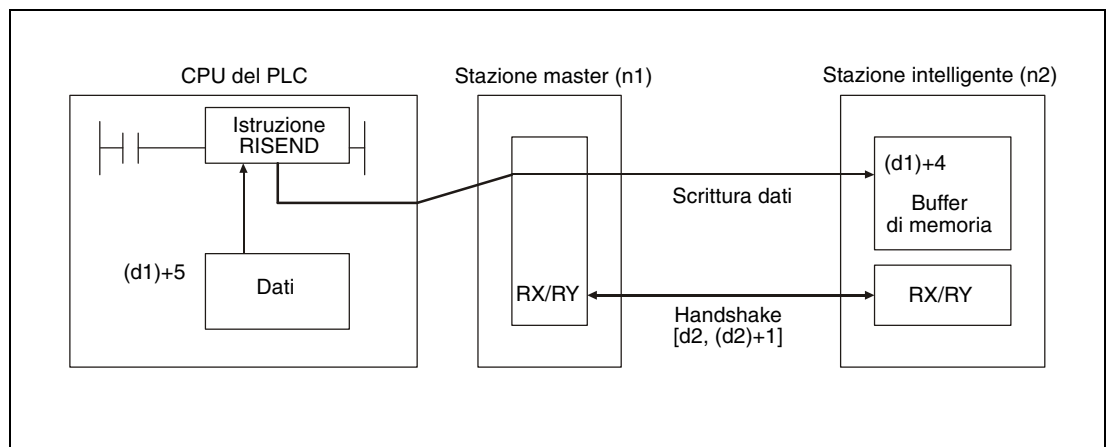
Scrittura (con handshake) nel buffer di memoria di una stazione intelligente

RISEND Invio dati (con handshake)

L'esecuzione di una istruzione RISEND è possibile solo sulla CPU del PLC di una stazione master. Questa istruzione viene usata per scrivere dati nel buffer di memoria di una stazione intelligente. Lo scambio dati viene controllato da un operando di handshake.

Il numero di punti da scrivere è contenuto in (d1)+1. L'indirizzo iniziale del buffer di memoria, specificato da (d1)+3, è il primo indirizzo in cui scrivere. Il numero di stazione della stazione intelligente è specificato da n2. Questa stazione è collegata alla stazione master specificata da n1. I dati da scrivere sono contenuto nella CPU che esegue l'istruzione RISEND, nell'area operandi che inizia da (d1)+5.

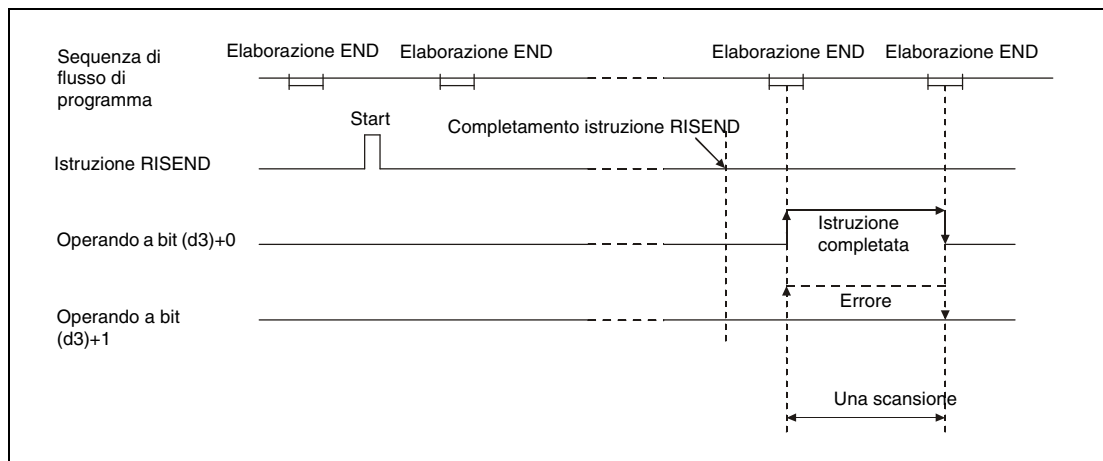
Funzionamento dell'istruzione RISEND:



Lo stato di esecuzione dell'istruzione RISEND può essere controllato con gli operandi (d3)+0 e (d3)+1:

- L'operando a bit (d3)+0 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione RISEND, e si disattiva con l'elaborazione di END della scansione successiva.
- L'operando a bit (d3)+1 segnala la presenza di un errore durante l'esecuzione di una istruzione RISEND. Se l'istruzione termina normalmente, l'operando rimane OFF. Se si verifica un errore durante l'esecuzione dell'istruzione RISEND, (d3)+1 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione RISEND, e si disattiva con l'elaborazione di END della scansione successiva.

La figura seguente mostra la temporizzazione durante l'esecuzione di una istruzione RISEND:



Anche se è possibile eseguire contemporaneamente istruzioni RISEND da più stazioni intelligenti, non è possibile accedere simultaneamente alla stessa stazione da più stazioni diverse.

Condizioni di esecuzione

Se viene usata l'istruzione LEDA, l'istruzione RISEND viene eseguita ad ogni scansione in cui il comando di scrittura è ON.

Se viene usata l'istruzione LEDB, l'istruzione RISEND viene eseguita per una scansione sul fronte positivo (OFF -> ON) del comando di lettura.

Notare che il processo di scrittura seguito dall'istruzione RISEND richiede diverse scansioni prima del suo completamento. Eseguire quindi la successiva istruzione RISEND solo dopo che l'operando di completamento (d3)+0 viene attivato. (Una istruzione RISEND non viene elaborata se l'esecuzione viene avviata mentre è in corso l'esecuzione di un'altra istruzione RISEND.

Prima di eseguire l'istruzione RISEND, impostare i parametri di rete con l'istruzione RLPA.

Errori di esecuzione

Se viene inserito „0“ o un valore non compreso fra 1 e 480 come numero di dati da scrivere in (d1)+1, l'operando (d3)+1 viene attivato ed il codice di errore BB42_H viene scritto in (d1)+1 al completamento dell'istruzione RISEND.

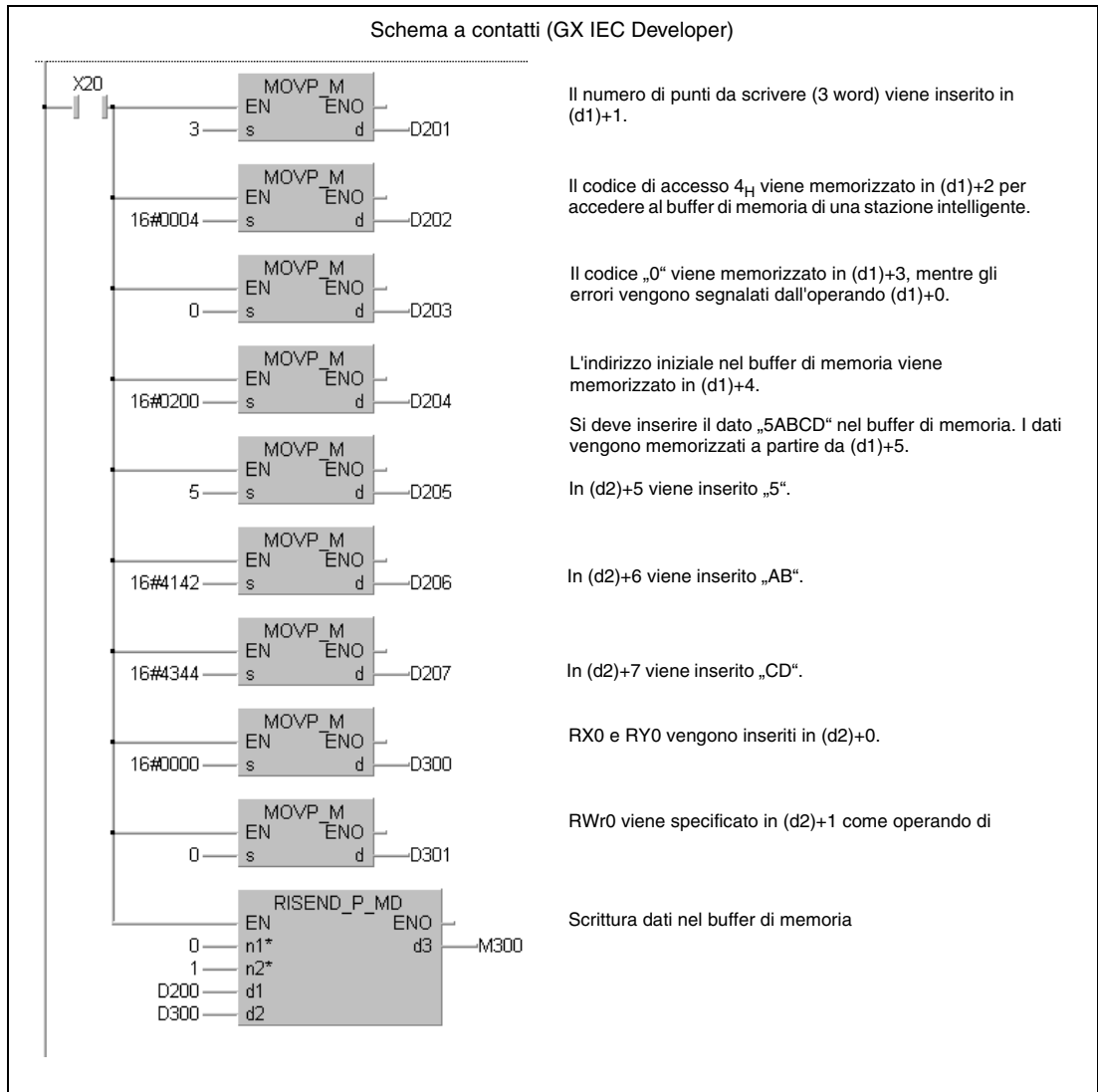
Esempio di programma

RISEND

Il Programma seguente, che viene eseguito nella CPU del PLC della stazione master, scrive nelle locazioni da 200_H a 202_H del buffer di memoria della stazione intelligente con numero di stazione 1. Per lo handshaking vengono utilizzati gli operandi RX0, RY0 e RWr0. Gli errori sono segnalati dall'operando indicato da (d1)+0. Il modulo master del CC-Link corrisponde all'indirizzo iniziale di I/O X/Y000.

● Editor IEC

(questo programma di esempio è mostrato a pagina seguente con lista istruzioni MELSEC e schema a contatti di GX Developer).

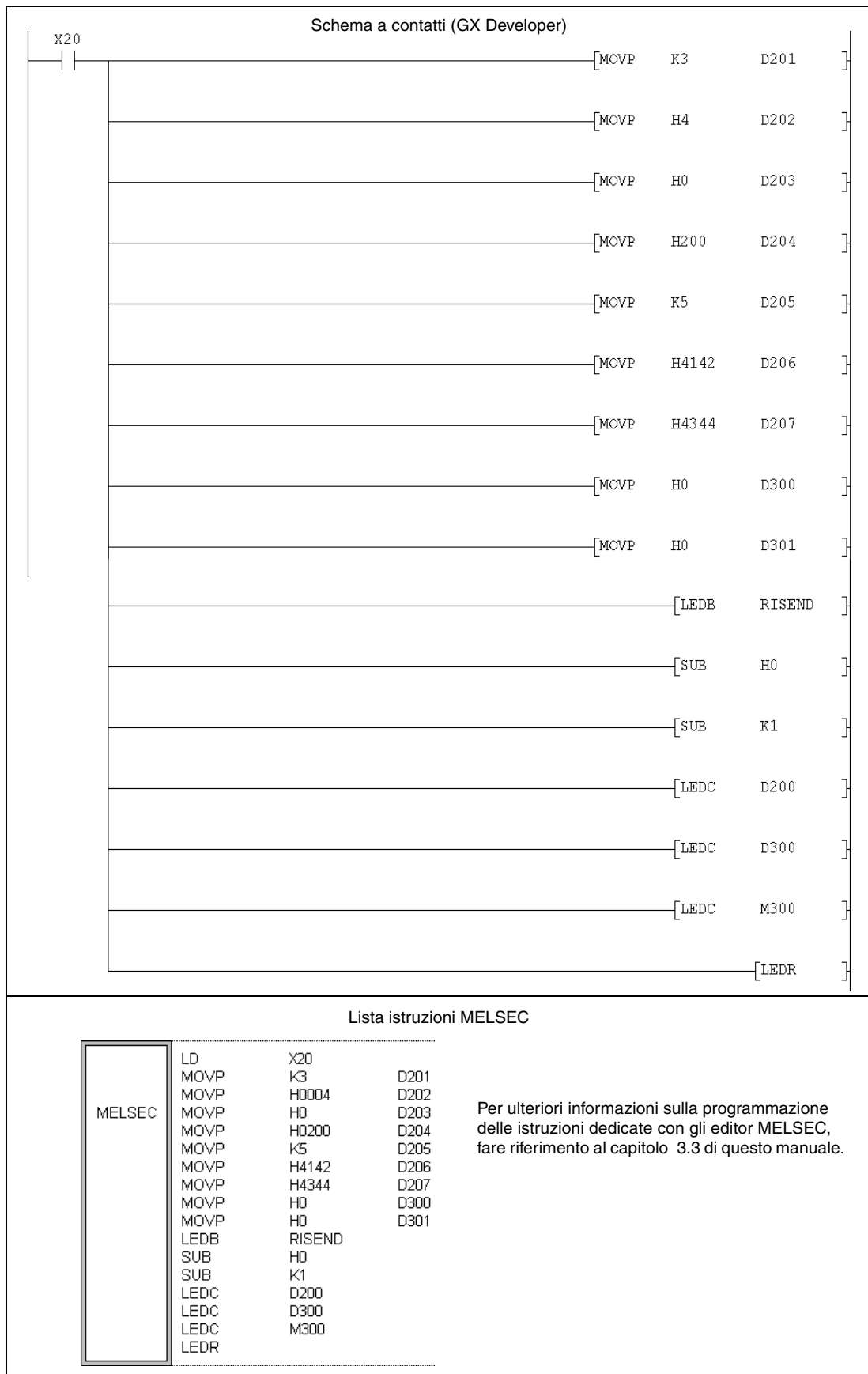


Lista istruzioni IEC

LD	X20			
MOV P M	2,	D201		
MOV P M	16#0004,	D202		
MOV P M	16#0000,	D203		
MOV P M	16#0200,	D204		
MOV P M	5,	D205		
MOV P M	16#4142,	D206		
MOV P M	16#4344,	D207		
MOV P M	16#0000,	D300		
MOV P M	0,	D301		
RISEND_P_MD	0,	1,	D200,	D300,
				M300

Per una spiegazione su operandi e istruzioni usate, fare riferimento al programma precedente.

- Lista istruzioni MELSEC e schema a contatti di GX Developer
Per una spiegazione degli operandi e istruzioni utilizzate, vedere il programma di esempio per lo schema a contatti di GX IEC Developer nella pagina precedente.



11.5.11 RISEND (serie QnA e System Q)

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

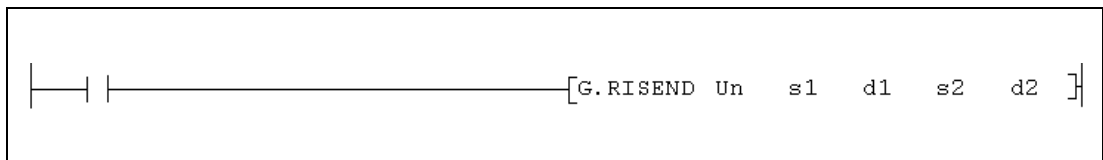
Operandi MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
s1	—	●	●	—	—	—	—	—	—	SM0	10
s2	—	●	●	—	—	—	—	—	—		
d1	—	●	●	—	—	—	—	—	—		
d2	●	●	●	—	—	—	—	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p>MELSEC</p> </div> <p>G.RISEND Un s1 d1 s2 d2</p>	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p>RISEND_MD Un, s1, d1, s2, d2</p> </div>
---	--------------------------	---

GX Developer



Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati	
Un	Indirizzo iniziale di I/O del modulo master CC-Link (Le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come 10 _H)	da 0 a FE _H	Utente	BIN 16-bit	
s1	Indirizzo iniziale dell'area operandi in cui sono contenuti i dati di controllo dell'istruzione.				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(s1)+0	Risultato dell'esecuzione dell'istruzione	Indica se si è verificato un errore durante l'esecuzione dell'istruzione: 0000 _H : Nessun errore Qualsiasi valore diverso da 0000 _H : Si è verificato un errore. Il valore memorizzato è un codice di errore descritto nel manuale utente del modulo CC-Link.	—	Sistema
	(s1)+1	Numero stazione	Numero di stazione della stazione intelligente a cui vengono inviati i dati	da 0 a 64	Utente
	(s1)+2	Codice di accesso	Impostare il valore „0004 _H “ (scrittura nel buffer di memoria di una stazione intelligente).	0004 _H	
(s1)+3	Indirizzo iniziale	Indirizzo iniziale nel buffer di memoria (primo indirizzo in cui vengono scritti i dati)	Dipende dalla stazione indirizzata		
(s1)+4	Numero di punti da scrivere	Specifica quanti dati (in unità "word") devono essere scritti nella stazione intelligente.	da 1 a 480		
s2	Operandi di comunicazione usati per lo handshaking				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(s2)+0	Uscita remota (RY) per richiedere l'invio dei dati	• Byte più pesante Imposta a „0“ gli 8 bit più significativi.	0	Utente
			• Byte più leggero Specifica un'uscita remota (RY) della stazione intelligente	da 0 a 127	
(s2)+1	Registro remoto (RWr) usato come operando per la memorizzazione del codice di errore Ingresso remoto (RX) usato come operando di completamento.	• Byte più pesante Specifica un registro remoto (RWr) della stazione intelligente, in cui viene inserito lo stesso codice di errore contenuto in (s1)+0.	da 0 a 15 o FF (se impostato FF, nessun numero specificato).		
		• Byte più leggero Specifica un ingresso remoto (RX) della stazione intelligente	da 0 a 127		
(s2)+2	Modo completamento	Specifica come deve essere indicato il completamento del processo di lettura: 0: tramite 1 operando (RXn) 1: tramite 2 operandi (RXn, RXn+1) (RXn+1 viene attivato in caso di errore).	0 o 1		
d1	Indirizzo iniziale dell'area in cui sono contenuti i dati per la stazione intelligente.			Utente	BIN 16-bit

Variabili

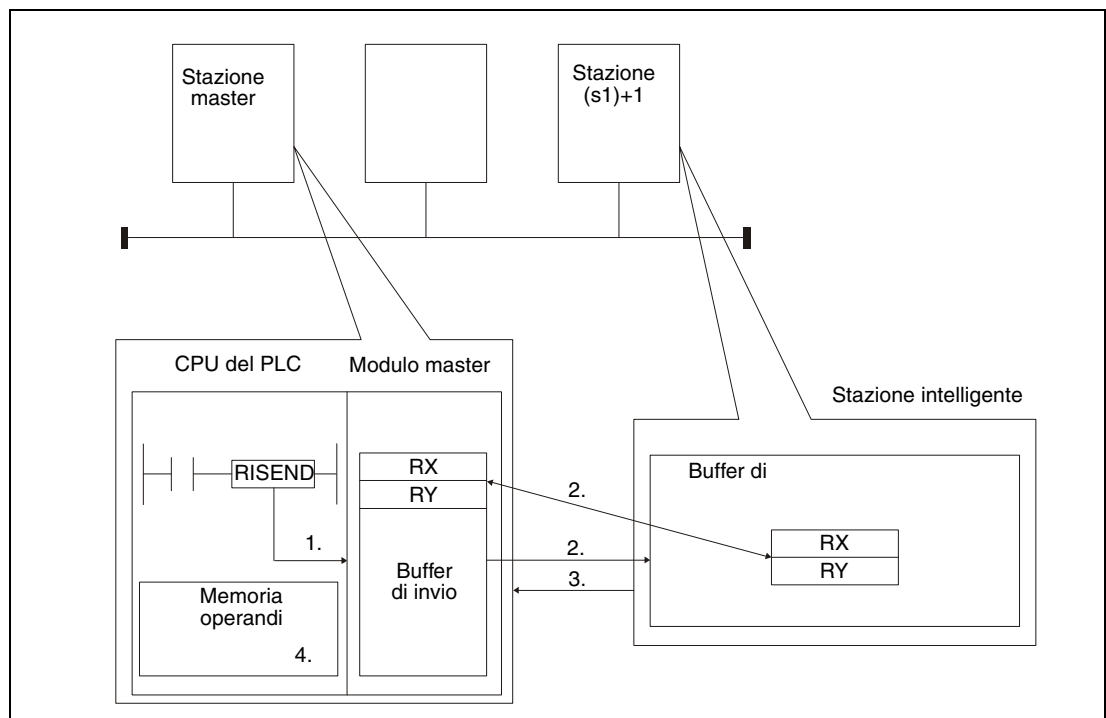
Dati impostati	Significato		Campo	Contenuto inserito da	Tipo dati
d2	Operando a bit attivato per una scansione al termine dell'istruzione RIRCV. (d2)+1 indica se si è verificato un errore durante l'esecuzione dell'istruzione.				
	Dati impostati	Significato	Descrizione	Campo	Contenuto inserito da
	(d2)+0	Istruzione completata	Indica lo stato di completamento dell'istruzione RISEND ON: Istruzione completata OFF: Istruzione non completata	—	Sistema
(d2)+1	Istruzione completata con errore	Indica l'occorrenza di un errore durante l'elaborazione dell'istruzione RISEND ON: Completamento anomalo OFF: Completamento normale	—		

Funzioni

Scrittura (con handshake) nel buffer di memoria di una stazione intelligente

RISEND Invio dati (con handshake)

L'istruzione RISEND può essere eseguita solo nella CPU del PLC di una stazione master e viene utilizzata per scrivere dati nel buffer di memoria di una stazione intelligente. Lo scambio dati viene controllato da operandi di handshake.



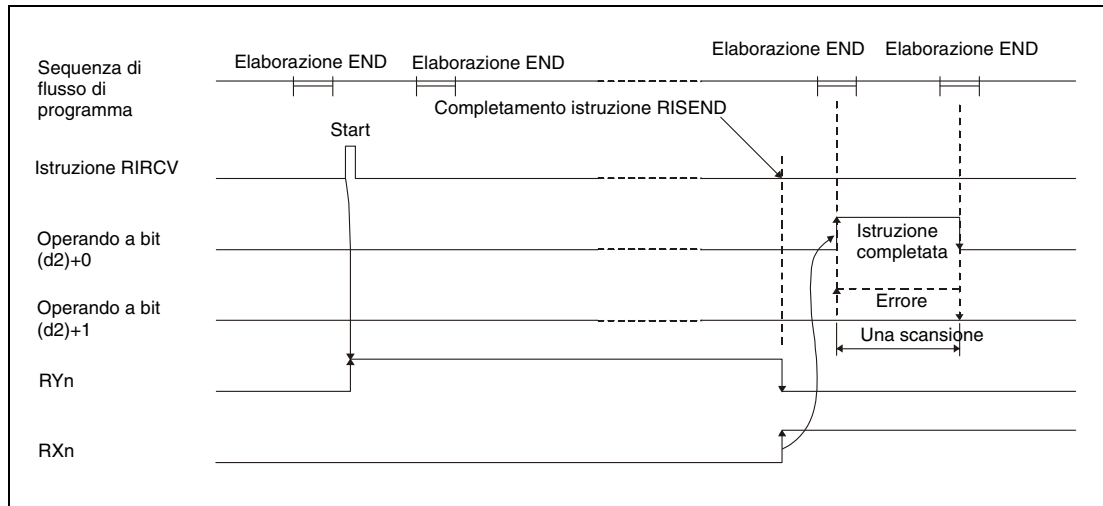
1. I dati per la stazione intelligente vengono inseriti nel buffer di invio della stazione master.
2. I dati vengono scritti nel buffer di memoria, a partire dall'indirizzo specificato da (s1)+3, della stazione indicata da da (s1)+1. Gli operandi specificati in s2 vengono utilizzati per lo handshake.
3. Una risposta di scrittura completata viene inviata alla stazione master.
4. Viene attivato l'operando specificato in (d2)+0.

Lo stato di esecuzione dell'istruzione RISEND può essere controllato con gli operandi (d2)+0 e (d2)+1:

- L'operando a bit (d2)+0 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione RISEND, e si disattiva con l'elaborazione di END della scansione successiva.

- L'operando a bit (d2)+1 segnala la presenza di un errore durante l'esecuzione della istruzione RISEND. Se l'istruzione termina normalmente, l'operando rimane OFF. Se si verifica un errore durante l'esecuzione dell'istruzione RISEND, (d2)+1 si attiva con l'elaborazione di END nella scansione in cui termina l'esecuzione dell'istruzione RISEND, e si disattiva con l'elaborazione di END della scansione successiva.

La figura seguente mostra la temporizzazione durante l'esecuzione di una istruzione RISEND:



Anche se è possibile eseguire contemporaneamente istruzioni RISEND da più stazioni intelligenti, non è possibile accedere simultaneamente alla stessa stazione da più stazioni diverse.

Errori di esecuzione

Nei casi seguenti, si verifica un errore di funzionamento, viene attivato il flag di errore SM0, ed un codice di errore viene memorizzato in SD0:

- Se il modulo specificato da Un non è un modulo intelligente o un modulo funzione speciale. (codice di errore: 2112)
- Se si cerca di eseguire una istruzione non supportata (codice di errore: 4002). 4002)
- Se il numero di operandi dell'istruzione non è corretto. (codice di errore: 4003)
- Se l'istruzione specifica un operando che non si può usare. (codice di errore: 4004)
- Se l'area specificata da s contiene dati che non si possono usare. (codice di errore: 4100)
- Se il numero di dati impostato supera il campo ammesso. (codice di errore: 4101)
- Se l'area di memoria o le costanti degli operandi specificati con l'istruzione superano il campo ammesso (codice di errore: 4101)

Esempio di programma

RISEND

Il programma seguente, eseguito nella CPU del PLC della stazione master, scrive 1 word di dati nella locazione 111_H del buffer di memoria della stazione intelligente con numero di stazione 63. Al modulo master CC-Link viene assegnato l'indirizzo iniziale di I/O X/Y000. Per lo handshake vengono utilizzati gli operandi RX4, RY4 e RWr4. Il completamento della scrittura è indicato da due operandi. ((s2)+2 è impostato a „1“.)

- Editor IEC (questo programma di esempio è mostrato a pagina seguente con lista istruzioni MELSEC e schema a contatti di GX Developer).

Schema a contatti (GX IEC Developer)

Lista istruzioni IEC

LD	M6	
MOV_P_M	63,	D1
MOV_P_M	16#0004,	D2
MOV_P_M	16#0111,	D3
MOV_P_M	1,	D4
MOV_P_M	16#0004,	D5
MOV_P_M	16#0104,	D6
MOV_P_M	16#0001,	D7
MOV_P_M	11,	D10
RISEND_P_MD	0,	var_D0, D10, var_D5, var_M40

Per una spiegazione su operandi e istruzioni usate, fare riferimento al programma precedente.

LD	M40
ANDN	M41

Una istruzione in questa posizione viene eseguita quando l'istruzione RISEND è stata completata normalmente.

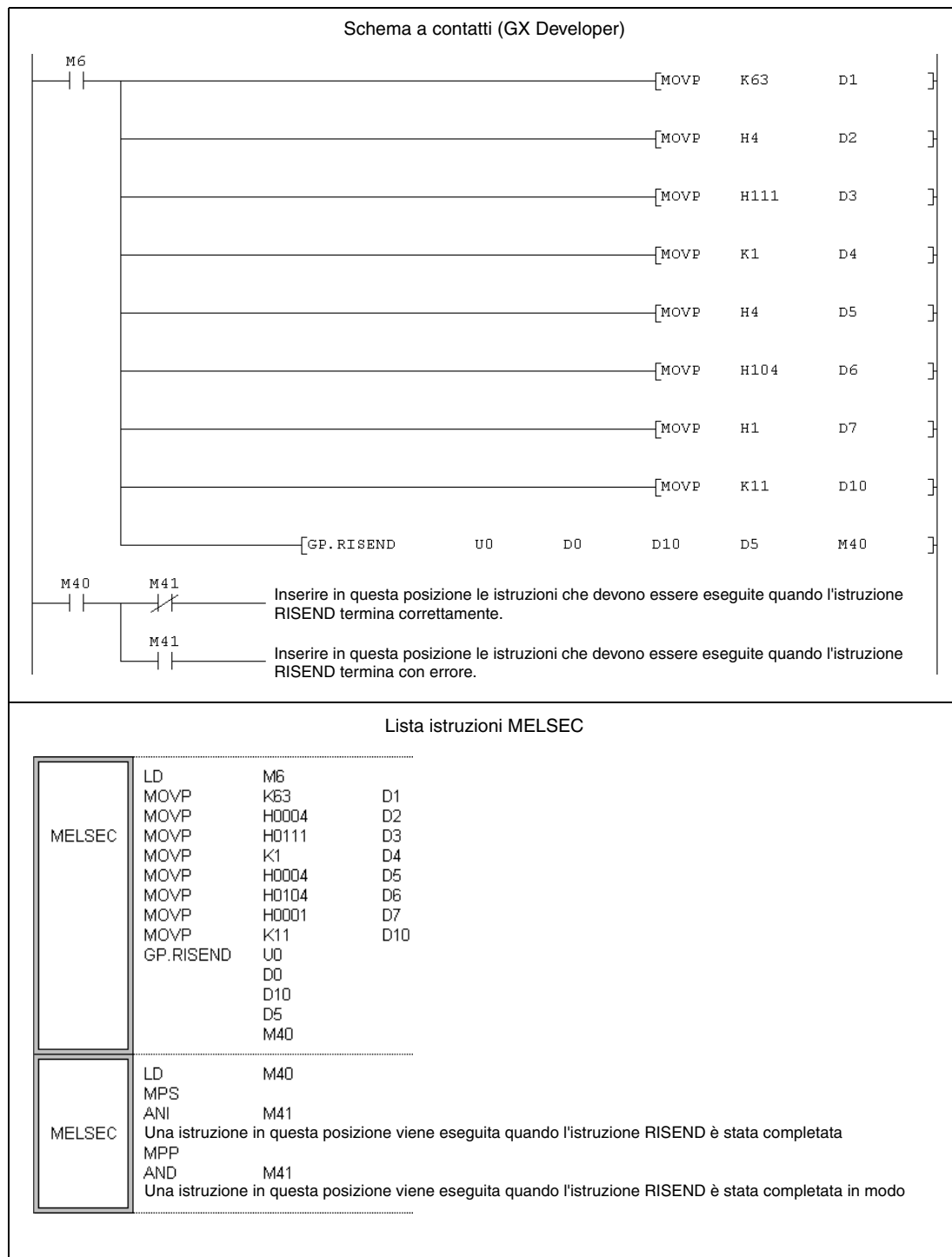
LD	M40
AND	M41

Una istruzione in questa posizione viene eseguita quando l'istruzione RISEND è stata completata in modo anomalo.

NOTA

Negli editor IEC è necessario definire le variabili nello header della unità organizzativa di programma (POU) La mancata definizione delle variabili provoca messaggi di errore del compilatore o della fase di controllo. Per i dettagli, vedi il capitolo 3.5.2 "Indirizzamento di array e registri in GX IEC Developer" di questo manuale.

- Lista istruzioni MELSEC e schema a contatti di GX Developer
Per una spiegazione degli operandi e istruzioni utilizzate, vedere il programma di esempio per lo schema a contatti di GX IEC Developer nella pagina precedente.



11.5.12 RITO (serie A)

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●		

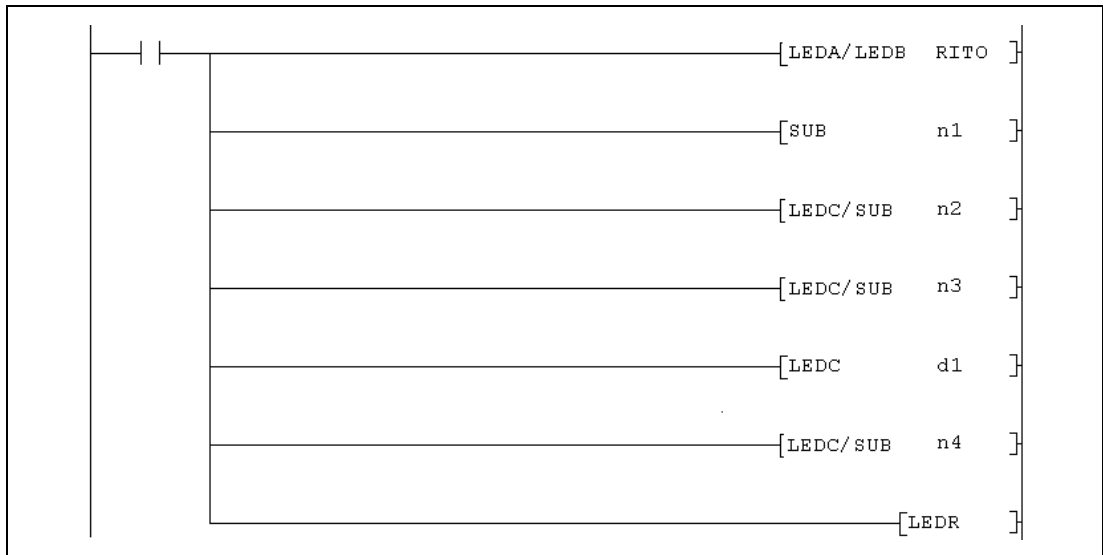
Operandi MELSEC A

	Operandi utilizzabili																Lunghezza blocco	Numero di passi	Indice	Riporto	Errore				
	Operandi a bit								Operandi a word (16-bit)											Costanti		Puntatore	Livello	M9012	M9011
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V				K	H (16#)	P	I		
n1																	●	●				29			
n2							●	●	●	●	●						●	●							
n3							●	●	●	●	●						●	●							
d1							●	●	●	●	●														
n4							●	●	●	●	●						●	●						●	

GX IEC Developer

<p style="text-align: center;">Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;">MELSEC</div> <ul style="list-style-type: none"> LEDA/LEDB RITO n1 SUB n1 LEDC/SUB n2 LEDC/SUB n3 LEDC d1 LEDC/SUB n4 LEDR 	<p style="text-align: center;">Schema a contatti</p>	<p style="text-align: center;">Lista istruzioni IEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;">RITO_MD</div> <p style="text-align: center;">n1, n2, n3, d1, n4</p>
---	--	---

GX Developer



Variabili

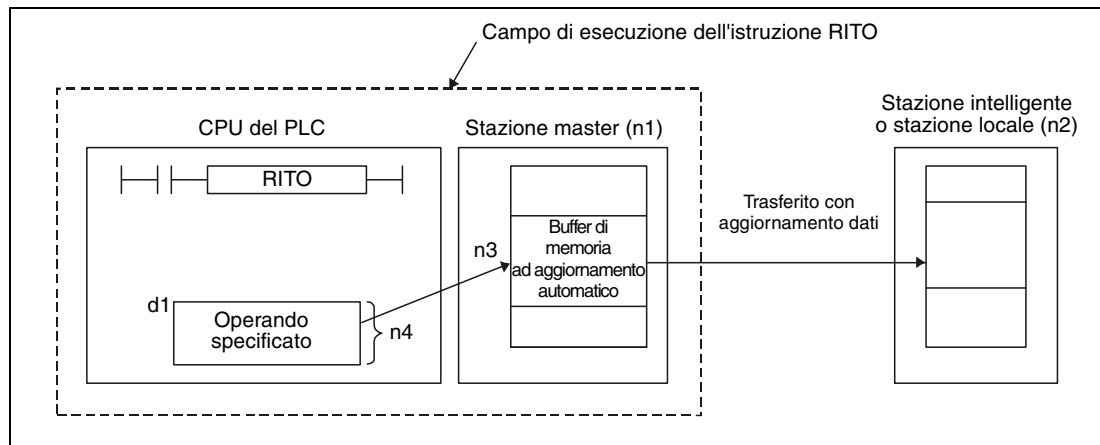
Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati
n1	Indirizzo iniziale di I/O del modulo master CC-Link (Vengono inserite solo le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come 10 _H)	da 0 a FE _H	Utente	BIN 16-bit
n2	Destinazione scrittura Specifica il numero di stazione della stazione intelligente in cui vengono scritti i dati (solo se la stazione che esegue l'istruzione RITO è la stazione master). Se i dati vengono inseriti nel buffer ad accesso casuale, specificare „FF _H “.	da 1 a 64 o FF _H		
n3	<ul style="list-style-type: none"> Indirizzo del buffer di invio/ricezione della stazione intelligente all'interno della stazione master, oppure offset del buffer ad accesso casuale 	Fra 0 e il valore massimo impostato nei parametri		
d1	Indirizzo iniziale dell'area operandi che contiene i dati da scrivere.	Nel campo dell'operando specificato		Indirizzo
n4	Numero di punti da scrivere (unità: word)	da 1 a 4096		BIN 16-bit

Funzioni **Scrittura nel buffer di memoria ad aggiornamento automatico****RITO** **Scrittura dati**

L'istruzione RITO trasferisce dati dalla memoria operandi della CPU del PLC al buffer di memoria ad aggiornamento automatico nella stazione master. Questi dati sono poi trasferiti ad un'altra stazione su CC-Link.

I dati vengono specificati come indirizzo iniziale (d1) e numero di word (n4). La destinazione nella stazione master è indicata da n2 (corrisponde al numero di stazione a cui i dati vengono alla fine inviati) e n3 (indirizzo iniziale del buffer di memoria ad aggiornamento automatico nella stazione master). L'indirizzo iniziale di I/O della stazione master è specificato da n1.

Il funzionamento dell'istruzione RITO viene mostrato nella figura seguente:



Con l'istruzione RITO si possono scrivere fino a 4096 word.

La dimensione del buffer ad aggiornamento automatico può essere impostata con l'istruzione RLPA.

Condizioni di esecuzione

Se viene usata l'istruzione LEDA, l'istruzione RITO viene eseguita ad ogni scansione in cui il comando di scrittura è ON.

Se viene usata l'istruzione LEDB, l'istruzione RITO viene eseguita per una scansione sul fronte positivo (OFF -> ON) del comando di scrittura.

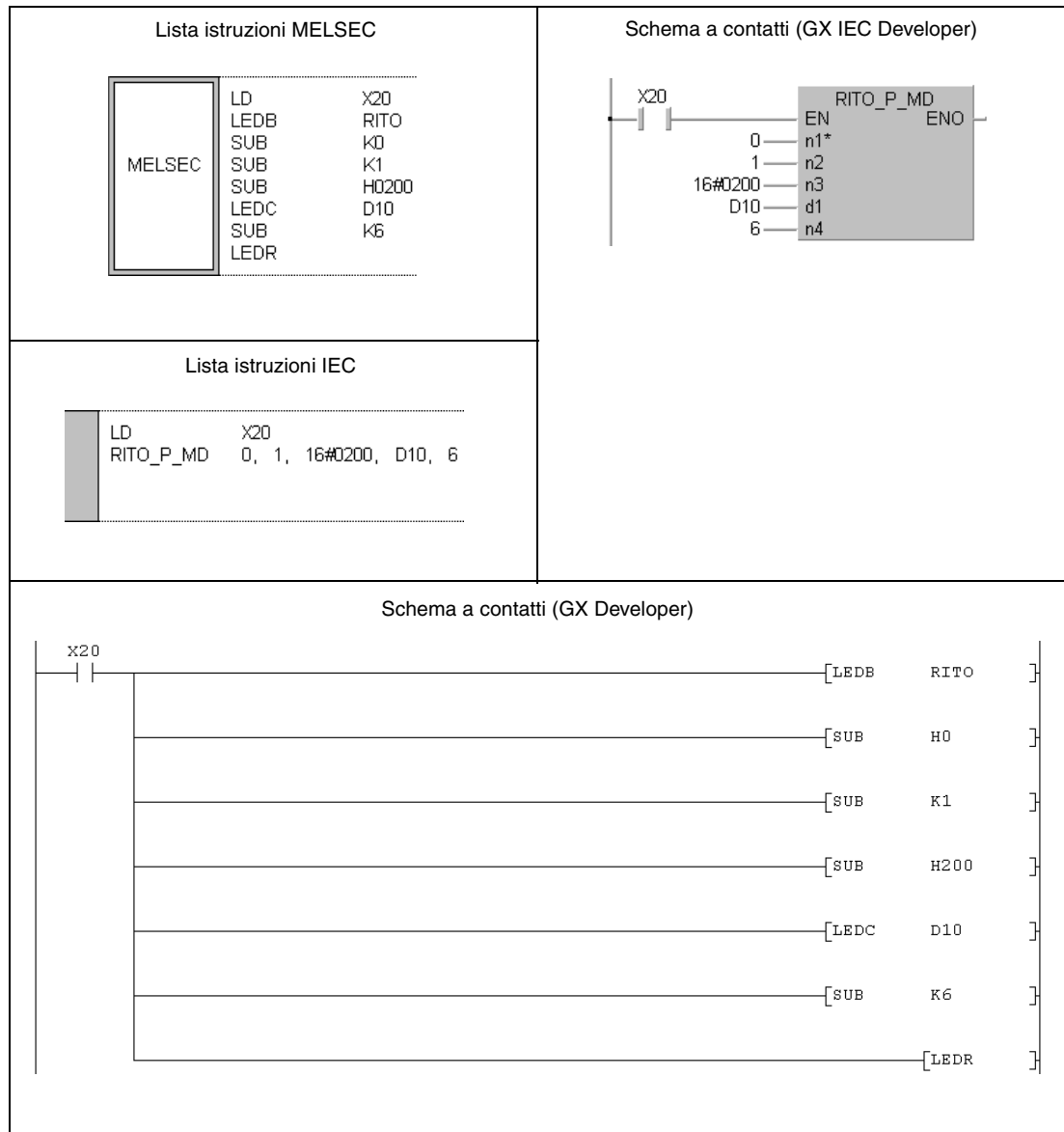
Errori di esecuzione

Una qualsiasi delle condizioni seguenti comporta un errore di esecuzione. In questi casi, viene attivato il flag di errore M9011 e viene emesso un codice di errore:

- L'indirizzo specificato per il buffer di memoria è oltre il campo ammesso.
(codice di errore in D9008: 50, codice di errore in D99091 (AnUCPU) o D9092 (AnSHCPU): 503)
- La quantità di dati da scrivere è superiore a 4096.
(codice di errore in D9008: 50, codice di errore in D99091 (AnUCPU) o D9092 (AnSHCPU): 503)

Esempio di programma**RITO**

Se l'ingresso X20 è attivo, il contenuto dei sei registri da D10 a D15 viene spostato nel buffer di memoria ad aggiornamento automatico per la stazione numero 1 nel modulo master. Qui i dati vengono memorizzati a partire dall'indirizzo 200_H. Il modulo master CC-Link è allocato con indirizzi di I/O da X/Y000 a X/Y01F.



Per ulteriori informazioni sulla programmazione delle istruzioni dedicate con gli editor MELSEC, fare riferimento al capitolo 3.3 di questo manuale.

11.5.13 RITO (serie QnA e System Q)

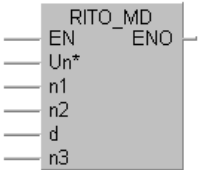
CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

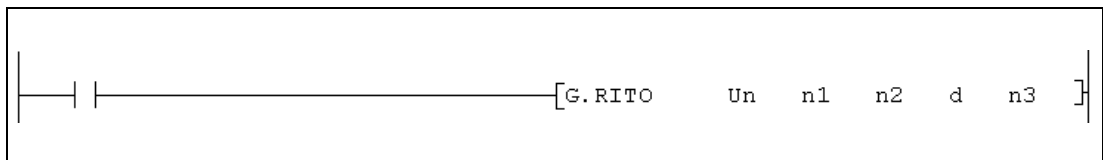
Operandi MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
n1	●	●	●	—	—	—	—	●	—	SM0	9
n2	●	●	●	—	—	—	—	●	—		
d	—	●	●	—	—	—	—	—	—		
n3	●	●	●	—	—	—	—	●	—		

GX IEC Developer

<p style="text-align: center;">Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p style="margin: 0;">MELSEC</p> </div> <p style="margin-top: 10px;">G.RITO Un n1 n2 d n3</p>	<p style="text-align: center;">Schema a contatti</p> 	<p style="text-align: center;">Lista istruzioni IEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p style="margin: 0;">RITO_MD Un, n1, n2, d, n3</p> </div>
--	---	--

GX Developer



Variabili

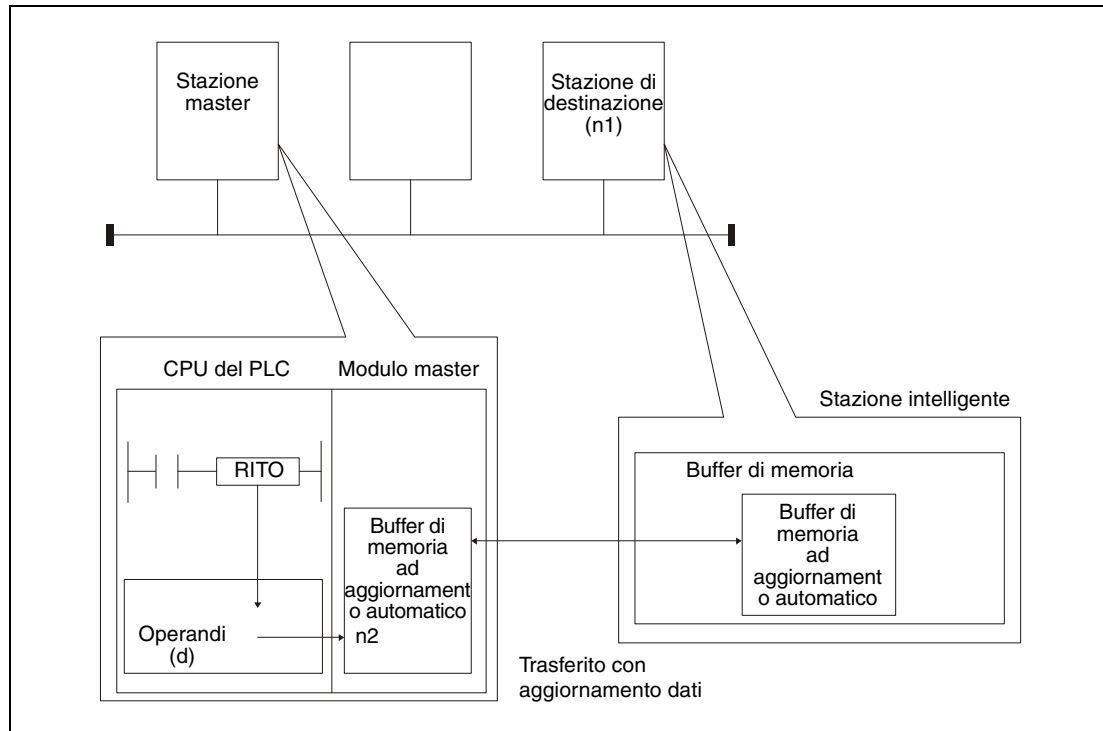
Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati
Un	Indirizzo iniziale di I/O del modulo master CC-Link (Vengono inserite solo le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come 10 _H)	da 0 a FE _H	Utente	BIN 16-bit
n1	Destinazione scrittura <ul style="list-style-type: none"> • Specifica il numero di stazione della stazione intelligente in cui vengono scritti i dati. • Se i dati vengono inseriti nel buffer ad accesso casuale, specificare „FF_H“. 	da 1 a 64 o FF _H		
n2	Il valore di offset nel buffer ad aggiornamento automatico della stazione intelligente specificato dalla stazione master, o del buffer ad aggiornamento casuale. L'indirizzo di inizio della zona da scrivere è indicato relativamente all'indirizzo iniziale del buffer ad aggiornamento automatico. Un esempio: per scrivere dati all'indirizzo 356 _H del buffer di memoria, che inizia dall'indirizzo 350 _H , in n2 si deve inserire il valore 6 _H .	Fra 0 e il valore massimo impostato nei parametri.		
d	Indirizzo iniziale dell'area operandi che contiene i dati da scrivere.	Nel campo dell'operando specificato		Indirizzo
n3	Numero di punti da scrivere (unità: word)	da 1 a 4096		BIN 16-bit

Funzioni **Scrittura nel buffer di memoria ad aggiornamento automatico****RITO** **Scrittura dati**

L'istruzione RITO trasferisce dati dalla memoria operandi della CPU del PLC al buffer di memoria ad aggiornamento automatico nella stazione master. Questi dati sono poi trasferiti ad un'altra stazione su CC-Link.

I dati vengono specificati come indirizzo iniziale (d) e numero di word (n3). La destinazione nella stazione master è indicata da n1 (corrisponde al numero di stazione a cui i dati vengono alla fine inviati) e n2 (indirizzo iniziale del buffer di memoria ad aggiornamento automatico nella stazione master). L'indirizzo iniziale di I/O della stazione master è specificato da Un.

Il funzionamento dell'istruzione RITO viene mostrato nella figura seguente:



L'istruzione RITO non può essere eseguita su più di una stazione per la stessa stazione intelligente.

Con l'istruzione RITO si possono scrivere fino a 4096 word.

L'assegnazione dei buffer ad aggiornamento automatico viene effettuata usando le "impostazioni informazioni stazione" dei parametri di rete di GX Developer e GX IEC Developer.

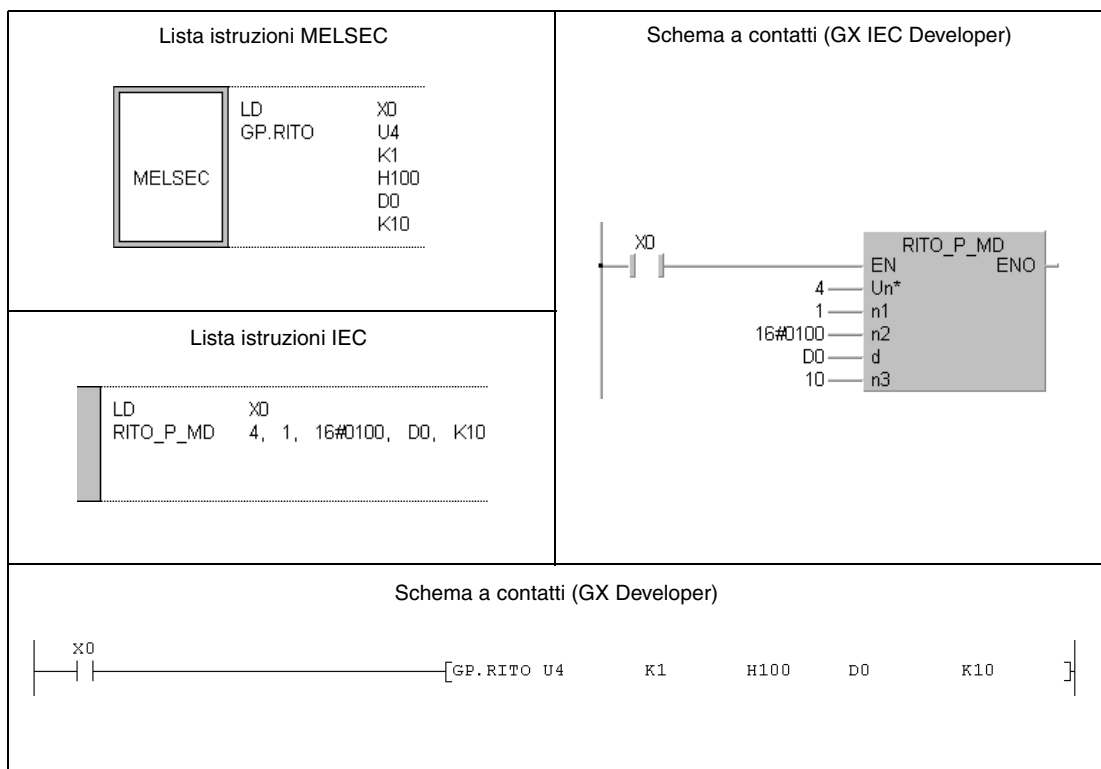
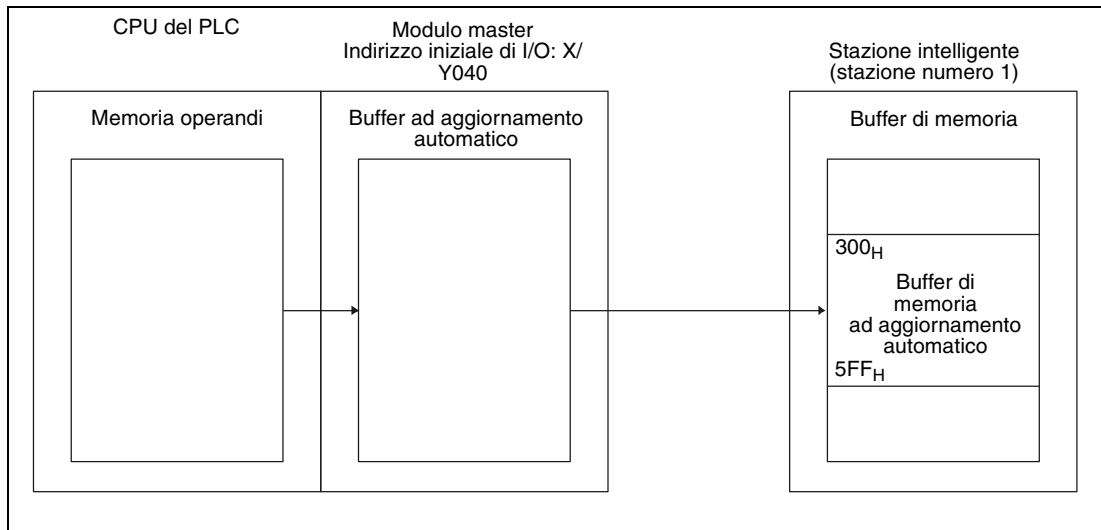
Errori di esecuzione

Nei casi seguenti, si verifica un errore di funzionamento, viene attivato il flag di errore SM0, ed un codice di errore viene memorizzato in SD0:

- Se il modulo specificato da Un non è un modulo intelligente o un modulo funzione speciale. (codice di errore: 2112)
- Se si cerca di eseguire una istruzione non supportata (codice di errore: 4002). 4002)
- Se il numero di operandi dell'istruzione non è corretto. (codice di errore: 4003)
- Se l'istruzione specifica un operando che non si può usare. (codice di errore: 4004)
- Se non esiste la stazione il cui numero è specificato da n1. (codice di errore: 4100)
- Se il numero di word da scrivere specificato da n3 è oltre il campo ammesso. (codice di errore: 4100)

Esempio di programma RITO

Se l'ingresso X0 è attivo, il contenuto dei dieci registri (da D0 a D10) viene spostato nel buffer di memoria ad aggiornamento automatico per la stazione numero 1 nel modulo master. Questo buffer inizia all'indirizzo 300_H. I dati vengono memorizzati a partire dall'indirizzo 400_H (offset = 100).



11.5.14 RIFR (serie A)

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
●	●	●	●		

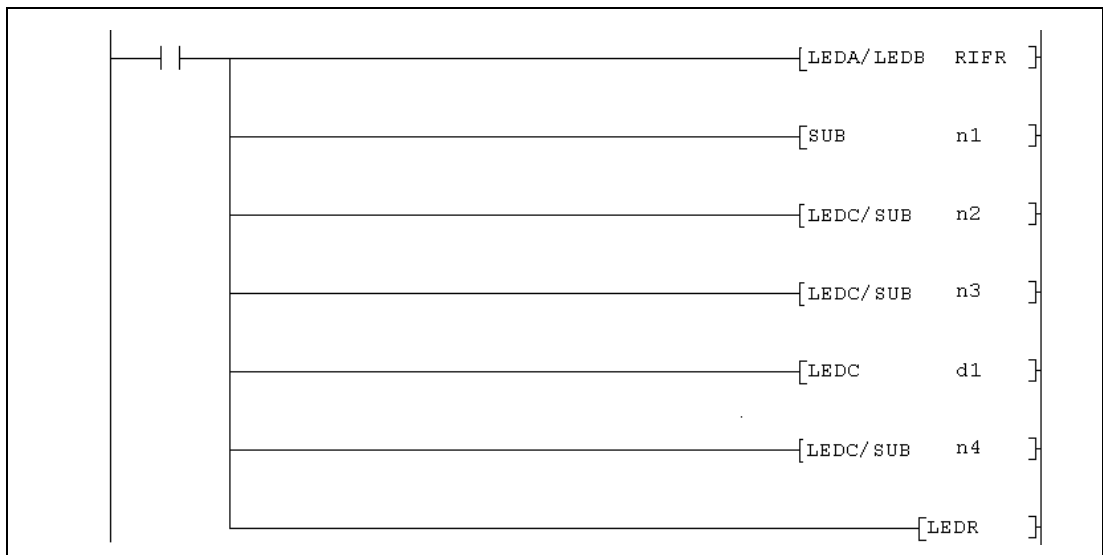
Operandi MELSEC A

	Operandi utilizzabili																Lunghezza blocco	Numero di passi	Indice	Riporto	Errore								
	Operandi a bit								Operandi a word (16-bit)											Costanti		Puntatore	Livello	M9012	M9011				
	X	Y	M	L	S	B	F	T	C	D	W	R	A0	A1	Z	V				K	H (16#)	P	I			N			
n1																	●	●											
n2							●	●	●	●	●						●	●											
n3							●	●	●	●	●						●	●											●
n4							●	●	●	●	●						●	●											
d1							●	●	●	●	●																		

GX IEC Developer

<p style="text-align: center;">Lista istruzioni MELSEC</p> <table border="1" style="width: 100%;"> <tr> <td style="text-align: center;">MELSEC</td> <td>LEDA/LEDB</td> <td>RIFR</td> </tr> <tr> <td></td> <td>SUB</td> <td>n1</td> </tr> <tr> <td></td> <td>LEDC/SUB</td> <td>n2</td> </tr> <tr> <td></td> <td>LEDC/SUB</td> <td>n3</td> </tr> <tr> <td></td> <td>LEDC</td> <td>d1</td> </tr> <tr> <td></td> <td>LEDC/SUB</td> <td>n4</td> </tr> <tr> <td></td> <td>LEDR</td> <td></td> </tr> </table>	MELSEC	LEDA/LEDB	RIFR		SUB	n1		LEDC/SUB	n2		LEDC/SUB	n3		LEDC	d1		LEDC/SUB	n4		LEDR		<p style="text-align: center;">Schema a contatti</p>	<p style="text-align: center;">Lista istruzioni IEC</p> <table border="1" style="width: 100%;"> <tr> <td>RIFR_MD</td> <td>n1, n2, n3, n4, d1</td> </tr> </table>	RIFR_MD	n1, n2, n3, n4, d1
MELSEC	LEDA/LEDB	RIFR																							
	SUB	n1																							
	LEDC/SUB	n2																							
	LEDC/SUB	n3																							
	LEDC	d1																							
	LEDC/SUB	n4																							
	LEDR																								
RIFR_MD	n1, n2, n3, n4, d1																								

GX Developer



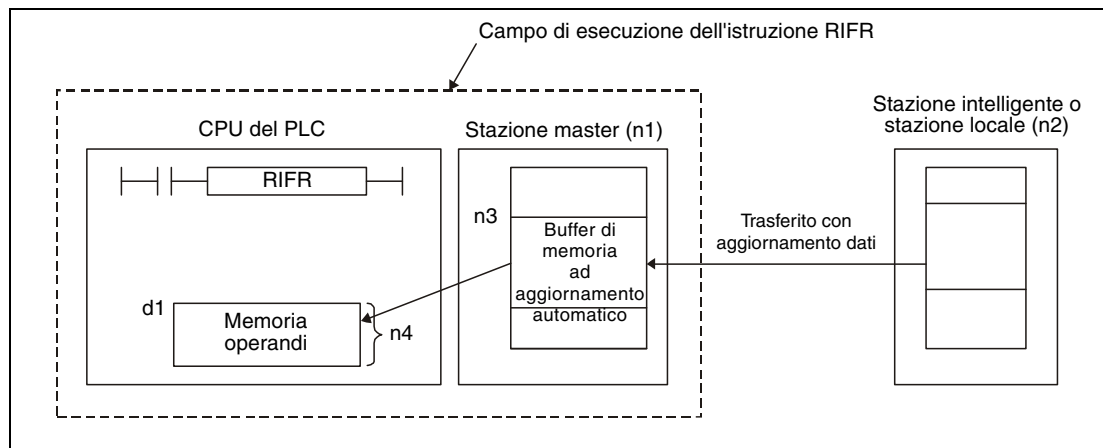
Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati
n1	Indirizzo iniziale di I/O del modulo master CC-Link (Vengono inserite solo le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come 10 _H)	da 0 a FE _H	Utente	BIN 16-bit
n2	Sorgente dei dati Specifica il numero di stazione della stazione intelligente da cui vengono letti i dati (solo se la stazione che esegue l'istruzione RIFR è la stazione master). Se i dati vengono letti dal buffer ad accesso casuale, specificare „FF _H “.	da 1 a 64 o FF _H		
n3	<ul style="list-style-type: none"> Indirizzo del buffer di invio/ricezione della stazione intelligente all'interno della stazione master, oppure offset del buffer ad accesso casuale 	Fra 0 e il valore massimo impostato nei parametri		
n4	Numero di punti da leggere (unità: word)	da 1 a 4096		
d1	Indirizzo iniziale dell'area operandi dove vengono memorizzati i dati letti.	Nel campo dell'operando specificato		Indirizzo

Funzioni Lettura da buffer di memoria ad aggiornamento automatico**RIFR Lettura dati**

L'istruzione RIFR trasferisce dati dal buffer di memoria ad aggiornamento automatico nella stazione master alla memoria operandi della CPU del PLC. L'area di memorizzazione di questi dati viene specificata come indirizzo iniziale (d1) e numero di word (n4). La sorgente dei dati è indicata dal numero di stazione contenuto in n2 e dall'indirizzo iniziale nel buffer ad aggiornamento automatico nella stazione master (n3). L'indirizzo iniziale di I/O della stazione master è specificato da n1.

Il funzionamento dell'istruzione RIFR viene mostrato nella figura seguente:



Con l'istruzione RIFR si possono leggere fino a 4096 word.

La dimensione del buffer ad aggiornamento automatico può essere impostata con l'istruzione RLPA.

Condizioni di esecuzione

Se viene usata l'istruzione LEDA, l'istruzione RIFR viene eseguita ad ogni scansione in cui il comando di scrittura è ON.

Se viene usata l'istruzione LEDB, l'istruzione RIFR viene eseguita per una scansione sul fronte positivo (OFF -> ON) del comando di lettura.

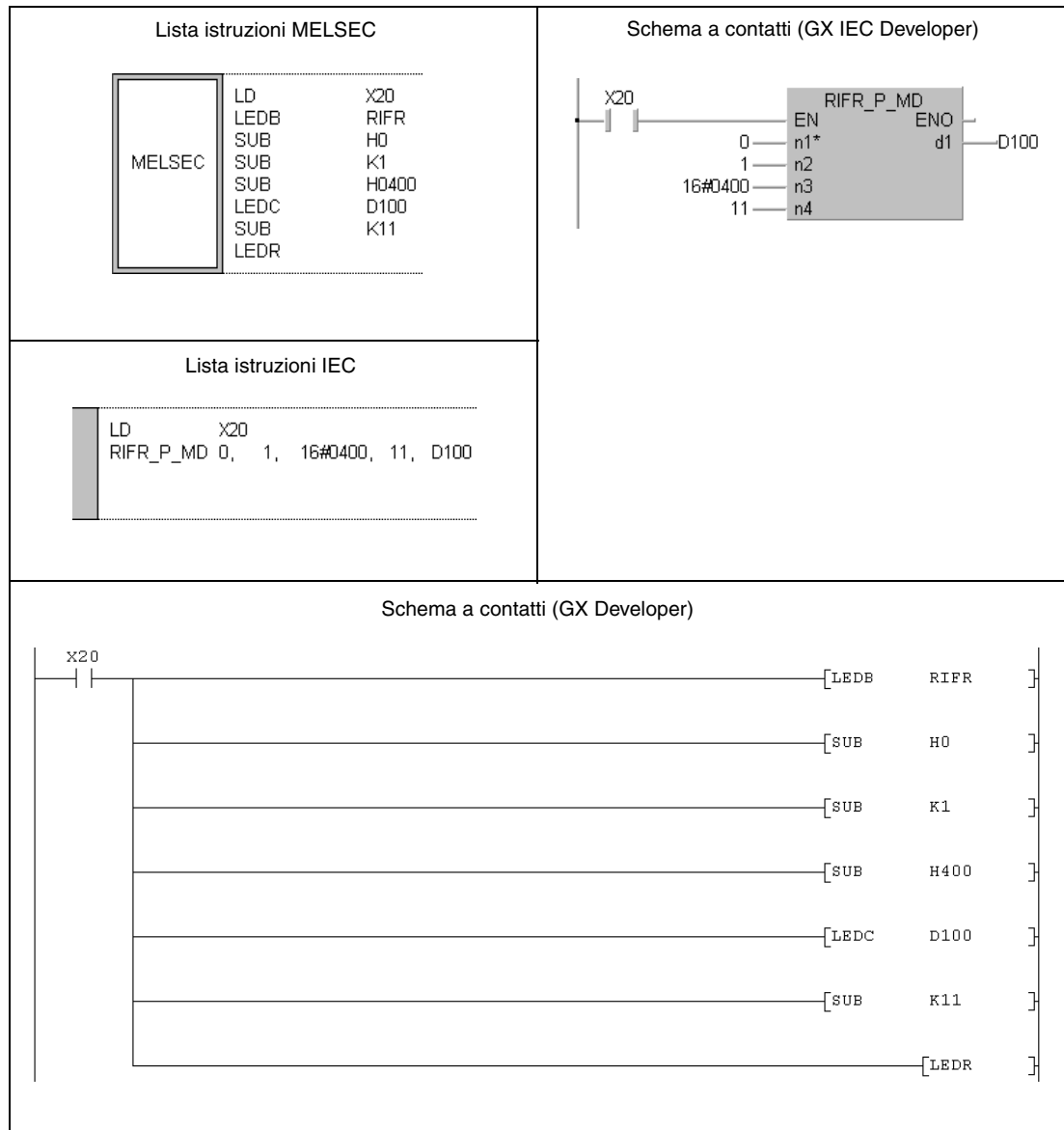
Errori di esecuzione

Una qualsiasi delle condizioni seguenti comporta un errore di esecuzione. In questi casi, viene attivato il flag di errore M9011 e viene emesso un codice di errore:

- L'indirizzo specificato per il buffer di memoria è oltre il campo ammesso.
(codice di errore in D9008: 50, codice di errore in D99091 (AnUCPU) o D9092 (AnSHCPU): 503)
- La quantità di dati da scrivere è superiore a 4096.
(codice di errore in D9008: 50, codice di errore in D99091 (AnUCPU) o D9092 (AnSHCPU): 503)

Esempio di programma RIFR

Se l'ingresso X20 è attivo, il programma seguente legge il contenuto di 11 punti del buffer ad aggiornamento automatico impostato per la stazione 1 nel modulo master, a partire dall'indirizzo 400_H. Questi dati vengono poi memorizzati nella CPU del PLC a partire dal registro D100. Il modulo master CC-Link è allocato con indirizzi di I/O da X/Y000 a X/Y01F.



Per ulteriori informazioni sulla programmazione delle istruzioni dedicate con gli editor MELSEC, fare riferimento al capitolo 3.3 di questo manuale.

11.5.15 RIFR (serie QnA e System Q)

CPU

AnS	AnN	AnA(S)	AnU	QnA(S), Q4AR	System Q
				●	●

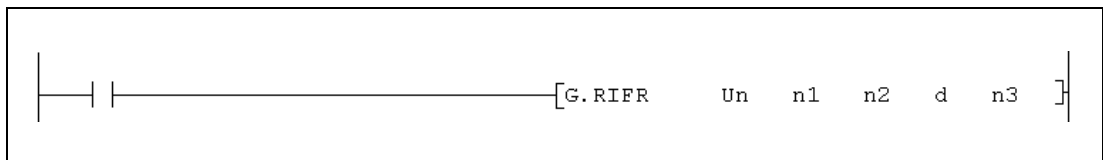
Operandi MELSEC Q

	Operandi utilizzabili									Errore	Numero di passi
	Operandi interni (sistema, utente)		Registro file	MELSECNET/10 Direct J□□		Modulo funzione speciale U□NG□	Registro indice Zn	Costanti K, H (16#)	Altro		
	Bit	Word		Bit	Word						
n1	●	●	●	—	—	—	—	●	—	SM0	9
n2	●	●	●	—	—	—	—	●	—		
n3	●	●	●	—	—	—	—	●	—		
d	—	●	●	—	—	—	—	—	—		

GX IEC Developer

<p>Lista istruzioni MELSEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> <p>MELSEC</p> </div> <p>G.RIFR Un n1 n2 d n3</p>	<p>Schema a contatti</p>	<p>Lista istruzioni IEC</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> <p>RIFR_MD Un, n1, n2, n3, d</p> </div>
---	--------------------------	---

GX Developer



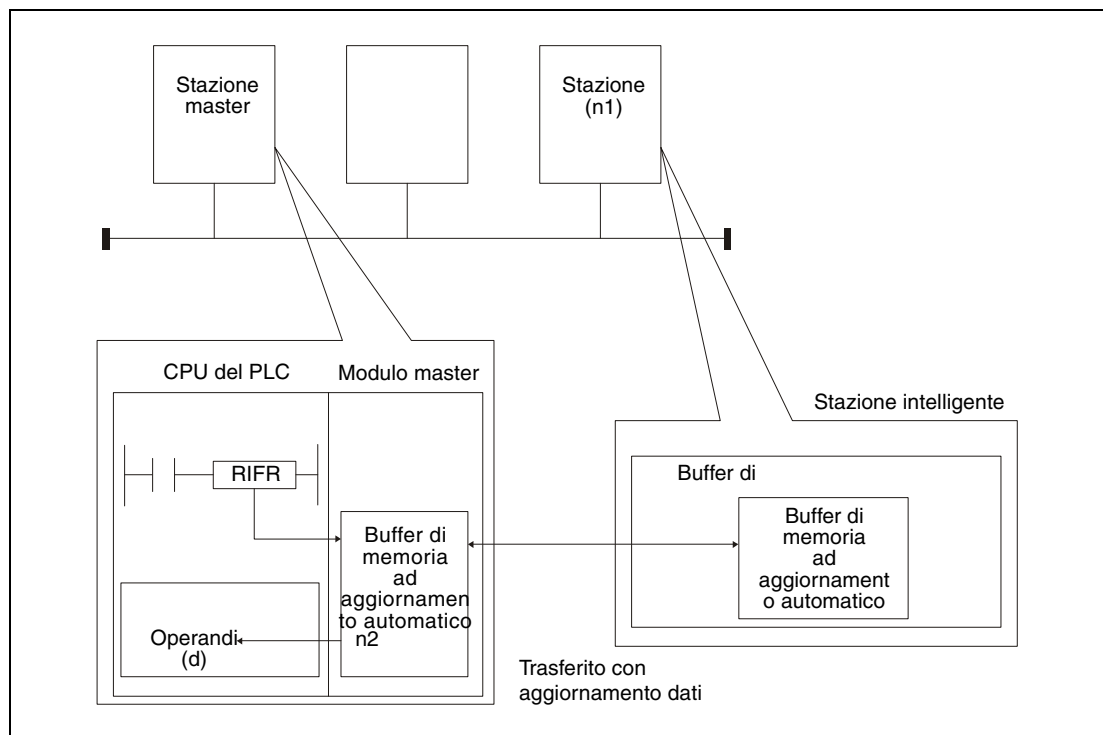
Variabili

Dati impostati	Significato	Campo	Contenuto inserito da	Tipo dati
Un	Indirizzo iniziale di I/O del modulo master CC-Link (Vengono inserite solo le due cifre più pesanti di un indirizzo espresso come numero da 3 cifre, ad es. l'indirizzo iniziale di X/Y100 viene impostato come 10 _H)	da 0 a FE _H	Utente	BIN 16-bit
n1	Sorgente dei dati <ul style="list-style-type: none"> • Specifica il numero di stazione della stazione intelligente da cui vengono letti i dati. • Se i dati vengono letti dal buffer ad accesso casuale, specificare „FF_H“. 	da 1 a 64 o FF _H		
n2	Il valore di offset nel buffer ad aggiornamento automatico della stazione intelligente specificato dalla stazione master, o del buffer ad aggiornamento casuale. L'indirizzo di inizio dei dati da leggere è indicato relativamente all'indirizzo iniziale del buffer ad aggiornamento automatico. Un esempio: per leggere dati dall'indirizzo 356 _H del buffer di memoria, che inizia dall'indirizzo 350 _H , in n2 si deve inserire il valore 6 _H .	Fra 0 e il valore massimo impostato nei parametri		
n3	Numero di punti da leggere (unità: word)	da 1 a 4096		
d	Indirizzo iniziale dell'area operandi dove vengono memorizzati i dati letti.	Nel campo dell'operando specificato		Indirizzo

Funzioni Lettura da buffer di memoria ad aggiornamento automatico**RIFR Lettura dati**

L'istruzione RIFR trasferisce dati dal buffer di memoria ad aggiornamento automatico nella stazione master alla memoria operandi della CPU del PLC. L'area di memorizzazione di questi dati viene specificata come indirizzo iniziale (d) e numero di word (n3). La sorgente dei dati è indicata dal numero di stazione contenuto in n1 e dall'offset nel buffer ad aggiornamento automatico nella stazione master (n2). L'indirizzo iniziale di I/O della stazione master è specificato da Un.

Il funzionamento dell'istruzione RIFR viene mostrato nella figura seguente:



L'istruzione RIFR non può essere eseguita su più di una stazione per la stessa stazione intelligente.

Con l'istruzione RIFR si possono leggere fino a 4096 word.

L'assegnazione dei buffer ad aggiornamento automatico viene effettuata usando le "impostazioni informazioni stazione" dei parametri di rete di GX Developer e GX IEC Developer.

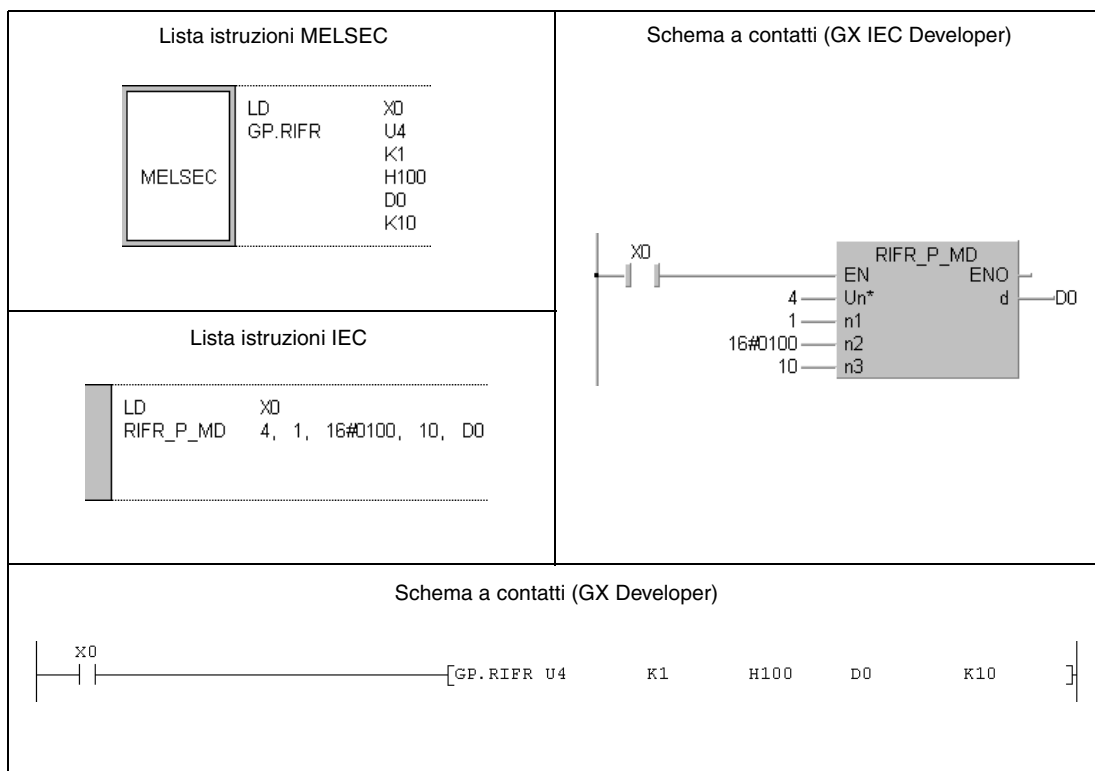
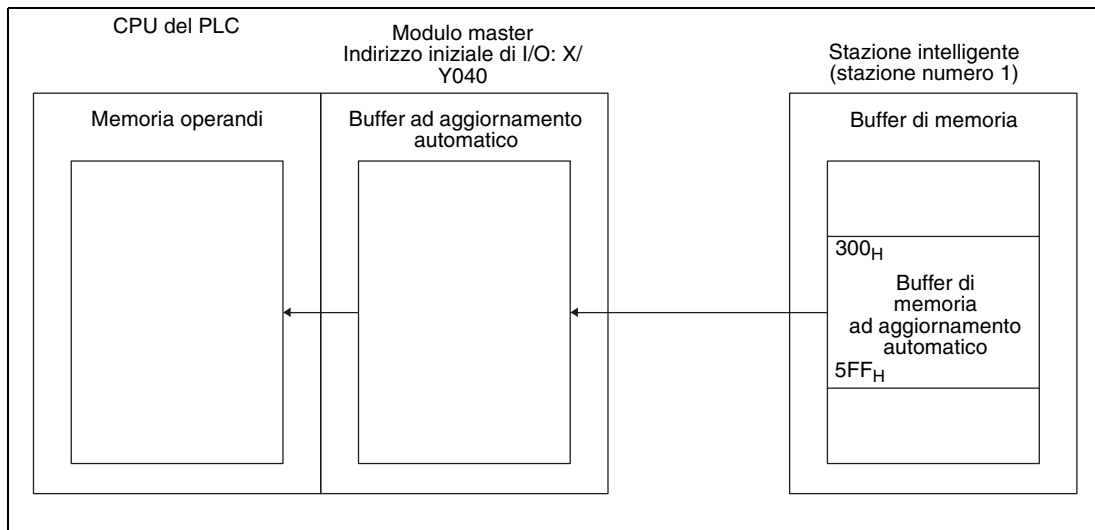
Errori di esecuzione

Nei casi seguenti, si verifica un errore di funzionamento, viene attivato il flag di errore SM0, ed un codice di errore viene memorizzato in SDO:

- Se il modulo specificato da Un non è un modulo intelligente o un modulo funzione speciale. (codice di errore: 2112)
- Se si cerca di eseguire una istruzione non supportata (codice di errore: 4002). 4002)
- Se il numero di operandi dell'istruzione non è corretto. (codice di errore: 4003)
- Se l'istruzione specifica un operando che non si può usare. (codice di errore: 4004)
- Se non esiste la stazione il cui numero è specificato da n1. (codice di errore: 4100)
- Se il numero di word da scrivere specificato da n3 è oltre il campo ammesso. (codice di errore: 4100)

Esempio di programma RIFR

Se l'ingresso X0 è attivo, il programma seguente legge il contenuto di 10 punti del buffer ad aggiornamento automatico predisposto per la stazione numero 1 nel modulo master, e memorizza questi dati nella CPU del PLC a partire dal registro D0. Il buffer inizia all'indirizzo 300_H. La lettura inizia dall'indirizzo 400_H (offset = 100). Il modulo master CC-Link è allocato con indirizzi di I/O a partire da X/Y040.



12 Modalità Microcomputer (AnN(S))

La serie A MELSEC (tranne AnA, AnAS e AnU) supportano l'esecuzione combinata di programmi a sequenza e programmi microcomputer. I programmi microcomputer consentono l'esecuzione di sequenze di programma oltre al livello delle macroistruzioni (programma principale e sottoprogramma). Un programma microcomputer viene invocato con una istruzione SUB(P).

Le CPU MELSEC AnA, AnAS, AnUS, QnA, QnAS e System Q non elaborano programmi microcomputer.

12.1 Capacità di memorizzazione ed aree di memoria

La tabella seguente mostra una panoramica delle capacità di memorizzazione e delle aree di memoria per programmi microcomputer per le varie CPU:

CPU	Processore	Area programmi microcomputer	Area di lavoro	Area stack	Istruzioni non supportate			
A1	8086 (8 MHz)	0 – 10 kByte	A100H – A1FFH (256 Byte)	Area utente: 128 Byte	INT, INTO, IRET, IN, OUT, HLT, WAIT, LOCK, ESC			
A2		0 – 26 kByte						
A3		0 – 58 kByte (MAIN) 0 – 58 kByte (SUB)						
A1N	8086 (10 MHz)	0 – 10 kByte						
A2N-S1		0 – 26 kByte						
A2S		0 – 26 kByte						
A3N		0 – 58 kByte (MAIN) 0 – 58 kByte (SUB)						
A1S	8086 (8 MHz)	0 – 14 kByte						
A1S-S1		0 – 14 kByte						
A2C		0 – 26 kByte						
A3H	80286 (8 MHz)	0 – 58 kByte (MAIN) 0 – 58 kByte (SUB)						INT, INTO, IRET, IN, OUT, HLT, WAIT, LOCK, ESC, CLI, STI
A3M		0 – 58 kByte (MAIN) 0 – 58 kByte (SUB)						

NOTA

L'area per programmi microcomputer viene specificata in multipli di 2 kByte. Le aree di memoria per le singole sezioni di programma (programma microcomputer e sequenziale nel campo MAIN e SUB) non devono essere sovrapposte.

Per evitare malfunzionamenti, non utilizzare mai istruzioni non supportate dai programmi microcomputer (fare riferimento alla tabella precedente).

12.2 Utilizzo di programmi microcomputer creati dall'utente

Il codice sorgente scritto dall'utente in linguaggio assembler 8086 viene compilato (convertito) nel codice macchina interpretabile dai programmi assembler del PLC sotto sistemi operativi CP/M[®] o MS-DOS[®]. Il programma compilato viene chiamato "programma oggetto" e viene memorizzato nell'area di memoria microcomputer della CPU. Un compilatore C trasferisce il file OBJ al PLC tramite un terminale di programmazione.

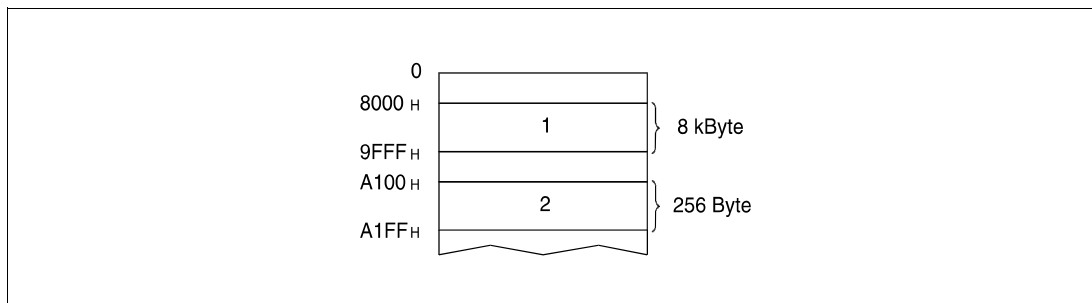
NOTA *Controllare se queste funzioni sono disponibili e supportate dalla versione del software di programmazione utilizzata.*

Attenzioni durante la stesura di programmi microcomputer

- Prevedere l'istruzione PUSH all'inizio del programma microcomputer, in modo da salvare nello stack il contenuto dei registri usati durante l'esecuzione. Inserire inoltre l'istruzione POP alla fine del programma per ripristinare il contenuto dei registri salvati nello stack.
- Inizializzare i registri utilizzati dal programma microcomputer all'inizio del programma. Quando il programma microcomputer viene richiamato dal programma di sequenza, il contenuto dei registri non è definito.
- Dato che il programma microcomputer viene eseguito solo se richiamato dal programma di sequenza tramite l'istruzione SUB(P), il programma di sequenza è sempre necessario.
- Per tornare dal programma microcomputer al programma di sequenza, usare l'istruzione RETF.

12.2.1 Mappa di memoria

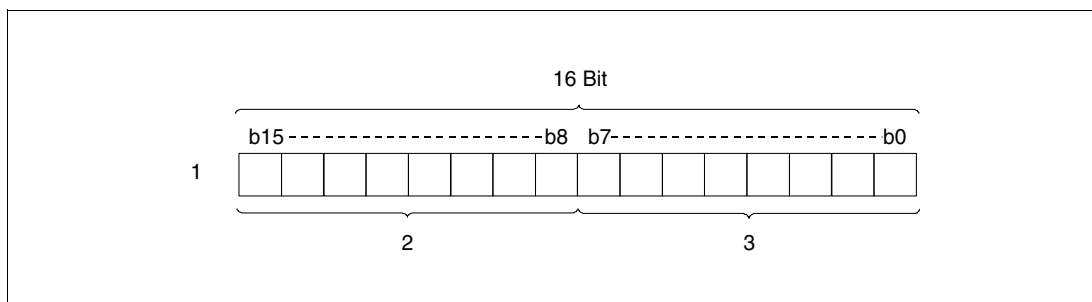
Il programma microcomputer è contenuto in due diverse aree di memoria della CPU. L'area da 8000H a 9FFFH, con una capacità di 8 kByte viene usata per contenere i dati, mentre l'area da A100H a A1FFH viene usata come area di lavoro per il programma microcomputer.



- 1 Area memorizzazione dati
- 2 Area di lavoro programma microcomputer

12.2.2 Configurazione indirizzi dell'area memorizzazione dati

Una locazione dell'area memorizzazione dati è composta da 16 bit e viene suddivisa in una zona pari e una dispari, ciascuna da 8 bit. La figura seguente mostra la configurazione di una locazione.



- 1 Una locazione 8000H
- 2 Zona dispari da 8 bit (8001H)
- 3 Zona pari da 8 bit (8000H)

12.2.3 Configurazione area memoria dati

L'area della memoria dati da 8000H a 9FFFH viene usata dalla CPU per memorizzare i dati degli operandi. La tabella seguente mostra la configurazione dell'area in funzione degli operandi:

Operando	CPU Tipo	Indirizzo	Configurazione
Istruzione (X)	A1 A1N	8000H – 803FH	X0 - FF
	A2 A2C A2N A2S	8000H – 803FH	X0 - 1FF
	A2N- S1	8000H – 80FFH	X0 - 3FF
	A3 A3N	8000H – 81FFH	X0 - 7FF
Uscita (Y)	A1 A1N	8200H – 823FH	Y0 - FF
	A2 A2C A2N A2S	8000H – 827FH	Y0 - 1FF
	A2N- S1	8200H – 827FH	X0 - 3FF
	A3 A3N	8200H – 83FFH	Y0 - 7FF

1 2

b15 b14 b13 b12 b11 b10 b9 b8 b7 b6 b5 b4 b3 b2 b1 b0

8000H XIM7 XIM6 XIM5 XIM4 XIM3 XIM2 XIM1 XIM0 X7 X6 X5 X4 X3 X2 X1 X0

8002H XIMF XIME XIMD XIMC XIMB XIMA XIM9 XIM8 XF XE XD XC XB XA X9 X8

8004H XIM1 XIM16 XIM15 XIM14 XIM13 XIM12 XIM11 XIM10 X17 X16 X15 X14 X13 X12 X11 X10

3 4

1 2

b15 ----- b8 b7 b6 b5 b4 b3 b2 b1 b0

8200H Y7 Y6 Y5 Y4 Y3 Y2 Y1 Y0

8202H YF YE YD YC YB YA Y9 Y8

8204H Y17 Y16 Y15 Y14 Y13 Y12 Y11 Y10

5

Nel funzionamento in modo rinfresco, i dati vengono scritti e letti dal modulo di uscita tramite la memoria di uscita.

Nel funzionamento in modo diretto, i dati vengono scritti direttamente nel modulo di uscita, ma letti dal modulo tramite la memoria di uscita.

- ¹ Indirizzi dispari
- ² Indirizzi pari
- ³ Area contenente lo stato ON/OFF dei dati di una stazione remota (lettura e scrittura): 0 = OFF, 1 = ON.
L'ingresso effettivo corrisponde alla espressione: (X)=(XIM) ∨ (X̄)
- ⁴ Area contenente lo stato ON/OFF dei dati di un modulo di ingresso (solo lettura): 0 = OFF, 1 = ON.
- ⁵ Area contenente i risultati delle operazioni del PLC (lettura e scrittura): 0 = OFF, 1 = ON.

Operando	CPU Tipo	Indirizzo	Configurazione
Relé interni (M) Relé retentivi (L) Relé di passo (S)	A1 A2 A3 A1N A2N-S1 A3N A1S A1S-S1 A2C	8400H – 85FFH	M/L/S 0 - 2047
Relé di comunicazione (B)		8600H – 86FFH	F0 - 255
Spie di errore (F)		8700H – 873FH	F0 - 255
Relé speciali (M)		8740H – 877FH	M 9000-9255
Temporizzatore Contatto (T)		8780H – 87BFH	T0 - 255
Contatore Contatto (C)		87C0H – 87FFH	C0 - 255
Temporizzatore Bobina (T)		9C00H – 9C3FH	T0 - 255
Contatore Bobina (C)		9C40H – 9C7FH	C0 - 255

8400H b15-----b8 b7 b6 b5 b4 b3 b2 b1 b0
 8402H M7 M6 M5 M4 M3 M2 M1 M0
 8404H M17 M16 M15 M14 M13 M12 M11 M10

↓
3

- 1 Zona dispari
- 2 Zona pari
- 3 Zona contenente i risultati dell'elaborazione del PLC (lettura e scrittura).

Operando	CPU Tipo	Indirizzo	Configurazione	
Dati EEPROM (D)	A1 A2 A3 A1N A2N-S1 A3N A1S1 A1S-S1 A2C	8800H – 8FFFH	<p>8800H b7-----b0 8801H b15-----b8</p>	
Registro di comunicazione (W)		9000H – 97FFH		W0 - 3FF
Valore attuale di timer (T)		9800H – 99FFH		T0 - 255
Valore attuale di contatori (C)		9A00H – 9BFFH		C0 - 255
Modulo EEPROM (D)		9D00H – 9EFFH		D9000-9255
Accumulatori (A0, A1)		9FF8H – 9FFAH		A0 / A1
Indice EEPROM (Z, V)		9FFCH – 9FFEH		Z / V

Operando	CPU Tipo	Indirizzo	Configurazione	
Istruzione (X)	A3H A3M	8000H — 80FFH	<div style="text-align: center;"> </div>	
Uscita (Y)		8200H — 82FFH	<div style="text-align: center;"> <p>Nel funzionamento in modo rinfresco, i dati vengono scritti e letti dal modulo di uscita tramite la memoria di uscita.</p> <p>Nel funzionamento in modo diretto, i dati vengono scritti direttamente nel modulo di uscita, ma letti dal modulo tramite la memoria di uscita.</p> </div>	
Relé interni (M) Relé retentivi (L) Relé di passo (S)		8400H — 84FFH	M/L/S 0 - 2047	
Relé di comunicazione (B)		8600H — 867FH	B0 - 3FF	<div style="text-align: center;"> </div>
Spie di errore (F)		8700H — 871FH	F0 - 255	<div style="text-align: center;"> </div>

¹ Indirizzi dispari

² Indirizzi pari

³ Area contenente lo stato ON/OFF dei dati di un modulo di ingresso (solo lettura)

⁴ Zona contenente i risultati dell'elaborazione del PLC (lettura e scrittura).

Operando	CPU Tipo	Indirizzo		Configurazione																																																																
Relé speciali (M)	A3H A3M	8740H	M 9000-9255	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;"> <p>1</p> <p>2</p> <table border="1" style="border-collapse: collapse; text-align: center; font-size: 8px;"> <tr> <td>b15</td><td>b14</td><td>b13</td><td>b12</td><td>b11</td><td>b10</td><td>b9</td><td>b8</td><td>b7</td><td>b6</td><td>b5</td><td>b4</td><td>b3</td><td>b2</td><td>b1</td><td>b0</td> </tr> <tr> <td>M15</td><td>M14</td><td>M13</td><td>M12</td><td>M11</td><td>M10</td><td>M9</td><td>M8</td><td>M7</td><td>M6</td><td>M5</td><td>M4</td><td>M3</td><td>M2</td><td>M1</td><td>M0</td> </tr> <tr> <td>M31</td><td>M30</td><td>M29</td><td>M28</td><td>M27</td><td>M26</td><td>M25</td><td>M24</td><td>M23</td><td>M22</td><td>M21</td><td>M20</td><td>M19</td><td>M18</td><td>M17</td><td>M16</td> </tr> <tr> <td>M47</td><td>M46</td><td>M45</td><td>M44</td><td>M43</td><td>M42</td><td>M41</td><td>M40</td><td>M39</td><td>M38</td><td>M37</td><td>M36</td><td>M35</td><td>M34</td><td>M33</td><td>M32</td> </tr> </table> </div> <div style="text-align: center; margin-left: 100px;"> <p>3</p> </div> </div>	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	M15	M14	M13	M12	M11	M10	M9	M8	M7	M6	M5	M4	M3	M2	M1	M0	M31	M30	M29	M28	M27	M26	M25	M24	M23	M22	M21	M20	M19	M18	M17	M16	M47	M46	M45	M44	M43	M42	M41	M40	M39	M38	M37	M36	M35	M34	M33	M32
b15		b14			b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																		
M15		M14			M13	M12	M11	M10	M9	M8	M7	M6	M5	M4	M3	M2	M1	M0																																																		
M31		M30	M29		M28	M27	M26	M25	M24	M23	M22	M21	M20	M19	M18	M17	M16																																																			
M47		M46	M45		M44	M43	M42	M41	M40	M39	M38	M37	M36	M35	M34	M33	M32																																																			
–		–																																																																		
875FH		–																																																																		
Temporizzatore Contatto (T)		8780H	T0 - 255																																																																	
–		–																																																																		
879FH		–																																																																		
Contatore Contatto (C)	87C0H	C0 - 255																																																																		
–	–																																																																			
87DFH	–																																																																			
Temporizzatore Bobina (T)	9C00H	T0 - 255																																																																		
–	–																																																																			
9C1FH	–																																																																			
Contatore Bobina (C)	9C40H	C0 - 255																																																																		
–	–																																																																			
9C5FH	–																																																																			

¹ Indirizzi dispari

² Indirizzi pari

³ Zona contenente i risultati dell'elaborazione del PLC (lettura e scrittura).

Operando	CPU Tipo	Indirizzo	Configurazione
Dati EEPROM (D)	A3H A3M	8800H – 8FFFH	<div style="display: flex; align-items: center; justify-content: center;"> <div style="margin-right: 20px;">8800H</div> <div style="border: 1px solid black; width: 150px; height: 15px; margin-right: 5px;"></div> <div style="margin-right: 20px;">8801H</div> <div style="border: 1px solid black; width: 150px; height: 15px; margin-right: 5px;"></div> </div> <p style="text-align: center; margin-top: 5px;">b7-----b0 b15-----b8</p>
Registro di comunicazione (B)		9000H – 97FFH	
Valore attuale di timer (T)		9800H – 99FFH	
Valore attuale di contatori (C)		9A00H – 9BFFH	
Modulo EEPROM (D)		9D00H – 9EFFH	
Accumulatori (A0, A1)		9FF8H – 9FFAH	
Indice EEPROM (Z, V)		9FFCH – 9FFEH	

Operando	CPU Tipo	Indirizzo
<p>Registro file (R)</p> <p>Numero blocco 0</p>	<p>A2</p> <p>A3</p> <p>A2N</p> <p>A2N-S1</p> <p>A2S</p> <p>A3N</p> <p>A3H</p> <p>A3M</p> <p>A1S</p> <p>A1S-S1</p> <p>A2C</p>	<p>Indirizzo base del file registri</p> <p>= 20000H + (capacità cartuccia RAM) - (capacità del file registri)</p> <p>Capacità cartucce RAM (valori per calcolo)</p> <p>A3(N)MCA-0 = 16 kByte A3(N)MCA-2 = 16 kByte A3(N)MCA-4 = 32 kByte A3(N)MCA-8 = 64 kByte A3MCA-12 = 96 kByte A3MCA-16 = 144 kByte (capacità effettiva: 128k) A3MCA-18 = 144 kByte A3MCA-24 = 144 kByte (capacità effettiva: 192k) A3NMCA-40 = 144 kByte (capacità effettiva: 320k) A3NMCA-56 = 144 kByte (capacità effettiva: 448k)</p> <p>Valore per calcoli</p> <p>Capacità per commenti: (numero di commenti) x 16 Byte + 1 kByte</p> <p>Capacità file registri: (numero di file registri) x 2 Byte</p> <p>Nota: Per il calcolo delle capacità, notare che 1 kByte corrisponde a 1024 byte e non a 1000 byte.</p>
<p>Espansione File registri (R)</p> <p>Numero blocco 1 – 9</p>		<p>Indirizzo iniziale del file registri per ciascun numero di blocco</p> <p>= 20000H + (capacità cartuccia RAM) - (capacità commenti) - (capacità file registri) - (capacità zona retentiva) - (capacità zona campioni) - 4000H x n</p> <p>Capacità per commenti: (numero di commenti) x 16 Byte + 1 kByte</p> <p>Capacità file registri: (numero di file registri) x 2 Byte</p> <p>Capacità zona retentiva: Numero byte impostati</p> <p>Capacità zona campioni: 8 kByte (se impostato)</p> <p>n: numero di blocco</p>

Operando	CPU Tipo	Indirizzo																																																				
Espansione file registri (R) Numero blocco 10 - 28	A2 A3 A2N A2N-S1 A2S A3N A3H A3M A1S A1S-S1 A2C	Cartucce di memoria <div style="display: flex; justify-content: space-around; align-items: flex-start;"> <table border="1" style="margin: 10px;"> <thead> <tr> <th colspan="2" style="text-align: center;">1</th> </tr> <tr> <th style="width: 50px;">3</th> <th style="width: 50px;">4</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">11</td> <td style="text-align: center;">38000 H</td> </tr> <tr> <td style="text-align: center;">10</td> <td style="text-align: center;">3C000 H</td> </tr> <tr> <td> </td> <td> </td> </tr> </tbody> </table> <table border="1" style="margin: 10px;"> <thead> <tr> <th colspan="2" style="text-align: center;">2</th> </tr> <tr> <th style="width: 50px;">3</th> <th style="width: 50px;">4</th> </tr> </thead> <tbody> <tr><td style="text-align: center;">28</td><td style="text-align: center;">A0000 H</td></tr> <tr><td style="text-align: center;">27</td><td style="text-align: center;">A4000 H</td></tr> <tr><td style="text-align: center;">26</td><td style="text-align: center;">A8000 H</td></tr> <tr><td style="text-align: center;">25</td><td style="text-align: center;">AC000 H</td></tr> <tr><td style="text-align: center;">24</td><td style="text-align: center;">B0000 H</td></tr> <tr><td style="text-align: center;">23</td><td style="text-align: center;">B4000 H</td></tr> <tr><td style="text-align: center;">22</td><td style="text-align: center;">B8000 H</td></tr> <tr><td style="text-align: center;">21</td><td style="text-align: center;">BC000 H</td></tr> <tr><td style="text-align: center;">20</td><td style="text-align: center;">C0000 H</td></tr> <tr><td style="text-align: center;">19</td><td style="text-align: center;">C4000 H</td></tr> <tr><td style="text-align: center;">18</td><td style="text-align: center;">C8000 H</td></tr> <tr><td style="text-align: center;">17</td><td style="text-align: center;">CC000 H</td></tr> <tr><td style="text-align: center;">16</td><td style="text-align: center;">D0000 H</td></tr> <tr><td style="text-align: center;">15</td><td style="text-align: center;">D4000 H</td></tr> <tr><td style="text-align: center;">14</td><td style="text-align: center;">D8000 H</td></tr> <tr><td style="text-align: center;">13</td><td style="text-align: center;">DC000 H</td></tr> <tr><td style="text-align: center;">12</td><td style="text-align: center;">E4000 H</td></tr> <tr><td style="text-align: center;">11</td><td style="text-align: center;">E8000 H</td></tr> <tr><td style="text-align: center;">10</td><td style="text-align: center;">EC000 H</td></tr> </tbody> </table> </div>	1		3	4	11	38000 H	10	3C000 H			2		3	4	28	A0000 H	27	A4000 H	26	A8000 H	25	AC000 H	24	B0000 H	23	B4000 H	22	B8000 H	21	BC000 H	20	C0000 H	19	C4000 H	18	C8000 H	17	CC000 H	16	D0000 H	15	D4000 H	14	D8000 H	13	DC000 H	12	E4000 H	11	E8000 H	10	EC000 H
		1																																																				
3	4																																																					
11	38000 H																																																					
10	3C000 H																																																					
2																																																						
3	4																																																					
28	A0000 H																																																					
27	A4000 H																																																					
26	A8000 H																																																					
25	AC000 H																																																					
24	B0000 H																																																					
23	B4000 H																																																					
22	B8000 H																																																					
21	BC000 H																																																					
20	C0000 H																																																					
19	C4000 H																																																					
18	C8000 H																																																					
17	CC000 H																																																					
16	D0000 H																																																					
15	D4000 H																																																					
14	D8000 H																																																					
13	DC000 H																																																					
12	E4000 H																																																					
11	E8000 H																																																					
10	EC000 H																																																					

- ¹ A3NMCA-16
- ² A3NMCA-24, 40 o 56
- ³ Numero blocco
- ⁴ Indirizzo iniziale

13 Codici di errore

Se si verifica un errore mentre il PLC è alimentato e predisposto in modo RUN, oppure durante il funzionamento, la funzione di autodiagnosi della CPU genera una segnalazione di errore (spia LED o messaggio sul pannello frontale) e memorizza informazioni sull'errore in relé speciali (M) o relé diagnostici (SM) e nel registro speciale (D9008) o nei registri diagnostici (SD).

13.1 Tabella dei codici di errore; CPU Q00J, Q00 e Q01

La tabella seguente contiene una panoramica di tutti i possibili errori ed i messaggi di errore corrispondenti, le cause possibili e le azioni consigliate. Sono indicati solo gli errori relativi a Q00JCPU, Q00CPU e Q01CPU.☆

Codice di errore (SD0) ¹	Messaggio di errore	Informazioni generali (da SD5 a 15) ¹	Informazioni specifiche (da SD13 a 20)	Stato LED		CPU Stato	Temporizzazione diagnostica
				RUN	ERROR		
1000	MAIN CPU DOWN	—	—	OFF	Lampeggiante/ON	Stop	Sempre
1010	END NOT EXECUTE	—	—	OFF	Lampeggiante	Stop	Durante l'esecuzione dell'istruzione END.
1011							
1012							
1101	RAM ERROR	—	—	OFF	Lampeggiante	Stop	Accensione/reset
1102							
1103							
1104							
1200	OPE. CIRCUIT ERR.	—	—	OFF	Lampeggiante	Stop	Accensione/reset
1201							
1202							
1300	FUSE BREAK OFF	Num. unità/modulo	—	OFF/ON	Lampeggiante/ON	Stop/Continua ²	Durante l'esecuzione dell'istruzione END.
1310	I/O INT ERROR	Num. unità/modulo	—	OFF	Lampeggiante	Stop	Durante una interruzione
1401	SP. UNIT DOWN	Num. unità/modulo	—	OFF	Lampeggiante	Stop/Continua ³	Accensione/reset Accedendo a un modulo funzione intelligente.
1402			Posizione errore programmazione				Accedendo a un modulo funzione intelligente.
1403			—				Durante l'esecuzione dell'istruzione END.

¹ Le informazioni in parentesi () indicano gli indirizzi dei registri speciali in cui sono memorizzate le singole informazioni.

² Lo stato operativo della CPU a seguito di un errore può essere configurato tramite parametri (il display a LED cambia corrispondentemente).

³ Per ciascun modulo è possibile configurare il funzionamento stop/continua tramite parametri.

	Descrizione errore e causa	Rimedio
	Modo RUN sospeso o guasto della CPU principale 1.) Malfunzionamento dovuto a rumori o altre cause 2.) Guasto hardware	1.) Misurare il livello di rumore. 2.) Resetare la CPU e tornare in modo RUN.
	L'intero programma è stato eseguito senza incontrare una istruzione END. 1.) Quando è stata eseguita l'istruzione END è stata letta come un diverso codice istruzione. 2.) L'istruzione END è stata alterata dal codice di un'altra istruzione.	Se si ripresenta di nuovo lo stesso errore, è possibile che la CPU sia guasta. Contattare il più vicino centro di Assistenza MITSUBISHI.
	Errore nella RAM interna della CPU contenente il programma di sequenza. Errore nella RAM usata come area di lavoro della CPU. Errore interno della CPU Errore indirizzamento RAM della CPU	Questo indica un guasto hardware della CPU. Contattare il più vicino centro di Assistenza MITSUBISHI.
	Malfunzionamento della circuiteria che esegue l'indicizzazione interna alla CPU. Malfunzionamento hardware CPU (logica) Malfunzionamento della circuiteria che esegue l'elaborazione della sequenza.	
	Fusibile interrotto nel modulo di uscita.	1.) Controllare i LED di errore dei moduli di uscita e sostituire il modulo con il LED acceso. 2.) I moduli con fusibile interrotto possono anche essere controllati con un terminale di programmazione. Controllare i registri speciali da SD130 a SD137 sul display del terminale di programmazione e controllare la presenza di bit attivi (1) che corrispondono ai moduli con fusibile interrotto.
	Si è verificata una interruzione, anche se non esistono moduli con interruzione nel sistema.	Uno dei moduli collegati presenta un guasto hardware. Controllare i moduli collegati. Contattare il più vicino centro di Assistenza MITSUBISHI.
	1.) Mancata risposta da un modulo funzione intelligente durante le comunicazioni iniziali. 2.) La dimensione del buffer di memoria del modulo funzione intelligente non è normale.	Il modulo CPU presenta un guasto hardware. Contattare il più vicino centro di Assistenza MITSUBISHI.
	Il programma ha interrogato un modulo funzione speciale, senza ottenere risposta. 1.) Mancata risposta da un modulo funzione intelligente durante l'esecuzione dell'istruzione END. 2.) Rilevato errore sul modulo funzione intelligente.	Il modulo CPU presente un guasto hardware. Contattare il più vicino centro di Assistenza MITSUBISHI. Il modulo interrogato presenta un guasto hardware. Contattare il più vicino centro di Assistenza MITSUBISHI.

Codici di errore da 1411 a 2112

Codice di errore (SD0) ¹	Messaggio di errore	Informazioni generali (da SD5 a 15) ¹	Informazioni specifiche (da SD13 a 20) ¹	Stato LED		Stato CPU	Temporizzazione diagnostica
				RUN	ERROR		
1411	CONTROL-BUS ERR.	Num. unità/modulo	Posizione errore programmazione	OFF	Lampeggiante	Stop	Accensione/reset
1412							Durante l'esecuzione della coppia di istruzioni FROM/TO.
1413							Sempre
1414							Durante l'esecuzione dell'istruzione END.
1415							
1500	AC DOWN	—	—	ON	OFF	Continua	Sempre
1600	ERRORE BATTERIA	Nome drive	—	ON	OFF	Continua	Sempre
2000	UNIT VERFIY ERR.	Num. unità/modulo	—	OFF/ON	Lampeggiante/ON	Stop/Continua ²	Durante l'esecuzione dell'istruzione END.
2100	SP. UNIT LAY ERR.	Num. unità/modulo	—	OFF	Lampeggiante	Stop	Accensione/reset
2103							
2106							
2107							
2110	SP UNIT ERROR	Num. unità/modulo	Posizione errore programmazione	OFF/ON	Lampeggiante/ON	Stop/Continua ²	Durante l'esecuzione dell'istruzione
2111							
2112							Durante l'esecuzione dell'istruzione STOP ® RUN

¹ Le informazioni in parentesi () indicano gli indirizzi dei registri speciali in cui sono memorizzate le singole informazioni.

² Lo stato operativo della CPU a seguito di un errore può essere configurato tramite parametri (il display a LED cambia corrispondentemente).

Descrizione errore e causa	Rimedio
<p>Dopo aver effettuato una allocazione I/O tramite parametri, un modulo funzione speciale non può essere oggetto di una comunicazione iniziale. All'occorrenza di questo errore, viene memorizzato l'indirizzo iniziale di I/O del modulo corrispondente.</p> <p>La coppia di istruzioni FROM/TO non può essere eseguita per errore nell'impulso di comando con un modulo funzione speciale. All'occorrenza di questo errore, viene memorizzata la locazione dell'errore di programma.</p> <p>È stato rilevato un errore sul bus di sistema (timeout di attesa, timeout di arbitraggio).</p> <p>Rilevato un errore sul bus di sistema.</p> <p>Rilevato guasto sul telaio base o di espansione.</p>	<p>Un modulo funzione speciale, un modulo CPU o un telaio base presenta un guasto hardware. Contattare il più vicino centro di Assistenza MITSUBISHI.</p>
<p>Rilevata interruzione momentanea dell'alimentazione.</p>	<p>Controllare l'alimentatore.</p>
<p>1.) Tensione batteria insufficiente 2.) La batteria della CPU non è collegata.</p> <p>Informazioni modulo di I/O modificate all'accensione Modulo di I/O (o modulo funzione speciale) non correttamente montato sul telaio di base.</p>	<p>1.) Sostituire la batteria. 2.) Se la batteria serve per la RAM interna della CPU o per la ritenzione dati, ponticellare la batteria con la CPU.</p> <p>Leggere le informazioni generali sull'errore sul display del terminale di programmazione e controllare il montaggio del modulo corrispondente sul telaio di base. In alternativa, controllare i registri speciali da SD150 a SD157 sul display del terminale di programmazione e controllare il montaggio dei moduli i cui bit corrispondenti sono impostati a „1„.</p>
<p>Nei parametri di configurazione per l'allocazione degli I/O, un modulo funzione intelligente è stato allocato in una posizione riservata per un modulo di I/O o viceversa. Nei parametri di configurazione per l'allocazione degli I/O, un modulo diverso da una CPU è stato allocato in una posizione riservata per un modulo di CPU o viceversa. Nessun modulo CPU allocato nelle posizioni per modulo CPU. Un commutatore per uso generale è stato impostato per un modulo che non lo prevede.</p>	<p>Correggere opportunamente la configurazione di allocazione I/O. Cancellare l'impostazione del commutatore.</p>
<p>Più di un modulo di interruzione QI60 montato sul telaio di base.</p>	<p>Montare un solo modulo QI60</p>
<p>1.) Installato più di un modulo MELSECNET/H. 2.) Installato più di un modulo Ethernet. 3.) Installati più di due moduli CC-Link. 4.) Presenti numeri di rete o numeri di stazione identici in una rete MELSECNET/H.</p>	<p>1.) Inserire un solo modulo 2.) Inserire un solo modulo 3.) Inserire un massimo di 2 moduli 4.) Controllare numeri di rete e di stazione.</p>
<p>L'indirizzo iniziale X/Y configurato corrisponde all'indirizzo iniziale X/Y di un altro modulo.</p>	<p>Cancellare i parametri di configurazione I/O e adattarli alla configurazione reale.</p>
<p>Il modulo indirizzato dalle istruzioni FROM/TO non è un modulo funzione speciale. Il modulo funzione speciale indirizzato è guasto.</p> <p>La locazione indicata dall'operando diretto di comunicazione non è un modulo di rete.</p> <p>Il modulo funzione speciale indicato non è un modulo funzione speciale o non è corretto.</p>	<p>Leggere le informazioni di errore specifiche e controllare/modificare le istruzioni FROM/TO corrispondenti (posizione errore di programmazione). In caso di modulo guasto, contattare il centro Assistenza MITSUBISHI più vicino.</p> <p>Leggere le informazioni di errore specifiche e controllare/modificare le istruzioni dedicate per il modulo funzione speciale che corrispondono al valore numerico (posizione errore di programmazione).</p>

Codici di errore da 2120 a 3004

Codice di errore (SD0) ¹	Messaggio di errore	Informazioni generali (da SD5 a 15) ¹	Informazioni specifiche (da SD13 a 20) ¹	Stato LED		Stato CPU	Temporizzazione diagnostica
				RUN	ERROR		
2120	SP. UNIT LAY ERR.	—	—	OFF	Lampeggiante	Stop	Accensione/reset
2122							
2124							
2125							
2200	MISSING PARA.	Nome drive	—	OFF	Lampeggiante	Stop	Accensione/reset
2400	FILE SET ERROR	Nome file	Numero parametro	OFF	Lampeggiante	Stop	Accensione/reset
2401							
2500	CAN'T EXE. PRG.	Nome file	—	OFF	Lampeggiante	Stop	Accensione/reset
2501							
2502							
2503							
3000	PARAMETER ERROR	Nome file	Numero parametro	OFF	Lampeggiante	Stop	Accensione/reset/ STOP © RUN
3001							
3003							
3004							

¹ Le informazioni in parentesi () indicano gli indirizzi dei registri speciali in cui sono memorizzate le singole informazioni.

² Lo stato operativo della CPU a seguito di un errore può essere configurato tramite parametri (il display a LED cambia corrispondentemente).

Descrizione errore e causa	Rimedio
Viene usato un telaio base QA[]B o QA1S[]B.	Usare un telaio base Q[]B.
Viene usato un telaio QA1S[]B come telaio base principale.	Usare un telaio Q[]B come telaio base principale.
<p>Un modulo è inserito nel 25° slot o successivo (17° o successivo per Q00JCPU).</p> <p>Un modulo è inserito in uno slot superiore al numero di slot specificato nella configurazione del telaio.</p> <p>Un modulo è installato con indirizzi di I/O superiori al campo I/O effettivo.</p> <p>Un modulo installato con gli ultimi indirizzi di I/O occupa ulteriori indirizzi.</p> <p>Collegate più di 4 telai di espansione (più di 2 telai per Q00JCPU).</p>	<p>Smontare il modulo inserito nel 25° slot o successivo (17° o successivo per Q00JCPU).</p> <p>Smontare il modulo inserito nello slot superiore al numero di slot specificato nella configurazione del telaio.</p> <p>Smontare il modulo installato con indirizzi di I/O superiori al campo I/O effettivo.</p> <p>Sostituire l'ultimo modulo con un modulo che occupa un numero di punti di I/O che non superano la configurazione attuale.</p> <p>Collegare non oltre 4 telai di espansione (2 per Q00JCPU).</p>
È stato installato un modulo non riconosciuto dalla CPU. Mancata risposta da un modulo funzione intelligente.	Installare un modulo capace di essere utilizzato dalla Q-CPU Il modulo intelligente presenta un guasto hardware. Contattare il più vicino centro di Assistenza MITSUBISHI.
Nessun file di parametri nella memoria programma.	Controllare e correggere i parametri di configurazione in base ai drive esistenti. Inserire un file parametri nel drive indicato.
Il file indicato nei parametri di configurazione PC è inesistente.	Leggere le informazioni di errore specifiche sul display del terminale di programmazione ed accertarsi che drive e destinazione del file corrispondano ai valori numerici visualizzati. Creare il file indicato.
Il file specificato come impostazione PLC-RAS non è stato creato nell'area storico errori.	Leggere le informazioni di errore specifiche sul display del terminale di programmazione ed accertarsi che drive e destinazione del file corrispondano ai valori numerici visualizzati. Controllare la memoria restante sulla memory card.
Un file di programma usa un operando che supera il campo di allocazione previsto con i parametri di configurazione operandi.	Leggere le informazioni di errore specifiche sul display del terminale di programmazione ed accertarsi che drive e destinazione del file di programma corrispondano ai valori numerici visualizzati (nome file).
Esistono più file di programma, ma il parametro di impostazione programma specifica "nessuno".	Modificare il parametro relativo al programma in "si". Cancellare i programmi non necessari.
Il file di programma non è corretto. In alternativa, il contenuto del file non corrisponde a un programma a sequenza.	Controllare che il formato del file sia *.QPG e che il file contenga effettivamente un programma a sequenza.
Non esiste alcun file.	Controllare la configurazione di programma ed i parametri.
I parametri di configurazione per impostazione limite di tempo dei timer, contatto RUN/PAUSE, indirizzo puntatore comune, numero slot vacanti per elaborazione dati generale o sistema di interrupt, superano il campo ammesso dalla CPU.	1.) Leggere le informazioni di errore dettagliate sul display del terminale di programmazione, controllare i parametri corrispondenti ai valori numerici indicati, modificando quanto necessario. 2.) Se l'errore permane, è probabile un guasto sia sulla RAM interna della CPU o sulla memory card. Contattare il più vicino centro di Assistenza MITSUBISHI.
Configurazione parametri corrotta.	
Il numero di operando impostati con i dati di configurazione operandi, supera il campo ammesso dalla CPU.	
File parametri non corretto. In alternativa, il file non contiene parametri.	Controllare che il formato del file sia *.QPA e che il file contenga effettivamente dei parametri.

Codici di errore da 3100 a 4030

Codice di errore (SD0) ¹	Messaggio di errore	Informazioni generali (da SD5 a 15) ¹	Informazioni specifiche (da SD13 a 20) ¹	Stato LED		Stato CPU	Temporizzazione diagnostica
				RUN	ERROR		
3100	LINK PARA. ERROR	Nome file	Numero parametro	OFF	Lampeggiante	Stop	Accensione/reset/ STOP ® RUN
3101							
3102							
3103							
3104							
3105							
3106							
3107							
3300	SP. PARA. ERROR	Nome file	Numero parametro	OFF	Lampeggiante	Stop	Accensione/reset/ STOP ® RUN
3301							
3302							
4000	INSTRCT CODE. ERR.	Posizione errore programmazione	—	OFF	Lampeggiante	Stop	Accensione/reset/ STOP ® RUN
4002							
4003							
4004							
4010	MISSING END INS.	Posizione errore programmazione	—	OFF	Lampeggiante	Stop	Accensione/reset/ STOP ® RUN
4021	CAN'T SET (P)						
4030	CAN'T SET (I)						

¹ Le informazioni in parentesi () indicano gli indirizzi dei registri speciali in cui sono memorizzate le singole informazioni.

Descrizione errore e causa	Rimedio
<p>1.) Il numero di moduli effettivamente installati è diverso da quello specificato dal parametro di impostazione numero di moduli di MELSECNET/H</p> <p>2.) L'indirizzo iniziale di I/O dei moduli effettivamente installati è diverso da quello specificato dal parametro di impostazione della rete MELSECNET/H</p> <p>3.) Alcuni dati di configurazione non possono essere gestiti.</p> <p>4.) Il tipo di stazione di MELSECNET/H è stato modificato con sistema alimentato. (È necessario un passaggio da RESET a RUN per modificare il tipo di stazione).</p>	<p>1.) Riscrivere i parametri di rete corretti.</p> <p>2.) Se l'errore permane, contattare il Centro Assistenza MITSUBISHI più vicino</p>
<p>Il numero di rete specificato da un parametro è diverso da quello della rete attualmente installata.</p> <p>L'indirizzo di I/O specificato da un parametro è diverso da quello delle unità di I/O attualmente installate.</p> <p>La classe di rete specificata da un parametro è diversa da quella della rete attualmente installata.</p> <p>Il parametro di rinfresco di rete di MELSECNET/H è oltre il campo specificato.</p>	<p>Far corrispondere i parametri specificati con quelli effettivi relativi alla rete e unità installate.</p>
<p>Rilevato un errore durante il controllo dei parametri di rete su un modulo di rete.</p>	<p>1.) Riscrivere i parametri di rete corretti.</p> <p>2.) Se l'errore permane, contattare il Centro Assistenza MITSUBISHI più vicino.</p>
<p>Anche se il numero di unità Ethernet configurate con parametri è maggiore di zero, il numero di unità attualmente montate è zero.</p> <p>L'indirizzo iniziale di I/O nei parametri di configurazione Ethernet è diverso da quello delle unità di I/O effettivamente montate.</p>	
<p>Ethernet e MELSECNET/H usano lo stesso numero di rete.</p> <p>Numero di rete, numero di stazione e numero di gruppo impostati sono fuori campo ammesso.</p> <p>L'indirizzo di I/O è fuori del campo ammesso per la CPU usata.</p> <p>La configurazione dei parametri specifici per Ethernet non è normale.</p> <p>La parametrizzazione specifica per Ethernet non è normale.</p>	
<p>Anche se il numero di unità CC-Link configurate con parametri è maggiore di zero, il numero di unità attualmente montate è zero.</p> <p>L'indirizzo di I/O specificato dai parametri comuni è diverso da quello delle unità di I/O attualmente installate.</p> <p>La classe di stazione per una unità CC-Link impostata nei parametri non corrisponde a quella della stazione effettivamente montata.</p>	
<p>Il parametro di rinfresco della rete CC-Link è fuori campo ammesso.</p> <p>Il contenuto del parametro CC-Link non è corretto.</p>	
<p>L'indirizzo iniziale di I/O del modulo funzione intelligente impostato nei parametri con GX Configurator è diverso dall'indirizzo di I/O effettivo.</p> <p>Il campo di rinfresco del modulo funzione intelligente supera la capacità del file registri.</p> <p>L'impostazione del parametro di rinfresco del modulo funzione intelligente è oltre il campo ammesso.</p> <p>I parametri di rinfresco del modulo funzione intelligente non sono corretti.</p>	<p>Controllare i parametri impostati</p>
<p>Il programma contiene un codice di istruzione non decodificabile.</p> <p>Il programma contiene una istruzione non utilizzabile.</p> <p>Il nome istruzione non è corretto.</p> <p>L'istruzione indica un numero di operandi non corretto.</p> <p>L'istruzione specifica un operando che non si può usare.</p>	<p>Leggere le informazioni di errore generali sul display del terminale di programmazione e controllare che i file corrispondano con i valori indicati (locazione dell'errore di programmazione).</p>
<p>Il programma non contiene nessuna istruzione END (FEND).</p>	
<p>Gli indirizzi del puntatore comune usato dai singoli file sono sovrapposti.</p>	
<p>Gli indirizzi del puntatore di allocazione usato dai singoli file sono sovrapposti.</p>	

Codici di errore da 4100 a 900

Codice di errore (SD0) ¹	Messaggio di errore	Informazioni generali (da SD5 a 15) ¹	Informazioni specifiche (da SD13 a 20) ¹	Stato LED		Stato CPU	Temporizzazione diagnostica
				RUN	ERROR		
4100	ERRORE DI ESECUZIONE	Posizione errore programmazione	—	OFF/ON	Lampeggiante/ON	Stop/Continua ²	Durante l'esecuzione di un'istruzione
4101							
4102							
4108							
4200	FOR NEXT ERROR	Posizione errore programmazione	—	OFF	Lampeggiante	Stop	Durante l'esecuzione di un'istruzione
4201							
4202							
4203							
4210	CAN'T EXECUTE (P)	Posizione errore programmazione	—	OFF	Lampeggiante	Stop	Durante l'esecuzione di un'istruzione
4211							
4212							
4213							
4220	CAN'T EXECUTE (I)	Posizione errore programmazione	—	OFF	Lampeggiante	Stop	Durante l'esecuzione di un'istruzione
4221							
4223							
4231	INST. FORMAT ERR	Posizione errore programmazione	—	OFF	Lampeggiante	Stop	Durante l'esecuzione di un'istruzione
5001	WDT ERROR	Tempo (valore impostato)	Tempo (valore effettivo)	OFF	Lampeggiante	Stop	Sempre
5010	PRG. TIME OVER	Tempo (valore impostato)	Tempo (valore effettivo)	ON	ON	Continua	Sempre
9000	F**** ³	Posizione errore programmazione	Numero spia di errore	ON	OFF	Continua	Durante l'esecuzione di un'istruzione
				USER LED ON			

¹ Le informazioni in parentesi () indicano gli indirizzi dei registri speciali in cui sono memorizzate le singole informazioni.

² Lo stato operativo della CPU a seguito di un errore può essere configurato tramite parametri (il display a LED cambia corrispondentemente).

³ **** indica il numero della spia di errore.

Descrizione errore e causa	Rimedio
<p>L'istruzione non può processare i dati contenuti.</p> <p>Gli indirizzi degli operandi specificati per i dati elaborati dall'istruzione superano i rispettivi campi ammessi. In alternativa, i dati memorizzati o le costanti per gli operandi utilizzati dall'istruzione superano i campi ammessi.</p> <p>Il numero di rete o numero di stazione per una istruzione dedicata di rete non è corretto. L'operando diretto di comunicazione non è impostato correttamente.</p> <p>I parametri CC-Link non sono impostati quando l'istruzione CC-Link viene eseguita.</p>	<p>Leggere le informazioni di errore generali sul display del terminale di programmazione e controllare il passo di programma indicato.</p> <p>Eseguire l'istruzione CC-Link dopo aver impostato i parametri CC-Link.</p>
<p>Nessuna istruzione NEXT è stata eseguita dopo l'esecuzione di una istruzione FOR. In alternativa, esistono meno istruzioni NEXT rispetto alle istruzioni FOR.</p> <p>Viene eseguita una istruzione NEXT senza una precedente istruzione FOR. In alternativa, esistono più istruzioni NEXT rispetto alle istruzioni FOR.</p> <p>Programmati più di 16 livelli di annidamento.</p> <p>Viene eseguita una istruzione BREAK senza una precedente istruzione FOR.</p>	<p>Leggere le informazioni di errore generali sul display del terminale di programmazione e controllare il passo di programma indicato.</p> <p>Ridurre il numero di livelli di annidamento fino a un massimo di 16.</p>
<p>Viene eseguita una istruzione CALL, ma non esiste nessun sottoprogramma in corrispondenza del puntatore specificato.</p> <p>Il sottoprogramma non contiene nessuna istruzione RET.</p> <p>L'istruzione RET è stata programmata prima dell'istruzione FEND.</p> <p>Programmati più di 16 livelli di annidamento.</p>	<p>Leggere le informazioni di errore generali sul display del terminale di programmazione e controllare il passo di programma indicato.</p> <p>Ridurre il numero di livelli di annidamento fino a un massimo di 16.</p>
<p>È stata generata una interruzione, ma non è stato trovato un puntatore di interruzione corrispondente.</p> <p>Il sottoprogramma non contiene nessuna istruzione IRET.</p> <p>L'istruzione IRET è stata programmata prima dell'istruzione FEND.</p>	<p>Leggere le informazioni di errore generali sul display del terminale di programmazione e controllare il passo di programma indicato.</p>
<p>Le istruzioni IX e IXEND non sono programmate in coppia. I numeri delle istruzioni IX e IXEND non combaciano.</p>	
<p>Il tempo di scansione programma supera il valore WDT specificato dal parametro PC RAS.</p>	<p>Leggere le informazioni di errore specifiche sul display del terminale di programmazione e controllare il valore numerico (tempo) riportato, abbreviando se necessario il tempo di scansione.</p>
<p>Il tempo di esecuzione di un programma a bassa velocità impostato nel parametro di configurazione PC RAS supera il margine di tempo della scansione costante.</p>	<p>Controllare e modificare il tempo di scansione costante ed il tempo di esecuzione del programma a bassa velocità.</p>
<p>La spia di errore F è stata attivata.</p>	<p>Leggere le informazioni di errore specifiche sul display del terminale di programmazione e controllare il programma corrispondente al valore numerico riportato (numero spia di errore).</p>

13.2 Tabella dei codici di errore; CPU QnA e System Q

La tabella seguente contiene una panoramica di tutti i possibili errori ed i messaggi di errore corrispondenti, le cause possibili e le azioni consigliate. Sono elencati soli i messaggi di errore per le CPU Q02(H), Q06H, Q12H, Q25H, QnA, QnAS e Q4AR. Il simbolo „●“ nella colonna delle CPU corrispondenti indica che l'errore si applica a tutti i tipi di CPU sopra menzionati. La sigla „Rem“ indica compatibilità con i moduli di I/O remoti. Una sigla di CPU in questa colonna indica che l'errore si applica solo al tipo di CPU specificato.☆

Codice di errore (SD0) ¹	Messaggio di errore	Informazioni generali (da SD5 a 15) ¹	Informazioni specifiche (da SD13 a 20) ¹	Stato LED		Stato CPU	Temporizzazione diagnostica
				RUN	ERROR		
1000	MAIN CPU DOWN	—	—	OFF	Lampeggiante/ON	Stop	Sempre
1001							
1002							
1003							
1004							
1005							
1006							
1007							
1008							
1009							
1010	END NOT EXECUTE	—	—	OFF	Lampeggiante	Stop	Durante l'esecuzione dell'istruzione END.
1011							
1012							
1101	RAM ERROR	—	—	OFF	Lampeggiante	Stop	Accensione/reset
1102							
1103							
1104							
1105							
1200	OPE. CIRCUIT ERR.	—	—	OFF	Lampeggiante	Stop	Accensione/reset
1201							
1202							
1203							
1204							
1205							
1206							
							Durante l'esecuzione dell'istruzione END.
							Durante l'esecuzione di un'istruzione

¹ Le informazioni in parentesi () indicano gli indirizzi dei registri speciali in cui sono memorizzate le singole informazioni.

Tabella dei codici di errore; CPU QnA e System Q

	Descrizione errore e causa	Rimedio	Valido per:		
	<p>Modo RUN sospeso o guasto della CPU principale</p> <p>1.) Malfunzionamento dovuto a rumori o altre cause</p> <p>2.) Guasto hardware</p>	<p>1.) Misurare il livello di rumore.</p> <p>2.) Resettare la CPU e tornare in modo RUN.</p> <p>Se si ripresenta di nuovo lo stesso errore, è possibile che la CPU sia guasta. Contattare il più vicino centro di Assistenza MITSUBISHI.</p>	CPU QnA		
	<p>Modo RUN sospeso o guasto della CPU principale</p> <p>1.) Malfunzionamento dovuto a rumori o altre cause</p> <p>2.) Guasto hardware</p>	<p>1.) Misurare il livello di rumore.</p> <p>2.) Resettare la CPU e tornare in modo RUN.</p> <p>Se si ripresenta di nuovo lo stesso errore, è possibile che la CPU sia guasta. Contattare il più vicino centro di Assistenza MITSUBISHI.</p>	Q CPU		
			Q CPU/ Rem		
			Q CPU		
	<p>L'intero programma è stato eseguito senza incontrare una istruzione END.</p> <p>1.) Quando è stata eseguita l'istruzione END è stata letta come un diverso codice istruzione.</p> <p>2.) L'istruzione END è stata alterata dal codice di un'altra istruzione.</p>	<p>1.) Misurare il livello di rumore.</p> <p>2.) Resettare la CPU e tornare in modo RUN.</p> <p>Se si ripresenta di nuovo lo stesso errore, è possibile che la CPU sia guasta. Contattare il più vicino centro di Assistenza MITSUBISHI.</p>			
	<p>Errore nella RAM interna della CPU contenente il programma di sequenza.</p> <p>Errore nella RAM usata come area di lavoro della CPU.</p> <p>Errore interno della CPU</p> <p>Errore indirizzamento RAM della CPU.</p>	<p>Questo indica un guasto hardware della CPU.</p> <p>Contattare il più vicino centro di Assistenza MITSUBISHI.</p>	U		
	<p>Memoria condivisa CPU guasta</p>			<p>1.) Misurare il livello di rumore.</p> <p>2.) Resettare la CPU e tornare in modo RUN. Se si ripresenta di nuovo lo stesso errore, è possibile che la CPU sia guasta. Contattare il più vicino centro di Assistenza MITSUBISHI.</p>	Q CPU (Vers. B o successiva)
	<p>Malfunzionamento della circuiteria che esegue l'indicizzazione interna alla CPU.</p> <p>Malfunzionamento hardware CPU (logica)</p> <p>Malfunzionamento della circuiteria che esegue l'elaborazione della sequenza.</p>	<p>Questo indica un guasto hardware della CPU.</p> <p>Contattare il più vicino centro di Assistenza MITSUBISHI.</p>	U		
	<p>Malfunzionamento della circuiteria che esegue l'indicizzazione interna alla CPU.</p> <p>Malfunzionamento hardware CPU (logica)</p> <p>Malfunzionamento della circuiteria che esegue l'elaborazione della sequenza.</p>			<p>Questo indica un guasto hardware della CPU.</p> <p>Contattare il più vicino centro di Assistenza MITSUBISHI.</p>	Q4AR CPU
	<p>La circuiteria per la funzione DSP della CPU non funziona correttamente.</p>				

Tabella dei codici di errore; CPU QnA e System Q

Codici di errore da 1300 a 1413

Codice di errore (SD0) ¹	Messaggio di errore	Informazioni generali (da SD5 a 15) ¹	Informazioni specifiche (da SD13 a 20) ¹	Stato LED		Stato CPU	Temporizzazione diagnostica		
				RUN	ERROR				
1300	FUSE BREAK OFF	Num. unità/modulo	—	OFF/ON	Lampeggiante/ON	Stop/Continua ²	Durante l'esecuzione dell'istruzione END.		
1301	EX POWER OFF	Num. unità/modulo	—	OFF/ON	Lampeggiante/ON	Stop/Continua ²	Durante l'esecuzione dell'istruzione END.		
1310	I/O INT ERROR	Num. unità/modulo	—	OFF	Lampeggiante/ON	Stop	Durante una interruzione		
1401	SP. UNIT DOWN	Num. unità/modulo	—	OFF/ON	Lampeggiante/ON	Stop/Continua ³	Accensione/reset Accedendo a un modulo funzione intelligente.		
1402			Posizione errore programmazione				Lampeggiante/ON	Stop/Continue ⁴	Accensione/reset
1403			—						Accedendo a un modulo funzione intelligente.
1411	CONTROL-BUS ERR.	Num. unità/modulo	Posizione errore programmazione	OFF	Lampeggiante	Stop	Durante l'esecuzione della coppia di istruzioni FROM/TO.		
1412							Durante l'esecuzione dell'istruzione END.		
1413	CONTROL-BUS ERR.	—	—	OFF	Lampeggiante	Stop	Sempre		

¹ Le informazioni in parentesi () indicano gli indirizzi dei registri speciali in cui sono memorizzate le singole informazioni.

² Lo stato operativo della CPU a seguito di un errore può essere configurato tramite parametri (il display a LED cambia corrispondentemente).

³ Questo errore si verifica solo nei sistemi ridondanti. Può essere rilevato sia nel sistema attivo che in quello in standby.

⁴ Per ciascun modulo è possibile configurare il funzionamento stop/continua tramite parametri.

Tabella dei codici di errore; CPU QnA e System Q

	Descrizione errore e causa	Rimedio	Valido per:
	Fusibile interrotto nel modulo di uscita	1.) Controllare i LED di errore dei moduli di uscita e sostituire il modulo con il LED acceso. 2.) I moduli con fusibile interrotto possono anche essere controllati con un terminale di programmazione. Controllare se nei registri speciali da SD1300 a SD1331 ci sono bit attivi (1) che corrispondono ai moduli con fusibile interrotto.	Q CPU Rem
	Fusibile interrotto nel modulo di uscita	1.) Controllare i LED di segnalazione fusibile interrotto sui moduli di uscita e sostituire i fusibili relativi. 2.) Leggere le informazioni di errore generali sul display del terminale di programmazione e sostituire i fusibili dei moduli indicati. In alternativa, controllare i registri speciali da SD1300 a SD1331 sul display del terminale di programmazione e sostituire i fusibili dei moduli con i bit corrispondenti attivi („1“).	CPU QnA, Q4AR
	Fusibile interrotto nel modulo di uscita L'alimentazione esterna per il carico delle uscite è spenta o scollegata.	1.) Controllare i LED di segnalazione fusibile interrotto sui moduli di uscita e sostituire i fusibili relativi. 2.) Leggere le informazioni di errore generali sul display del terminale di programmazione e sostituire il fusibile dei moduli di uscita indicati. In alternativa, controllare i registri speciali da SD1300 a SD1331 sul display del terminale di programmazione e sostituire il fusibile dei moduli con i bit corrispondenti attivi („1“). 3.) Controllare se l'alimentazione esterna per il carico delle uscite è scollegata.	Q2AS CPU
	L'alimentazione esterna per il carico delle uscite è spenta o scollegata. (per espansione futura).	Controllare il carico delle uscite sull'alimentatore esterno.	Q CPU Rem
	Si è verificata una interruzione, anche se non esistono moduli con interruzione nel sistema.	Uno dei moduli collegati presenta un guasto hardware. Controllare i moduli collegati. Contattare il più vicino centro di Assistenza MITSUBISHI.	●
	1.) Mancata risposta da un modulo funzione intelligente durante le comunicazioni iniziali. 2.) La dimensione del buffer di memoria del modulo funzione intelligente non è normale.	Il modulo CPU presenta un guasto hardware. Contattare il più vicino centro di Assistenza MITSUBISHI.	Q CPU Rem
	Dopo l'allocazione di un parametro di I/O, non c'è stata nessuna risposta dal modulo funzione speciale, durante la comunicazione iniziale. All'occorrenza di questo errore, viene memorizzato l'indirizzo iniziale di I/O del modulo corrispondente.	Il modulo interrogato presenta un guasto hardware. Contattare il più vicino centro di Assistenza MITSUBISHI.	CPU QnA
	Il programma ha interrogato un modulo funzione speciale, senza ottenere risposta.	Il modulo CPU presenta un guasto hardware. Contattare il più vicino centro di Assistenza MITSUBISHI.	Q CPU Rem
	È stato interrogato un modulo funzione speciale con una istruzione FROM/TO, senza ottenere nessuna risposta. All'occorrenza di questo errore, viene memorizzata la locazione dell'errore di programma.	Il modulo interrogato presenta un guasto hardware. Contattare il più vicino centro di Assistenza MITSUBISHI.	CPU QnA
	1.) Mancata risposta da un modulo funzione intelligente durante l'esecuzione dell'istruzione END. 2.) Rilevato errore sul modulo funzione intelligente.		Q CPU Rem
	Dopo aver effettuato una allocazione I/O tramite parametri, un modulo funzione speciale non può essere oggetto di una comunicazione iniziale. All'occorrenza di questo errore, viene memorizzato l'indirizzo iniziale di I/O del modulo corrispondente.	Un modulo funzione speciale, un modulo CPU o un telaio base presenta un guasto hardware. Contattare il più vicino centro di Assistenza MITSUBISHI.	● Rem
	La coppia di istruzioni FROM/TO non può essere eseguita per errore nell'impulso di comando con un modulo funzione speciale. All'occorrenza di questo errore, viene memorizzata la locazione dell'errore di programma.		●
	Una CPU System Q con versione funzionale A viene usata in un sistema multi CPU.	1.) Sostituire il modulo CPU con uno di versione funzionale B o successiva. 2.) Un modulo funzione speciale, un modulo CPU o un telaio base presenta un guasto hardware. Contattare il più vicino centro di Assistenza MITSUBISHI.	Q CPU (Vers. B o successiva)
	È stato rilevato un errore sul bus di sistema (timeout di attesa, timeout di arbitraggio).	Un modulo funzione speciale, un modulo CPU o un telaio base presenta un guasto hardware. Contattare il più vicino centro di Assistenza MITSUBISHI.	Q CPU Rem

Tabella dei codici di errore; CPU QnA e System Q

Codici di errore da 1414 a 2101

Codice di errore (SD0) ¹	Messaggio di errore	Informazioni generali (da SD5 a 15) ¹	Informazioni specifiche (da SD13 a 20) ¹	Stato LED		Stato CPU	Temporizzazione diagnostica
				RUN	ERROR		
1414	CONTROL-BUS ERR.	Num. unità/modulo	—	OFF	Lampeggiante	Stop	Durante l'esecuzione dell'istruzione END.
1415		Numero telaio					
1416		Num. unità/modulo					
1421	SYS. UNIT DOWN ³	—	—	OFF	Lampeggiante	Stop	Sempre
1500	AC DOWN	—	—	ON	OFF	Continua	Sempre
1510	DUAL DC DOWN 5V ⁴	—	—	ON	ON	Continua	Sempre
1520	DC DOWN 5V ⁵	—	—	OFF	Lampeggiante	Stop	Sempre
1530	DC DOWN 24V ³	—	—	ON	ON	Continua	Sempre
1600	ERRORE BATTERIA	Nome drive	—	ON	OFF	Continua	Sempre
1601				BAT. ALM LED ON			
1602				BAT. ALM LED ON			
2000	UNIT VERFIY ERR.	Num. unità/modulo	—	OFF/ON	Lampeggiante/ON	Stop/Continua ²	Durante l'esecuzione dell'istruzione END.
2100	SP. UNIT LAY ERR.	Num. unità/modulo	—	OFF	Lampeggiante	Stop	Accensione/reset
2101							

¹ Le informazioni in parentesi () indicano gli indirizzi dei registri speciali in cui sono memorizzate le singole informazioni.

² Lo stato operativo della CPU a seguito di un errore può essere configurato tramite parametri (il display a LED cambia corrispondentemente).

³ Questo errore si verifica solo nei sistemi ridondanti. Può essere rilevato sia nel sistema attivo che in quello in standby.

⁴ Questo errore si verifica solo nei sistemi ridondanti.

⁵ Questo errore si può verificare sia in un sistema standalone che nel sistema attivo di un sistema ridondante..

Tabella dei codici di errore; CPU QnA e System Q

Descrizione errore e causa	Rimedio	Valido per:
Malfunzionamento in uno dei moduli installati. Una CPU System Q con versione funzionale A viene usata in un sistema multi CPU.	1.) Sostituire il modulo CPU con uno di versione funzionale B o successiva. 2.) Un modulo funzione speciale, un modulo CPU o un telaio base presenta un guasto hardware. Contattare il più vicino centro di Assistenza MITSUBISHI.	Q CPU (Vers. B o successiva)
Rilevato un errore sul bus di sistema.	Un modulo funzione speciale, un modulo CPU o un telaio base presenta un guasto hardware. Contattare il più vicino centro di Assistenza MITSUBISHI.	Q CPU Rem
Rilevato guasto sul telaio principale o di espansione.	Un modulo funzione speciale, un modulo CPU o un telaio base presenta un guasto hardware. Contattare il più vicino centro di Assistenza MITSUBISHI.	Q CPU (Vers. B o successiva)
Rilevato un errore di bus durante l'accensione o reset.		
Guasto hardware sul modulo di gestione sistema AS92R.	Contattare il più vicino centro di Assistenza MITSUBISHI.	Q4AR CPU
Rilevata interruzione momentanea dell'alimentazione.	Controllare l'alimentatore.	U/Rem
La tensione 5 V DC di uno dei due alimentatori dei telai di espansione di un sistema ridondante è scesa sotto al 85 % della tensione nominale.	Controllare la tensione del modulo alimentatore. Sostituire il modulo alimentatore se la tensione è inferiore al valore nominale.	Q4AR CPU
La tensione 5 V DC di uno dei moduli alimentatori del telaio di espansione è scesa sotto al 80 % della tensione nominale.		
La tensione 24 V CC fornita al modulo di gestione sistema AS92R è scesa sotto al 85% del valore nominale.		
1.) Tensione batteria CPU insufficiente 2.) Modulo batteria CPU non collegato.	1.) Sostituire la batteria. 2.) Se la batteria serve per la RAM interna della CPU o per la ritenzione dati inserire un ponticello.	U
Tensione insufficiente sulla batteria della memory card 1.	Sostituire la batteria.	U
Tensione insufficiente sulla batteria della memory card 2.		Q4AR CPU
Informazioni modulo di I/O modificate all'accensione Modulo di I/O (o modulo funzione speciale) non correttamente montato sul telaio di base.	Leggere le informazioni comuni sull'errore sul display del terminale di programmazione e controllare il montaggio del modulo corrispondente sul telaio di base. In alternativa, controllare i registri speciali da SD1400 a SD1431 sul display del terminale di programmazione e controllare il montaggio dei moduli i cui bit corrispondenti sono impostati a „1„.	U Rem
Una CPU System Q con versione funzionale A viene usata in un sistema multi CPU.	Sostituire il modulo CPU con uno di versione funzionale B o successiva.	Q CPU (Vers. B o successiva)
Uno slot in cui è inserito un modulo di interrupt QI60 è configurato per un modulo diverso da un modulo funzione o modulo di interrupt.	Modificare la configurazione per rispecchiare la situazione reale.	Q CPU (Vers. B o successiva)
Nei parametri di configurazione per l'allocazione degli I/O, un modulo funzione intelligente è stato allocato in una posizione riservata per un modulo di I/O o viceversa. Nei parametri di configurazione per l'allocazione degli I/O, un modulo diverso da una CPU è stato allocato in una posizione riservata per un modulo di CPU o viceversa. Nessun modulo CPU allocato nelle posizioni per modulo CPU. Un commutatore per uso generale è stato impostato per un modulo che non lo prevede.	Correggere opportunamente la configurazione di allocazione I/O. Cancellare l'impostazione del commutatore.	Q CPU Rem
Nei parametri di configurazione per l'allocazione degli I/O, un modulo funzione speciale è stato allocato in una posizione riservata per un modulo di I/O o viceversa.	Correggere opportunamente la configurazione di allocazione I/O.	Q4AR CPU
Inseriti più di 12 moduli funzione speciali (tranne A1S161 e QI60) capaci di generare una interruzione per la CPU installata.	Ridurre il numero di moduli funzione speciali installati (tranne A1S161 e QI60) a 12 o meno.	Q CPU
Inseriti più di 12 moduli funzione speciali (tranne A161) capaci di generare una interruzione per la CPU installata.	Ridurre il numero di moduli funzione speciali installati (tranne A161) a 12 o meno.	Q4AR CPU

Tabella dei codici di errore; CPU QnA e System Q

Codici di errore da 2102 a 2109

Codice di errore (SD0) ¹	Messaggio di errore	Informazioni generali (da SD5 a 15) ¹	Informazioni specifiche (da SD13 a 20) ¹	Stato LED		Stato CPU	Temporizzazione diagnostica
				RUN	ERROR		
2102	SP. UNIT LAY ERR.	Num. unità/modulo	—	OFF	Lampeggiante	Stop	Accensione/reset
2103							
2104							
2105							
2106	SP. UNIT LAY ERR.	Num. unità/modulo	—	OFF	Lampeggiante	Stop	Accensione/reset
2107							
2108							
2109 ²							

¹ Le informazioni in parentesi () indicano gli indirizzi dei registri speciali in cui sono memorizzate le singole informazioni.

² Questo errore si verifica solo nel sistema in standby di un sistema ridondante.

Tabella dei codici di errore; CPU QnA e System Q

Descrizione errore e causa	Rimedio	Valido per:
Più di 6 moduli A1SD51S installati.	Ridurre il numero di moduli A1SD51S a 6 o meno.	Q CPU
Più di 6 moduli di comunicazione seriale installati (tranne A(1S)J71QC24).	Ridurre il numero di moduli di comunicazione seriale (tranne A(1S)J71QC24) a 6 o meno	CPU QnA Rem
1.) Più di un modulo di interruzione QI60 o A1SI61 installato in un sistema a singola CPU. 2.) Configurato più di un modulo QI60/A1SI61 per la stessa CPU attiva in un sistema multi CPU. 3.) Più di un modulo A1SI61 installato in un sistema multi CPU.	1.) Installare un solo modulo QI60 o A1SI61 in un sistema a singola CPU. 2.) Ciascuna CPU può essere associata a un solo modulo QI60 o A1SI61. 3.) Installare un solo modulo A1SI61 in un sistema multi CPU. Usare il modulo QI60 se ciascuna CPU di un sistema multi CPU deve controllare un modulo di interruzione. (È possibile usare una combinazione di un modulo A1SI61 con 3 QI60, oppure usare solo moduli QI60).	Q CPU (Vers. B o successiva)
Più di un modulo di interruzione A1SI61 o QI60 montato sul telaio di base.	Montare un solo modulo A1SI61 o QI60	Q CPU
Più di un modulo di interruzione A1SI61 montato sul telaio di base.	Montare un solo modulo A1SI61.	CPU QnA
Nella configurazione del parametro di rinfresco automatico MELSECNET/MINI per alcuni moduli della rete non corrisponde con i moduli effettivamente installati.	Cancellare e correggere la configurazione di rinfresco automatico MELSECNET/MINI.	CPU QnA
Superato il numero massimo di moduli funzione speciali che eseguono istruzioni dedicate, assegnati a una CPU (massimo = 1344). (numero di AD59 installati x 5) (numero di AD57(S1)/AD58 installati x 8) (numero di AJ71C24(S3/S6/S8 installati x 10) (numero di AJ71UC24 installati x 10) (numero di AJ71C21(S1) installati x 29) (numero di AJ71PT32(S3) installati x 125)* (numero di AJ71QC24 (R2, R4) installati x 29) (numero di AJ71ID1 (2)-R4 installati x 18) (numero di AD75 installati x 12) Il totale deve essere uguale o inferiore a 1344 *: Se utilizzato il modo espansione.	Ridurre il numero di moduli funzione speciali installati.	CPU QnA
1.) Più di quattro moduli MELSECNET/H installati un un sistema multi CPU. Più di quattro moduli ETHERNET System Q installati un un sistema multi CPU.	Inserire 5 moduli al massimo	Q CPU (Vers. B o successiva)
1.) Installato più di quattro moduli MELSECNET/H. 2.) Installati più di quattro moduli ETHERNET System Q. 3.) Presenti numeri di rete o numeri di stazione identici in una rete MELSECNET/H.	1.) Inserire 4 moduli al massimo 2.) Inserire 4 moduli al massimo 3.) Controllare numeri di rete e di stazione.	Q CPU Rem
1.)Più di quattro AJ71QLP21 o AJ71QBR11 installati nel sistema. 2.)Più di due AJ71AP21/R21 o AJ71AT21B installati nel sistema. 3.)Più di quattro AJ71QLP21, AJ71QBR11, AJ71AP21/R21, o AJ71AT21B installati nel sistema. 4.)Presenti numeri di rete o numeri di stazione identici in una rete MELSECNET/10. 5.)Più di una stazione master o stazione locale in una rete MELSECNET (II) o MELSECNET/B.	1.) Inserire 4 moduli al massimo 2.) Inserire 2 moduli al massimo 3.) Inserire 4 moduli al massimo 4.) Controllare numeri di rete e di stazione. 5.) Controllare i numeri di stazione	CPU QnA
L'indirizzo iniziale X/Y configurato corrisponde all'indirizzo iniziale X/Y di un altro modulo.	Cancellare i parametri di configurazione I/O e adattarli alla configurazione reale.	U Rem
Installati moduli di rete A1SJ71LP21, A1SJ71BR11, A1SJ71AP21, A1SJ71AR21 o A1SJ71AT2 adatti per l'uso con CPU A2US. Installati moduli di rete A1SJ71QLP21 o A1SJ71QBR11 adatti per l'uso con CPU Q2AS.	Sostituire i moduli con un QJ71LP21 o QJ71BR11.	Q CPU
Installati moduli A(1S)J71LP21 o A(1S)J71BR11 adatti per l'uso con una CPU AnU.	Sostituire i moduli con un A(1S)J71QLP21 o A(1S)J71QBR11.	CPU QnA
Le configurazioni del sistema attivo e in standby sono diverse con il sistema ridondante in modo backup.	Controllare la configurazione del sistema in standby.	Q4AR CPU

Tabella dei codici di errore; CPU QnA e System Q

Codici di errore da 2110 a 2150

Codice di errore (SD0) ¹	Messaggio di errore	Informazioni generali (da SD5 a 15) ¹	Informazioni specifiche (da SD13 a 20) ¹	Stato LED		Stato CPU	Temporizzazione diagnostica
				RUN	ERROR		
2110	SP UNIT ERROR	Num. unità/modulo	Posizione errore programmazione	OFF/ON	Lampeggiante/ON	Stop/Continua ²	Durante l'esecuzione della coppia di istruzioni FROM/TO.
2111							
2112	SP UNIT ERROR	Num. unità/modulo	Posizione errore programmazione	OFF/ON	Lampeggiante/ON	Stop/Continua ²	Durante l'esecuzione della coppia di istruzioni FROM/TO.
2113		FFFFH (fisso)					
2114	SP UNIT ERROR	Num. unità/modulo	Posizione errore programmazione	Lampeggiante/ON	Lampeggiante/ON	Stop/Continua	Durante l'esecuzione di un'istruzione
2115							
2116							
2117							
2120	SP. UNIT LAY ERR.	—	—	OFF	Lampeggiante	Stop	Accensione/reset
2121							
2122							
2124							
2125							
2126							
2150	SP.UNIT VER. ERR.	Num. unità/modulo	—	OFF	Lampeggiante	Stop	Accensione/reset

¹ Le informazioni in parentesi () indicano gli indirizzi dei registri speciali in cui sono memorizzate le singole informazioni.

² Lo stato operativo della CPU a seguito di un errore può essere configurato tramite parametri (il display a LED cambia corrispondentemente).

Tabella dei codici di errore; CPU QnA e System Q

	Descrizione errore e causa	Rimedio	Valido per:
	È stata specificata una CPU inesistente in una istruzione per l'accesso alla memoria condivisa di una CPU.	Leggere le informazioni di errore specifiche, poi controllare il programma che corrisponde al valore numerico riportato (locazione errore di programma).	Q CPU (Vers. B o successiva)
	Il modulo indirizzato dalle istruzioni FROM/TO non è un modulo funzione speciale. Il modulo funzione speciale indirizzato è guasto.	Leggere le informazioni di errore specifiche e controllare/modificare le istruzioni FROM/TO corrispondenti (posizione errore di programmazione).	U
	La locazione indicata dall'operando diretto di comunicazione non è un modulo di rete.	In caso di modulo guasto, contattare il centro Assistenza MITSUBISHI più vicino.	
	Il modulo funzione speciale indicato non è un modulo funzione speciale o non è corretto. Il numero di rete indicato non esiste, oppure il modulo di rete non è quello corretto.	Leggere le informazioni di errore specifiche e controllare/modificare le istruzioni dedicate per il modulo funzione speciale (modulo di rete) che corrispondono al valore numerico (posizione errore di programmazione).	U Rem
	Il modulo specificato in una istruzione dedicata di rete non è un modulo di rete, oppure la rete target non esiste.		U
	La CPU che sta eseguendo l'istruzione è stata specificata in una istruzione che si deve riferire ad una CPU diversa.	Leggere le informazioni di errore specifiche, poi controllare e modificare il programma che corrisponde al valore numerico riportato (locazione errore di programma).	Q CPU (Versione B o successiva)
	È stata specificata una CPU diversa per una istruzione che richiede la specifica della CPU che sta eseguendo l'istruzione.		
	Una istruzione fa riferimento a un modulo funzione intelligente controllato da un'altra stazione.		
	In una istruzione dedicata per sistemi multi CPU è stata specificata una CPU non consentita.		
	La posizione di Q[JB e QA1S[JB non è corretta.	Controllare la posizione del telaio di base.	Q CPU Rem
	Il modulo CPU non è installato nello slot per la CPU o negli slot da 0 a 2.	Controllare la posizione del modulo CPU.	
	Viene usato un telaio QA1S[] come telaio base principale.	Usare un telaio Q[JB come telaio base principale.	
	Un modulo è installato nel 65° slot o superiore. Un modulo è inserito in uno slot superiore al numero di slot specificato nella configurazione del telaio. Un modulo è installato con indirizzi di I/O superiori al 4096° punto. Un modulo installato al 4096° indirizzo occupa punti successivi.	Smontare il modulo installato nel 65° slot o superiore. Smontare il modulo inserito nello slot superiore al numero di slot specificato nella configurazione del telaio. Smontare il modulo installato con indirizzi di I/O superiori al 4096°. Sostituire l'ultimo modulo con uno che non oltrepassi il 4096° indirizzo.	
	È stato installato un modulo non riconosciuto dalla CPU System Q. Mancata risposta da un modulo funzione intelligente.	Installare un modulo capace di essere utilizzato dalla CPU System Q. Il modulo intelligente presenta un guasto hardware. Contattare il più vicino centro di Assistenza MITSUBISHI.	
	1.) Slot libero fra i moduli CPU di un sistema multi CPU. 2.) Fra due moduli CPU è stato installato un modulo diverso da una CPU (ad es. modulo di I/O o controllore di motion).	1.) Non sono consentiti slot liberi fra moduli CPU. Sulla destra dei moduli CPU si possono avere slot liberi. 2.) Smontare il modulo inserito fra i moduli CPU. Installare il controllore di motion alla destra delle CPU System Q (PLC).	
	Un modulo funzione intelligente incompatibile con i sistemi multi CPU è stato assegnato alla CPU 2, 3 o 4.	1.) Sostituire il modulo con uno compatibile con il sistema multi CPU. 2.) Assegnare il modulo incompatibile con il sistema multi CPU alla CPU 1.	Q CPU (Versione B o successiva)

Tabella dei codici di errore; CPU QnA e System Q

Codici di errore da 2200 a 2413

Codice di errore (SD0) ¹	Messaggio di errore	Informazioni generali (da SD5 a 15) ¹	Informazioni specifiche (da SD13 a 20) ¹	Stato LED		Stato CPU	Temporizzazione diagnostica
				RUN	ERROR		
2200	MISSING PARA.	Nome drive	—	OFF	Lampeggiante	Stop	Accensione/reset
2210	BOOT ERROR	Nome drive	—	OFF	Lampeggiante	Stop	Accensione/reset
2300	ICM. OPE. ERROR	Nome drive	—	OFF/ON	Lampeggiante/ON	Stop/Continua ²	Se viene inserita o rimossa una memory card.
2301							
2302							
2400	FILE SET ERROR	Nome file	Numero parametro	OFF	Lampeggiante	Stop	Accensione/reset
2401							
2410	FILE OPE. ERROR	Nome file	Posizione errore programmazione	OFF/ON	Lampeggiante/ON	Stop/Continua ²	Durante l'esecuzione di un'istruzione
2411							
2412							
2413							

¹ Le informazioni in parentesi () indicano gli indirizzi dei registri speciali in cui sono memorizzate le singole informazioni.

² Lo stato operativo della CPU a seguito di un errore può essere configurato tramite parametri (il display a LED cambia corrispondentemente).

Tabella dei codici di errore; CPU QnA e System Q

	Descrizione errore e causa	Rimedio	Valido per:
	Nessun file parametri nel drive specificato dal DIP switch come drive valido.	Controllare e correggere i parametri di configurazione in base ai drive esistenti. Inserire un file parametri nel drive indicato.	U
	Il contenuto del file di boot non è corretto.	Controllare i parametri di boot.	Q CPU
	Nessun file di boot nel drive designato dai DIP switch, anche se l'interruttore BOOT è su ON.	Controllare e correggere i parametri di configurazione in base ai drive esistenti. Inserire un file parametri nel drive indicato.	CPU QnA
	1.) È stata rimossa una memory card senza aver portato il commutatore in/out della memory card su OFF. 2.) L'interruttore di scheda inserita è su ON anche se nessuna memory card è installata.	Rimuovere la memory card solo dopo aver portato il commutatore in/out della memory card su OFF. Commutare su OFF l'interruttore di scheda inserita, quando non è presente una memory card.	U
	1.) La memory card non è formattata. 2.) Stato di formattazione della memory card non corretto.	1.) Formattare la memory card. 2.) Riformattare memory card.	
	Inserita una memory card non adatta per CPU Q/QnA.	Controllare la memory card.	
	È stata eseguita una scrittura automatica sulla ROM standard di una CPU System A incompatibile con questa funzione. (La scrittura automatica dalla memory card alla ROM standard viene selezionata nel file di boot e il parametro di abilitazione drive è stato impostato su memory card).	1.) Eseguire la scrittura automatica sulla ROM standard solo su una CPU che supporta questa funzione. 2.) Scrivere parametri e programmi nella ROM standard usando il software di programmazione. 3.) Disabilitare la scrittura automatica nella ROM standard ed eseguire l'operazione di boot dalla memory card.	Q CPU (Versione B o successiva)
	Il file indicato nei parametri di configurazione PC è inesistente.	Leggere le informazioni di errore specifiche sul display del terminale di programmazione ed accertarsi che drive e destinazione del file corrispondano ai valori numerici visualizzati. Creare il file indicato.	U
	Il parametro Ethernet relativo ad una CPU QnA con versione funzionale "B" è stato impostato per una CPU QnA con versione funzionale inferiore a "B".	Sostituire la CPU con una CPU QnA versione funzionale "B". Cancellare il parametro Ethernet.	CPU QnA
	Durante l'operazione di boot o la scrittura automatica, è stata superata la capacità di memoria della ROM standard.	1.) Controllare e correggere i parametri (configurazione di boot). 2.) Cancellare i file non necessari dalla memoria programma. 3.) Impostare il parametro „Cancella memoria programmi“ in modo da avviare il boot dopo la cancellazione della memoria.	Q CPU (Versione B o successiva)
	Il file specificato come impostazione PLC-RAS non è stato creato nell'area storico errori.	Leggere le informazioni di errore specifiche sul display del terminale di programmazione ed accertarsi che drive e destinazione del file corrispondano ai valori numerici visualizzati. Controllare la memoria restante sulla memory card.	U
	Il file specificato dal programma non esiste.	Leggere le informazioni di errore specifiche sul display del terminale di programmazione ed accertarsi che il programma corrisponda al valore numerico visualizzato. Creare il file indicato.	U
	Il programma di sequenza non può indicare questo tipo di file (file commenti ecc.)	Leggere le informazioni di errore specifiche sul display del terminale di programmazione ed accertarsi che il programma corrisponda al valore numerico visualizzato.	
	Il file di programma SFC non può essere utilizzato dal programma di sequenza.	Leggere le informazioni di errore specifiche sul display del terminale di programmazione ed accertarsi che il programma corrisponda al valore numerico visualizzato.	
	Il file designato dal programma di sequenza non contiene dati.	Leggere le informazioni di errore specifiche sul display del terminale di programmazione ed accertarsi che il programma corrisponda al valore numerico visualizzato. Accertarsi che il file indicato non sia protetto in scrittura.	

Tabella dei codici di errore; CPU QnA e System Q

Codici di errore da 2500 a 3013

Codice di errore (SD0) ¹	Messaggio di errore	Informazioni generali (da SD5 a 15) ¹	Informazioni specifiche (da SD13 a 20) ¹	Stato LED		Stato CPU	Temporizzazione diagnostica
				RUN	ERROR		
2500	CAN'T EXE. PRG.	Nome file	—	OFF	Lampeggiante	Stop	Accensione/reset
2501							
2502							
2503							
2504							
3000	PARAMETER ERROR	Nome file	Numero parametro	OFF	Lampeggiante	Stop	Accensione/reset/ STOP ® RUN
3001							
3002							
3003		Nome file/nome drive					Durante l'esecuzione dell'istruzione END.
		Nome file					Accensione/reset/ STOP ® RUN
3004							
3009	PARAMETER ERROR	Nome file/nome drive	Numero parametro	OFF	Lampeggiante	Stop	Accensione/reset/ STOP ® RUN
3010							
3012							
3013							

¹ Le informazioni in parentesi () indicano gli indirizzi dei registri speciali in cui sono memorizzate le singole informazioni.

Tabella dei codici di errore; CPU QnA e System Q

Descrizione errore e causa	Rimedio	Valido per:
Un file di programma usa un operando che supera il campo di allocazione previsto con i parametri di configurazione operandi.	Leggere le informazioni di errore specifiche sul display del terminale di programmazione ed accertarsi che drive e destinazione del file di programma corrispondano ai valori numerici visualizzati (nome file).	U
Esistono più file di programma, ma il parametro di impostazione programma specifica "nessuno".	Modificare il parametro relativo al programma in "si". Cancellare i programmi non necessari.	
Il file di programma non è adatto per una CPU QnA. In alternativa, il contenuto del file non corrisponde a un programma a sequenza.	Controllare che il formato del file sia *.QPG e che il file contenga effettivamente un programma a sequenza.	
Non esiste alcun file.	Controllare la configurazione del programma	
Specificati due o più programmi normali SFC o programmi di controllo.	Controllare i parametri e la configurazione di programma.	
In un sistema multi CPU, un modulo funzione intelligente controllato da un'altra CPU è specificato nella configurazione del puntatore di interruzione.	1.) Specificare l'indirizzo iniziale di I/O di un modulo che non sia controllato da un'altra CPU. 2.) Cancellare i parametri di configurazione del puntatore di interruzione.	Q CPU (Vers. B o successiva)
I parametri di configurazione per impostazione limite di tempo dei timer, contatto RUN/PAUSE, indirizzo puntatore comune, numero slot vacanti per elaborazione dati generale o sistema di interrupt, superano il campo ammesso dalla CPU.	1.) Leggere le informazioni di errore dettagliate sul display del terminale di programmazione, controllare i parametri corrispondenti ai valori numerici indicati (numeri dei parametri), modificando quanto necessario. 2.) Se l'errore permane, è probabile un guasto sia sulla RAM interna della CPU o sulla memory card. Contattare il più vicino centro di Assistenza MITSUBISHI.	U Rem
Configurazione parametri corrotta.		U
Il file registri specificato durante la configurazione alla voce "use following file" non esiste, anche se la sua dimensione è stata definita.		
Il campo di rinfresco automatico del sistema multi CPU supera la capacità del file registri.	Usare un'area del file registri per cui sia possibile il rinfresco automatico.	Q CPU (Vers. B o successiva)
Il numero di operando impostati con i dati di configurazione operandi, supera il campo ammesso dalla CPU.	1.) Leggere le informazioni di errore dettagliate sul display del terminale di programmazione, controllare i parametri corrispondenti ai valori numerici indicati (numeri dei parametri), modificando quanto necessario. 2.) Se l'errore permane, è probabile un guasto sia sulla RAM interna della CPU o sulla memory card. Contattare il più vicino centro di Assistenza MITSUBISHI.	U
File parametri non corretto per la CPU QnA. In alternativa, il file non contiene parametri.		Controllare che il formato del file sia *.QPA e che il file contenga effettivamente dei parametri.
In un sistema multi CPU, un modulo è assegnato a più di una CPU.	Un modulo può essere controllato da una sola CPU. Modificare le configurazioni di tutte le CPU del sistema.	Q CPU (Versione B o successiva)
Il numero impostato di moduli CPU non coincide con il numero effettivo di moduli CPU.	Modificare la configurazione in base alla situazione reale del sistema.	
La parametrizzazione presente nei singoli moduli CPU di un sistema multi CPU, è differente dalla parametrizzazione della CPU 1.	Far corrispondere la configurazione dei singoli moduli CPU con quella della CPU 1.	
Configurazione rinfresco automatico di un sistema multi CPU non corretta. 1.) Se vengono specificati operandi a bit come operandi di rinfresco, sono stati definiti indirizzi diversi da 0 o da multipli di 16. 2.) L'operando specificato non è corretto. 3.) Il numero di operandi è un numero dispari.	1.) Specificare 0 o un multiplo di 16 come indirizzo del primo operando a bit da rinfrescare. 2.) Specificare l'operando corretto. 3.) Specificare un numero pari di operandi.	

Tabella dei codici di errore; CPU QnA e System Q

Codici di errore da 3100 a 3105

Codice di errore (SD0) ¹	Messaggio di errore	Informazioni generali (da SD5 a 15) ¹	Informazioni specifiche (da SD13 a 20) ¹	Stato LED		Stato CPU	Temporizzazione diagnostica
				RUN	ERROR		
3100							
3101							
3102	LINK PARA. ERROR	Nome file	Numero parametro	OFF	OFF	Stop	Accensione/reset/ STOP © RUN
3103							
3104							
3105							

¹ Le informazioni in parentesi () indicano gli indirizzi dei registri speciali in cui sono memorizzate le singole informazioni.

Tabella dei codici di errore; CPU QnA e System Q

	Descrizione errore e causa	Rimedio	Valido per:
	Il modulo MELSECNET/H con l'indirizzo iniziale di I/O impostato è controllato da un'altra CPU del sistema multi CPU.	Cancellare i parametri di rete per questo modulo MELSECNET/H e specificare l'indirizzo iniziale di I/O del modulo corretto.	Q CPU (Versione B o successiva)
	I parametri di rete di una normale stazione MELSECNET/H sono stati scritti nella stazione attiva o viceversa.	Resetare la CPU.	
	1.) Il numero di moduli effettivamente installati è diverso da quello specificato dal parametro di impostazione numero di moduli di MELSECNET/H 2.) L'indirizzo iniziale di I/O dei moduli effettivamente installati è diverso da quello specificato dal parametro di impostazione della rete MELSECNET/H 3.) Alcuni dati di configurazione non possono essere gestiti. 4.) Il tipo di stazione di MELSECNET/H è stato modificato con sistema' alimentato. (È necessario un passaggio da RESET a RUN per modificare il tipo di stazione).	1.) Riscrivere i parametri di rete corretti. 2.) Se l'errore permane, contattare il Centro Assistenza MITSUBISHI più vicino	Q CPU
	I parametri di rete non sono stati scritti anche se la CPU QnA è la stazione di controllo o la stazione master.	1.) Riscrivere i parametri di rete corretti. 2.) Se l'errore permane, contattare il Centro Assistenza MITSUBISHI più vicino	CPU QnA
	1.) La configurazione dei parametri di rete per comunicazione fra PLC è stata inserita in un modulo MELSECNET/H con numero di stazione 0. 2.) La configurazione dei parametri di rete per master remoto è stata inserita in un modulo MELSECNET/H con numero di stazione 0.	Modificare il tipo di stazione o il numero di stazione.	Q CPU (Versione B o successiva)
	Il numero di rete specificato da un parametro è diverso da quello della rete attualmente installata. L'indirizzo di I/O specificato da un parametro è diverso da quello delle unità di I/O attualmente installate. La classe di rete specificata da un parametro è diversa da quello della rete attualmente installata. Il parametro di rinfresco di rete di MELSECNET/10(H) è oltre il campo specificato.	Far corrispondere i parametri specificati con quelli effettivi relativi alla rete e unità installate.	U
	Rilevato un errore durante il controllo dei parametri di rete su un modulo di rete.	1.) Riscrivere i parametri di rete corretti. 2.) Se l'errore permane, contattare il Centro Assistenza MITSUBISHI più vicino	U
	Il modulo ETHERNET con l'indirizzo iniziale di I/O impostato è controllato da un'altra CPU del sistema multi CPU.	Cancellare i parametri di rete per questo modulo ETHERNET e specificare l'indirizzo iniziale di I/O del modulo corretto.	Q CPU (Vers. B o successiva)
	Anche se il numero di unità Ethernet configurate con parametri è maggiore di zero, nessun modulo è attualmente installato. L'indirizzo iniziale di I/O nei parametri di configurazione Ethernet è diverso da quello delle unità di I/O effettivamente montate.	1.) Riscrivere i parametri di rete corretti. 2.) Se l'errore permane, contattare il Centro Assistenza MITSUBISHI più vicino	U Rem
	AJ71QE71 non esiste nella posizione corrispondente all'indirizzo di I/O impostato. La specifica dell'indirizzo di I/O è sovrapposta. Il numero di moduli AJ71QE71 parametrizzato non corrisponde a quelli reali. Ethernet (parametro + istruzione dedicata) è superiore a 5.		CPU QnA
	Ethernet e MELSECNET/10 usano lo stesso numero di rete. Numero di rete, numero di stazione e numero di gruppo impostati sono fuori campo ammesso. L'indirizzo di I/O è fuori del campo ammesso per la CPU usata.		U Rem
	Il modulo CC-Link con l'indirizzo iniziale di I/O impostato è controllato da un'altra CPU del sistema multi CPU.	Cancellare i parametri di rete per questo modulo CC-Link e specificare l'indirizzo iniziale di I/O del modulo corretto.	Q CPU (Vers. B o successiva)
	Anche se il numero di unità CC-Link configurate con parametri è maggiore di zero, il numero di unità attualmente montate è zero. L'indirizzo di I/O specificato dai parametri comuni è diverso da quello delle unità di I/O attualmente installate. La classe di stazione per una unità CC-Link impostata nei parametri non corrisponde a quella della stazione effettivamente montata.	1.) Riscrivere i parametri di rete corretti. 2.) Se l'errore permane, contattare il Centro Assistenza MITSUBISHI più vicino	U Rem
	Il contenuto dei parametri specifici per Ethernet non è corretto.	Riscrivere i parametri di rete corretti.	CPU QnA

Tabella dei codici di errore; CPU QnA e System Q

Codici di errore da 3106 a 4004

Codice di errore (SD0) ¹	Messaggio di errore	Informazioni generali (da SD5 a 15) ¹	Informazioni specifiche (da SD13 a 20) ¹	Stato LED		Stato CPU	Temporizzazione diagnostica
				RUN	ERROR		
3106	LINK PARA. ERROR	Nome file/nome drive	Numero parametro	OFF	Lampeggiante	Stop	Durante l'esecuzione dell'istruzione END.
3107		Nome file					Accensione/reset/ STOP ® RUN
3200		Nome file					Numero parametro
3201	SFC PARA. ERROR	Nome file	Numero parametro	OFF	Lampeggiante	Stop	STOP ® RUN
3202							
3203							
3300	SP. PARA. ERROR	Nome file	Numero parametro	OFF	Lampeggiante	Stop	Accensione/reset/ STOP ® RUN
3301		Nome file	Numero parametro	OFF	Lampeggiante	Stop	Durante l'esecuzione dell'istruzione END.
3302		Nome file/nome drive	Numero parametro				Accensione/reset/ STOP ® RUN
3303		Nome file/nome drive	Numero parametro	Accensione/reset/ STOP ® RUN			
3400	REMOTE PASS. ERROR	—	—	OFF	Lampeggiante	Stop	Accensione/reset/ STOP ® RUN
3401							
4000	INSTRCT CODE. ERR.	Posizione errore programmazione	—	OFF	Lampeggiante	Stop	Accensione/reset/ STOP ® RUN
4001							
4002							
4003							
4004							

¹ Le informazioni in parentesi () indicano gli indirizzi dei registri speciali in cui sono memorizzate le singole informazioni.

Tabella dei codici di errore; CPU QnA e System Q

	Descrizione errore e causa	Rimedio	Valido per:
	Il campo di rinfresco CC-Link supera la capacità del file registri.	Usare un'area del file registri per cui sia possibile il rinfresco.	Q CPU (Vers. B o successiva)
	Il parametro di rinfresco della rete CC-Link è fuori campo ammesso.	Controllare i parametri impostati	Q CPU/Rem
	Il contenuto del parametro CC-Link non è corretto.	Controllare i parametri impostati	U
	Il contenuto dei parametri non è corretto.	Riscrivere i parametri di rete corretti.	U
	Il contenuto delle informazioni sugli attributi di un blocco SFC non è corretto.		
	Il numero di relé di passo configurato è inferiore al numero di relé utilizzati dal programma.		
	Il tipo di esecuzione configurato per un programma SFC non corrisponde a esecuzione scansione.		
	L'indirizzo iniziale di I/O del modulo funzione intelligente impostato nei parametri con GX Configurator è diverso dall'indirizzo di I/O effettivo.	Controllare i parametri impostati	Q CPU/Rem
	La configurazione di rinfresco per il modulo funzione intelligente supera la capacità del file registri.	Usare un'area del file registri per cui sia possibile il rinfresco nell'intero campo.	Q CPU (Versione B o successiva)
	La configurazione dei parametri di rinfresco per il modulo funzione intelligente è oltre il campo ammesso.	Controllare i parametri impostati	
	I parametri di rinfresco del modulo funzione intelligente non sono corretti.		
	È stato configurato il rinfresco automatico o una parametrizzazione simile, per un modulo funzione intelligente controllato da un'altra CPU del sistema multi CPU.	Spostare la configurazione su un modulo controllato dalla CPU in cui viene eseguita l'istruzione.	U
	L'indirizzo iniziale di I/O per il modulo target nel file delle password remote è diverso da 0 _H o 0FF0 _H .	Modificare l'indirizzo iniziale del modulo indirizzato in 0 _H o 0FF0 _H .	Q CPU (Versione B o successiva)
	Lo slot specificato come indirizzo iniziale di I/O nel file password remote non è corretto per una delle ragioni seguenti: - Il modulo non è installato. - Il modulo non è compatibile con il System Q. - Il modulo è diverso da QJ71C24(-R2) o da un modulo ETHERNET del System Q. - Installato un modulo con versione funzionale A (QJ71C24(-R2) o un modulo ETHERNET del System Q).	Installare un modulo QJ71C24(-R2), versione funzionale B, oppure un modulo ETHERNET System Q con versione funzionale B nello slot specificato dall'indirizzo iniziale di I/O.	
	Un modulo QJ71C24(-R2) con versione funzionale B, oppure un modulo ETHERNET System Q (versione funzionale B) controllato da un'altra CPU è stato configurato in un sistema multi-CPU.	Specificare il modulo corretto. Cancellare l'impostazione della password remota.	
	Il programma contiene un codice di istruzione non decodificabile. Il programma contiene una istruzione non utilizzabile.	Leggere le informazioni di errore generali sul display del terminale di programmazione e controllare il passo in errore corrispondente al valore indicati (locazione dell'errore di programmazione).	U
	Il programma contiene una istruzione dedicata per programmi SFC, ma non esistono programmi SFC.		U Rem
	Il nome istruzione non è corretto.		
	L'istruzione indica un numero di operandi non corretto.		
	L'istruzione specifica un operando che non si può usare.		U

Tabella dei codici di errore; CPU QnA e System Q

Codici di errore da 4010 a 4213

Codice di errore (SD0) ¹	Messaggio di errore	Informazioni generali (da SD5 a 15) ¹	Informazioni specifiche (da SD13 a 20) ¹	Stato LED		Stato CPU	Temporizzazione diagnostica		
				RUN	ERROR				
4010	MISSING END INS.	Posizione errore programmazione	—	OFF	Lampeggiante	Stop	Accensione/reset/ STOP © RUN		
4020	CAN'T SET (P)	Posizione errore programmazione	—	OFF	Lampeggiante	Stop	Accensione/reset/ STOP © RUN		
4021									
4030	CAN'T SET (I)	Posizione errore programmazione	—	OFF	Lampeggiante	Stop	Accensione/reset/ STOP © RUN		
4100	ERRORE DI ESECUZIONE	Posizione errore programmazione	—	OFF/ON	Lampeggiante/ON	Stop/Continua ²	Durante l'esecuzione di un istruzione		
4101									
4102								Programma	Posizione errore programmazione
4103								Posizione errore programmazione	—
								Posizione errore programmazione	—
4107								Programma	Posizione errore programmazione
								Posizione errore programmazione	—
4108								Posizione errore programmazione	—
4200	FOR NEXT ERROR	Posizione errore programmazione	—	OFF	Lampeggiante	STOPP	Durante l'esecuzione di un istruzione		
4201									
4202									
4203									
4210	CAN'T EXECUTE (P)	Posizione errore programmazione	—	OFF	Lampeggiante	STOPP	Durante l'esecuzione di un istruzione		
4211									
4212									
4213									

¹ Le informazioni in parentesi () indicano gli indirizzi dei registri speciali in cui sono memorizzate le singole informazioni.

² Lo stato operativo della CPU a seguito di un errore può essere configurato tramite parametri (il display a LED cambia corrispondentemente).

Tabella dei codici di errore; CPU QnA e System Q

	Descrizione errore e causa	Rimedio	Valido per:
	Il programma non contiene nessuna istruzione END (FEND).		U
	Il numero totale di puntatori interni a file usato dal programma supera il numero di puntatori configurato.	Leggere le informazioni di errore generali sul display del terminale di programmazione e controllare che i file corrispondano con i valori indicati (locazione dell'errore di programmazione).	U
	Gli indirizzi del puntatore comune usato dai singoli file sono sovrapposti.		
	Gli indirizzi del puntatore di allocazione usato dai singoli file sono sovrapposti.		
	L'istruzione non può processare i dati contenuti.	Leggere le informazioni di errore generali sul display del terminale di programmazione e controllare il passo di programma indicato.	U
	Gli indirizzi degli operandi specificati per i dati elaborati dall'istruzione superano i rispettivi campi ammessi. In alternativa, i dati memorizzati o le costanti per gli operandi utilizzati dall'istruzione superano i campi ammessi.		
	Un operando di comunicazione diretto (J[]G[]) è stato specificato in un sistema multi CPU per un modulo di rete controllato da un'altra CPU.	Cancelare l'operando diretto di comunicazione che ha provocato l'errore. Specificare un modulo controllato dalla CPU in cui viene eseguita l'istruzione.	Q CPU (Versione B o successiva)
	Il numero di rete o numero di stazione indicato da una istruzione dedicata di rete non è corretto. L'operando diretto di comunicazione non è configurato correttamente.	Leggere le informazioni di errore generali sul display del terminale di programmazione e controllare il passo di programma indicato.	U Rem
	La configurazione dell'istruzione PID dedicata non è corretta.		U
	Più di 32 istruzioni dedicate ai sistemi multi CPU sono state eseguite da una CPU.	Usare l'operando a bit che segnala il completamento di una istruzione per impedire che una CPU possa eseguire più di 32 istruzioni dedicate per sistemi multi CPU.	Q CPU (Versione B o successiva)
	Una istruzione CC-Link è stata eseguita più di 64 volte.	Impostare il numero di ripetizioni delle istruzioni CC-Link a 64 o meno.	CPU QnA
	I parametri CC-Link non sono impostati quando l'istruzione CC-Link viene eseguita.	Eseguire l'istruzione CC-Link dopo aver impostato i parametri CC-Link.	
	Nessuna istruzione NEXT è stata eseguita dopo l'esecuzione di una istruzione FOR. In alternativa, esistono meno istruzioni NEXT rispetto alle istruzioni FOR	Leggere le informazioni di errore generali sul display del terminale di programmazione e controllare il passo di programma indicato.	U
	Viene eseguita una istruzione NEXT senza una precedente istruzione FOR. In alternativa, esistono più istruzioni NEXT rispetto alle istruzioni FOR.		
	Programmati più di 16 livelli di annidamento.	Ridurre il numero di livelli di annidamento fino a un massimo di 16.	U
	Viene eseguita una istruzione BREAK senza una precedente istruzione FOR.	Leggere le informazioni di errore generali sul display del terminale di programmazione e controllare il passo di programma indicato.	
	Viene eseguita una istruzione CALL, ma non esiste nessun sottoprogramma in corrispondenza del puntatore specificato.	Leggere le informazioni di errore generali sul display del terminale di programmazione e controllare il passo di programma indicato.	U
	Il sottoprogramma non contiene nessuna istruzione RET.		
	L'istruzione RET è stata programmata prima dell'istruzione FEND.		
	Programmati più di 16 livelli di annidamento.		
		Ridurre il numero di livelli di annidamento fino a un massimo di 16.	

Tabella dei codici di errore; CPU QnA e System Q

Codici di errore da 4220 a 4611

Codice di errore (SD0) ¹	Messaggio di errore	Informazioni generali (da SD5 a 15) ¹	Informazioni specifiche (da SD13 a 20) ¹	Stato LED		Stato CPU	Temporizzazione diagnostica
				RUN	ERROR		
4220	CAN'T EXECUTE (I)	Posizione errore programmazione	—	OFF	Lampeggiante	STOPP	Durante l'esecuzione di un istruzione
4221							
4223							
4230	INST. FORMAT ERR	Posizione errore programmazione	—	OFF	Lampeggiante	STOPP	Durante l'esecuzione di un istruzione
4231							
4235							
4300	EXTEND INST. ERR.	Posizione errore programmazione	—	OFF/ON	Lampeggiante/ON	Stop/Continua ²	Durante l'esecuzione di un istruzione
4301							
4400	SFCP. CODE ERROR	Posizione errore programmazione	—	OFF	Lampeggiante	STOPP	STOP ® RUN
4410	CAN'T SET (BL)	Posizione errore programmazione	—	OFF	Lampeggiante	STOPP	STOP ® RUN
4411							
4420	CAN'T SET (S)	Posizione errore programmazione	—	OFF	Lampeggiante	STOPP	STOP ® RUN
4421							
4422							
4500	SFCP. FORMAT ERR.	Posizione errore programmazione	—	OFF	Lampeggiante	STOPP	STOP ® RUN
4501							
4502							
4503							
4504							
4600	SFCP. OPE. ERROR	Posizione errore programmazione	—	OFF/ON	Lampeggiante/ON	Stop/Continua ²	Durante l'esecuzione di un istruzione
4601							
4602							
4610	SFCP. EXE. ERROR	Posizione errore programmazione	—	ON	ON	Continua	STOP ® RUN
4611							

¹ Le informazioni in parentesi () indicano gli indirizzi dei registri speciali in cui sono memorizzate le singole informazioni.

² Lo stato operativo della CPU a seguito di un errore può essere configurato tramite parametri (il display a LED cambia corrispondentemente).

Tabella dei codici di errore; CPU QnA e System Q

	Descrizione errore e causa	Rimedio	Valido per:
	È stata generata una interruzione, ma non è stato trovato un puntatore di interruzione corrispondente.	Leggere le informazioni di errore generali sul display del terminale di programmazione e controllare il passo di programma indicato.	U
	Il sottoprogramma non contiene nessuna istruzione IRET.		
	L'istruzione IRET è stata programmata prima dell'istruzione FEND.		
	L'istruzione CHKEND è stata eseguita prima dell'istruzione CHKCIR I numeri delle istruzioni CHK e CHKEND sono uguali.	Leggere le informazioni di errore generali sul display del terminale di programmazione e controllare il passo di programma indicato.	U
	Le istruzioni IX e IXEND non sono programmate in coppia. I numeri delle istruzioni IX e IXEND non combaciano.		
	La configurazione delle condizioni di controllo dell'istruzione CHK non sono corrette. In alternativa, una istruzione CHK è stata usata in un programma a bassa velocità.		
	La programmazione di una istruzione di comando per modulo master MELSEC-NET/MINI-S3 è errata.	Leggere le informazioni di errore generali sul display del terminale di programmazione e controllare il passo di programma indicato.	CPU QnA
	La programmazione di una istruzione di comando per AD57/AD58 è errata.		
	Nessuna istruzione SFCP o SFCPEND in un programma SFC.		U
	Il numero di blocco indicato dal programma SFC supera il massimo valore impostabile.	Leggere le informazioni di errore generali sul display del terminale di programmazione e controllare il passo di programma indicato.	U
	Le indicazioni di numero di blocco nel programma SFC sono sovrapposte.		
	Il numero di passo indicato nel programma SFC supera il passo 511.	Ridurre il numero di passi.	U
	Il numero totale di passi in tutti i programmi SFC supera il valore massimo.		
	Le indicazioni di numero di passo nel programma SFC sono sovrapposte.		
	Il numero di istruzioni BLOCK e BEND nel programma SGC non corrisponde.	Leggere le informazioni di errore generali sul display del terminale di programmazione e controllare il passo di programma indicato.	U
	La programmazione delle istruzioni STEP*, TRAN*, TSET, SEND nel programma SFC non è corretta.		
	Nessuna istruzione STEPI* nel blocco di programma SFC.		
	Il passo indicato dall'istruzione TSET nel programma SFC non esiste.	Leggere le informazioni di errore generali sul display del terminale di programmazione e controllare il passo di programma indicato.	U
	Il passo indicato dall'istruzione TAND nel programma SFC non esiste.		
	Il programma SFC contiene dati che non possono essere elaborati.	Leggere le informazioni di errore generali sul display del terminale di programmazione e controllare il passo di programma indicato. Il programma parte automaticamente dopo l'inizializzazione.	U
	Il programma SFC supera il campo operandi ammesso.		
	L'istruzione START nel programma SFC è preceduta da una istruzione END.		
	Le informazioni sul passo attivo durante un riavviamento del programma SFC non sono corrette.	Leggere le informazioni di errore generali sul display del terminale di programmazione e controllare il passo di programma indicato.	U
	Il commutatore RESET/L.CLR è stato commutato da RESET a RUN durante un riavviamento del programma SFC.		

Tabella dei codici di errore; CPU QnA e System Q

Codici di errore da 4620 a 6222

Codice di errore (SD0) ¹	Messaggio di errore	Informazioni generali (da SD5 a 15) ¹	Informazioni specifiche (da SD13 a 20) ¹	Stato LED		Stato CPU	Temporizzazione diagnostica
				RUN	ERROR		
4620	BLOCK EXE. ERROR	Posizione errore programmazione	—	OFF	Lampeggiante	STOPP	Durante l'esecuzione di un istruzione
4621							
4630	STEP EXE. ERROR	Posizione errore programmazione	—	OFF	Lampeggiante	STOPP	Durante l'esecuzione di un istruzione
4631							
4632							
4633							
5000	WDT ERROR	Tempo (valore impostato)	Tempo (valore effettivo)	OFF	Lampeggiante	STOPP	Sempre
5001							
5010	PRG. TIME OVER	Tempo (valore impostato)	Tempo (valore effettivo)	ON	ON	Continua	Sempre
5011							
6000	PRG. VERIFY ERR. ²	Nome file	—	OFF	Lampeggiante	Stop	Sempre
6010	MODF VERIFY ERR. ²	—	—	ON	ON	Continua	Sempre
6100	TRK. MEMORY ERR. ³	—	—	ON	ON	Continua	Accensione/reset/ STOP © RUN
6101							Durante l'esecuzione dell'istruzione END.
6200	CONTROL EXE. ⁴	Causa della commutazione	—	ON	OFF	Continua	Sempre
6210	CONTROL WAIT. ²	Causa della commutazione	—	ON	OFF	Continua	Sempre
6220	CAN'T EXE CHANGE ⁴	Causa della commutazione	—	ON	ON	Continua	Sempre
6221							
6222							

¹ Le informazioni in parentesi () indicano gli indirizzi dei registri speciali in cui sono memorizzate le singole informazioni.

² Questo errore si verifica solo nel sistema in standby di un sistema ridondante.

³ Questo errore si verifica solo nei sistemi ridondanti. Può essere rilevato sia nel sistema attivo che in quello in standby.

⁴ Questo errore si verifica solo nel sistema attivo di un sistema ridondante.

Tabella dei codici di errore; CPU QnA e System Q

	Descrizione errore e causa	Rimedio	Valido per:
	È stato richiamato un blocco di programma SFC che era già attivo.		●
	Tentativo di richiamare un blocco inesistente in un programma SFC.		
	È stato richiamato un blocco di programma SFC che era già attivo.	Leggere le informazioni di errore generali sul display del terminale di programmazione e controllare il passo di programma indicato.	●
	Tentativo di richiamare un blocco inesistente in un programma SFC.		
	Troppi passi attivi contemporaneamente nei blocchi richiamati dal programma SFC.		
	Troppi passi contemporaneamente attivi in tutti i blocchi progettati.		
	Il tempo di scansione del programma di inizializzazione supera il tempo WDT impostato per l'inizializzazione, indicato dai parametri PC RAS.	Leggere le informazioni di errore specifiche sul display del terminale di programmazione e controllare il valore numerico (tempo) riportato, abbreviando se necessario il tempo di scansione.	●
	Il tempo di scansione programma supera il valore WDT specificato dal parametro PC RAS.		
	1.) Il tempo di scansione del programma supera il tempo di scansione costante specificato nei parametri PC RAS. 2.) Il tempo di esecuzione di un programma a bassa velocità impostato nei parametri PC RAS supera la soglia di tolleranza della scansione costante.	1.) Controllare il tempo di scansione costante impostato. 2.) Controllare e modificare il tempo di scansione costante ed il tempo di esecuzione del programma a bassa velocità.	●
	Il tempo di scansione del programma a bassa velocità supera il tempo WDT impostato per l'esecuzione a bassa velocità, impostato nei parametri PC RAS.	Leggere le informazioni di errore specifiche sul display del terminale di programmazione e controllare il valore numerico (tempo) riportato, abbreviando se necessario il tempo di scansione.	
	Il sistema attivo e quello in standby di un sistema ridondante non hanno gli stessi programmi e/o parametri.	Sincronizzare programmi e parametri tra sistema attivo e sistema in standby.	
	Gli stati operativi del sistema attivo e di quello in standby in un sistema ridondante, non sono gli stessi.	Impostare lo stesso stato operativo sul sistema attivo e su quello in standby.	
	Rilevato errore sulla memoria di tracking del modulo CPU, durante l'inizializzazione del PLC.	Il modulo CPU presenta un guasto hardware. Contattare il più vicino centro di Assistenza MITSUBISHI. Per sostituire il modulo, sostituire per prima la CPU del sistema in standby, poi quella del sistema attivo.	
	Il modulo CPU rileva un errore durante lo handshake per il tracking.	Controllare lo stato dell'altra stazione.	Q4AR CPU
	Il sistema in standby di un sistema ridondante viene commutato come sistema attivo.	Controllare lo stato del sistema attivo.	
	Il sistema attivo di un sistema ridondante viene commutato come sistema in standby.		
	Il sistema in standby di un sistema ridondante non può essere commutato da sistema attivo a sistema in standby a causa di un errore di stato o per altra ragione.	Controllare lo stato del sistema in standby.	
	Commutazione disabilitata a causa di un errore sul modulo di commutazione di bus.	Il modulo di commutazione presenta un guasto hardware. Contattare il più vicino centro di Assistenza MITSUBISHI.	
	Commutazione disabilitata perché una stazione master multiplexata di una rete di I/O remoti è stata inserita nel sistema in standby durante l'inizializzazione.	Controllare la configurazione della rete di I/O remoti.	

Tabella dei codici di errore; CPU QnA e System Q

Codici di errore da 7000 a 10000

Codice di errore (SD0) ¹	Messaggio di errore	Informazioni generali (da SD5 a 15) ¹	Informazioni specifiche (da SD13 a 20) ¹	Stato LED		Stato CPU	Temporizzazione diagnostica
				RUN	ERROR		
7000	MULT CPU DOWN	Num. unità/modulo	—	OFF	Lampeggiante	Stop	Sempre
7002							Accensione/reset
7003							Accensione/reset
7010	MULTI EXE. ERROR	Num. unità/modulo	—	OFF	Lampeggiante	Stop	Accensione/reset
7020	MULTI CPU ERROR		—	ON	ON	Continua	Sempre
9000	F**** ²	Posizione errore programmazione	Numero spia di errore	ON	OFF	Continua	Durante l'esecuzione di un'istruzione
				USER LED ON			
9010	<CHK> ERR ***_*** ³	Posizione errore programmazione	Num. guasto	ON	OFF	Continua	Durante l'esecuzione di un'istruzione
				USER LED ON			
9020	BOOT OK	—	—	OFF	Lampeggiante	Stop	Accensione/reset
10000	CONT.UNIT ERROR	—	—	—	—	—	—

¹ Le informazioni in parentesi () indicano gli indirizzi dei registri speciali in cui sono memorizzate le singole informazioni.

² **** indica il numero della spia di errore rilevata.

³ *** indica il numero contatto e bobina rilevati.

Tabella dei codici di errore; CPU QnA e System Q

Descrizione errore e causa	Rimedio	Valido per:
<p>1.) Rilevato un errore in un modulo CPU in cui è stato impostato l'arresto di tutte le altre CPU di un sistema multi CPU in caso di errore.</p> <p>2.) Una CPU con versione funzionale A viene usata in un sistema multi CPU.</p>	<p>1.) Proseguire la diagnosi del modulo CPU che ha provocato l'arresto del sistema multi CPU.</p> <p>2.) In un sistema multi CPU si possono usare solo moduli CPU con versione funzionale B.</p>	Q CPU (Versione B o successiva)
La CPU 1 di un sistema multi CPU si è arrestata per errore durante l'accensione. Le altre CPU non possono quindi avviarsi. (L'errore si presenta sulle CPU 2, 3 e 4).	Proseguire la diagnosi della CPU.	
<p>1.) Mancata risposta dalla CPU target di un sistema multi CPU durante la fase di comunicazione iniziale.</p> <p>2.) Una CPU con versione funzionale A viene usata in un sistema multi CPU.</p>	<p>1.) Resettare la CPU. Se si presenta nuovamente lo stesso errore, è possibile un guasto su una CPU. Contattare il più vicino centro di Assistenza MITSUBISHI.</p> <p>2.) In un sistema multi CPU si possono usare solo moduli CPU con versione funzionale B.</p>	
Mancata risposta dalla CPU target di un sistema multi CPU durante la fase di comunicazione iniziale.	Resettare la CPU. Se si presenta nuovamente lo stesso errore, è possibile un guasto su una CPU. Contattare il più vicino centro di Assistenza MITSUBISHI.	
<p>1.) Una delle PU di un sistema multi CPU presenta un guasto hardware.</p> <p>2.) Una CPU con versione funzionale A viene usata in un sistema multi CPU. (Questo errore viene rilevato sulle CPU con versione funzionale B).</p> <p>3.) La CPU 2, 3 o 4 è stata resettata durante la fase di accensione. (Questo errore si presenta solo sulla CPU che è stata resettata).</p>	<p>1.) Leggere le informazioni di errore specifiche e sostituire la CPU guasta.</p> <p>2.) In un sistema multi CPU si possono usare solo moduli CPU con versione funzionale B. Sostituire la CPU versione A con una di versione funzionale B.</p> <p>3.) Non è possibile il reset singolo delle CPU 2, 3 o 4. Resettare la CPU 1 per reinizializzare l'intero sistema multi CPU.</p>	●
Rilevato un errore in un modulo CPU in cui è stato impostato l'arresto di tutte le altre CPU di un sistema multi CPU in caso di errore. (Questo errore viene rilevato su tutte le altre CPU, ma non nel modulo CPU in cui si è verificato il guasto).	Proseguire con la diagnosi del modulo CPU in cui si è verificato il guasto.	
Una spia di errore F è stata attivata.	Leggere le informazioni di errore specifiche sul display del terminale di programmazione e controllare il programma corrispondente al valore numerico riportato (numero spia di errore, numero di errore).	●
Rilevato un errore durante l'istruzione CHK.		
Salvataggio automatico dei dati nella ROM standard terminato con successo. (Il LED BOOT lampeggia).	Impostare il drive abilitato su ROM standard, alimentare di nuovo ed eseguire l'operazione di boot dalla ROM standard.	Q CPU (Versione B o successiva)
Si è verificato un errore in una CPU diversa da quella del PLC (ad es. controllore di motion) in un sistema multi CPU.	Proseguire con la diagnosi del modulo CPU in cui si è verificato il guasto.	

13.3 Tabella di codici di errore; serie A (tranne AnA e AnAS)

La tabella seguente contiene una panoramica di tutti i possibili errori ed i messaggi di errore corrispondenti, le cause possibili e le azioni consigliate. I codici di errore sono scritti nel registro speciale D9008 ed il corrispondente numero di passo in cui si è verificato l'errore è contenuto nei registri speciali D9010 e D9011. Questa tabella contiene solo i codici di errore per le CPU AnN, AnU, AnS, A3M, e A2C.

Messaggio di errore	Codice di errore in D9008	CPU Stato	Descrizione errore e causa	Rimedio
INSTRCT. CODE ERR“ (controllato durante l'esecuzione del programma)	10	Stop	Il programma contiene un codice di istruzione che la CPU non riesce a decodificare. Inserito un chip EPROM contenente un programma errato. Il contenuto della memoria è stato modificato. Programmata una istruzione PR o IRET.	Leggere il passo in errore con un terminale di programmazione e correggere il passo corrispondente di programma. Correggere il programma in EPROM o sostituire il chip EPROM.
PARAMETER ERROR“ (controllato a accensione/reset) Stop → RUN, and PAUSE → RUN)	11	Stop	Impostata una capacità superiore alla capacità di memoria della CPU, con successiva scrittura dei dati nella CPU. Il contenuto dei parametri nella memoria della CPU è stato modificato per disturbi elettrici o caricamento errato della memoria. RAM non caricata (nelle CPU A1 o A1N).	Controllare e correggere i parametri, riscrivendoli nella CPU con un terminale di programmazione. Controllare che il chip di RAM sia correttamente inserito nel suo zoccolo.
MISSING END INS.“ (controllato dopo l'attivazione di M9056 o M9057 oppure durante Stop → RUN, e PAUSE → RUN)	12	Stop	Nessuna istruzione END (FEND) nel programma. Nessuna istruzione END in un sottoprogramma impostato da parametri.	Aggiungere l'istruzione END alla fine del programma.
CAN'T EXECUTE (P)“ (controllato all'esecuzione di una delle seguenti istruzioni: CJ, SCJ, JMP, CALLP, FOR/NEXT e durante STOP @ RUN e PAUSE @ RUN)	13	Stop	Nessuna destinazione o destinazioni multiple specificate da una istruzione CJ, SCJ, CALL, CALLP o JMP. Programmata una istruzione CHG senza sottoprogrammi. Programmata una istruzione RET senza nessuna istruzione CALL all'interno del programma. Destinazione dell'istruzione CJ, SCJ, CALL, CALLP o JMP oltre l'istruzione END. Il numero di istruzioni FOR non corrisponde a quello dell'istruzione NEXT. Una istruzione JMP inserita in un loop FOR/NEXT provoca l'uscita dal loop. Una istruzione JMP provoca l'uscita da un sottoprogramma prima dell'istruzione END. Una destinazione di salto cade all'interno di un loop FOR/NEXT o all'interno di un sottoprogramma. Istruzione STOP inserita in un programma di interruzione, in un sottoprogramma o in un loop FOR/NEXT.	Leggere il passo in errore con un terminale di programmazione e correggere il passo corrispondente di programma.

Tabella di codici di errore; serie A (tranne AnA e AnAS)

Codici di errore da 14 a 21

Messaggio di errore	Codice di errore in D9008	CPU Stato	Descrizione errore e causa	Rimedio
<p style="text-align: center;">CHK FORMAT ERR^o</p> <p>(controllato a accensione/reset) Stop → RUN, and PAUSE → RUN)</p>	14	Stop	<p>Istruzioni (compreso NOP) diverse da LDX, LDIX, ANDX, e ANIX sono contenute nel blocco dell'istruzione CHK.</p> <p>Inserite più istruzioni CHK.</p> <p>Il numero di contatti del blocco dell'istruzione CHK supera 150.</p> <p>Il numero di istruzioni di ingresso X nel blocco dell'istruzione CHK supera il valore massimo.</p> <p>Nessuna istruzione CJP con relativa condizione di ingresso precede il blocco dell'istruzione CHK.</p> <p>L'indirizzo dell'operando D1 di una istruzione CHK D1 D2 non corrisponde all'operando prima dell'istruzione CJP.</p> <p>Il puntatore P254 non è inserito all'inizio del blocco dell'istruzione CHK.</p>	<p>Controllare il programma all'interno del blocco dell'istruzione CHK. Correggere il problema con un terminale di programmazione (ad es. inserendo una istruzione di salto) e ripetere l'operazione.</p>
<p style="text-align: center;">CAN'T EXECUTE (I)</p> <p>(controllato durante una interruzione)</p>	15	Stop	<p>L'indirizzo del puntatore di interruzione (I) usato per interrogare il modulo di interrupt non corrisponde a nessun indirizzo o è definito più volte nel programma.</p> <p>Nessuna istruzione IRET nel programma.</p> <p>L'istruzione IRET è programmata in una sezione di programma non in interruzione.</p>	<p>Controllare la presenza di un programma di interruzione che corrisponde all'unità di interrupt ed eliminare i duplicati dell'indirizzo (I).</p> <p>Controllare la presenza dell'istruzione IRET nel programma, inserendola eventualmente.</p> <p>Controllare la presenza dell'istruzione IRET in una sezione di programma non in interruzione, cancellandola eventualmente.</p>
<p style="text-align: center;">CASSETTE ERROR</p> <p>(controllato a accensione/reset) Acceso impossibile alla cartuccia di memoria</p>	16	Stop	<p>La cartuccia di memoria non è caricata.</p>	<p>Spegnere, inserire la cartuccia di memoria e riaccendere.</p>
<p style="text-align: center;">ROM-ERROR</p> <p>(controllato a accensione/reset)</p>	17	Stop	<p>I parametri e il programma non sono memorizzati correttamente nella EPROM inserita.</p> <p>EPROM difettosa.</p>	<p>Riscrivere programma e parametri nella EPROM.</p> <p>Sostituire la EPROM difettosa.</p>
<p style="text-align: center;">MEMORY PROTECT ERROR^o</p> <p>(controllato a accensione/reset)</p>	18	Stop	<p>La EPROM era protetta da scrittura quando la CPU ha cercato di accedere al programma contenuto nella EPROM (DIP switch in posizione ON).</p>	<p>Impostare l'interruttore di protezione scrittura su OFF.</p>
<p style="text-align: center;">RAM ERROR</p> <p>(controllato a accensione/reset e dopo l'attivazione di M9084 in STOP)</p>	20	Stop	<p>La CPU ha controllato la possibilità di effettuare operazioni di lettura e scrittura nella memoria dati della CPU ed ha rilevato un errore su una o entrambe le operazioni.</p>	<p>Dato che si tratta di errore hardware, consultare una rappresentanza Mitsubishi.</p>
<p style="text-align: center;">OPE. CIRCUIT ERR</p> <p>(controllato a accensione/reset)</p>	21	Stop	<p>La circuiteria che esegue l'elaborazione della sequenza nella CPU non funziona correttamente.</p>	<p>Dato che si tratta di errore hardware, consultare una rappresentanza Mitsubishi.</p>

Tabella di codici di errore; serie A (tranne AnA e AnAS)

Codici di errore da 22 a 41

Messaggio di errore	Codice di errore in D9008	CPU Stato	Descrizione errore e causa	Rimedio
WDT ERROR (1) [“] (controllato dopo l'esecuzione di una istruzione END)	22	Stop	Il tempo di scansione supera il tempo di watchdog impostato. Il tempo di scansione del programma utente è stato superato. Il tempo di scansione è aumentato a causa di una momentanea interruzione di alimentazione durante la scansione.	Calcolare e controllare il tempo di scansione del programma utente, riducendolo eventualmente. Controllare il contenuto del registro speciale D9005 usato dalle apparecchiature periferiche. Se il contenuto è diverso da 0, la tensione di rete è insufficiente. Controllare l'alimentazione e ridurre le ondulazioni della tensione.
SUB-CPU ERROR [“]	23	Stop	La Sub-CPU è bloccata o difettosa.	Dato che si tratta di errore hardware, consultare una rappresentanza Mitsubishi.
END NOT EXECUTE [“] (controllato durante l'esecuzione di una istruzione END)	24	Stop	L'istruzione END è stata modificata in un altro codice di istruzione per qualche ragione.	Resettare e riavviare il sistema. Se compare nuovamente lo stesso errore, si tratta di un errore hardware della CPU. Consultare il centro Assistenza, rappresentante o filiale più vicino.
WDT ERROR (2) [“] (controllato continuamente)	25	Stop	La CPU sta eseguendo un loop infinito. La CPU principale non funziona correttamente.	Dato che il programma esegue un loop infinito per una istruzione JMP o CJ, controllare il programma. Dato che si tratta di errore hardware, consultare una rappresentanza Mitsubishi.
MAIN CPU DOWN (1) [“] (controllato continuamente)	26	Stop	La CPU principale è difettosa.	Dato che si tratta di errore hardware, consultare una rappresentanza Mitsubishi.
UNIT VERIFY ERR. [“] (controllato continuamente)	31	RUN (Stop)	I dati del modulo di I/O sono diversi da quelli rilevati all'accensione. Il modulo di I/O (compresi i moduli funzione speciali) non è caricato correttamente o è stato rimosso, oppure è stata caricata una unità diversa.	Nei registri speciali da D9116 a D9123, il bit corrispondente al modulo in errore è „1“. Controllare i registri e controllare il modulo contrassegnato con „1“ sostituendolo. Quando la configurazione effettiva è corretta, eseguire un reset tramite il commutatore.
FUSE BREAK OFF [“] (controllato continuamente)	32	RUN (Stop)	Fusibile interrotto in un modulo di uscita.	Controllare il LED di fusibile interrotto sui moduli di uscita e sostituire il fusibile del modulo con il LED acceso. Nei registri speciali da D9100 a D9107, il bit corrispondente al modulo con il fusibile interrotto è „1“. Sostituire il fusibile del modulo corrispondente.
CONTROL BUS ERR. [“] (controllato durante l'esecuzione di una istruzione FROM/TO)	40	Stop	Le istruzioni FROM e TO non possono essere eseguite. Errore del bus di controllo con modulo funzione speciale.	Questo errore può essere provocato dall'hardware di un modulo funzione speciale, di un modulo CPU o di un telaio base; consultare una rappresentanza Mitsubishi.
SP.UNIT DOWN [“] (controllato durante l'esecuzione di una istruzione FROM/TO)	41	Stop	Durante l'esecuzione di una istruzione FROM o TO, è stato interrogato un modulo funzione speciale senza ottenere nessuna risposta. Il modulo funzione speciale indirizzato è difettoso.	Errore di accesso ad un modulo funzione speciale, consultare una rappresentanza Mitsubishi.

Tabella di codici di errore; serie A (tranne AnA e AnAS)

Codici di errore da 42 a 70

Messaggio di errore	Codice di errore in D9008	CPU Stato	Descrizione errore e causa	Rimedio
LINK UNIT ERROR ⁴	42	Stop	Modulo AJ71(A)R22 o AJ71(A)P22 caricato nella stazione master.	Rimuovere il modulo AJ71(A)R22 o AJ71(A)P22 dalla stazione master.
I/O INT. ERROR ⁴	43	Stop	Ricevuto un segnale di interruzione senza nessun modulo di interrupt caricato.	Errore hardware specifico del modulo, consultare una rappresentanza Mitsubishi.
SP.UNIT LAY ERROR ⁴	44	Stop	Caricati tre o più moduli di comunicazione su un modulo CPU. Caricati due o più moduli AJ71(A)P22, AJ71(A)R22, AJ71AP21 o AJ71AR21. Caricati due o più moduli di interrupt. Modulo funzione speciale assegnato invece di un modulo di I/O o viceversa, durante l'assegnazione dei parametri dei dispositivi periferici.	Ridurre i moduli di comunicazione a due o meno. Ridurre il numero di AJ71(A)P22, AJ71(A)R22, AJ71AP21 o AJ71AR21 a uno o meno. Inserire un solo modulo di interrupt. Correggere la configurazione dei parametri di I/O in base ai dispositivi periferici ed ai moduli funzione speciali effettivamente utilizzati.
SP. UNIT ERROR ⁴ (controllato durante l'esecuzione di una istruzione FROM/TO)	46	Stop	Accesso (esecuzione di una istruzione FROM o TO) ad una locazione che non contiene nessun modulo funzione speciale.	Leggere il passo in errore con un terminale di programmazione, controllare e correggere l'istruzione FROM o TO contenuta nel passo.
LINK PARA. ERROR ⁴	47	Continua	Se viene usata una CPU di comunicazione per configurare una stazione master (numero di stazione „00“): I parametri di collegamento scritti dalla CPU di comunicazione non corrispondono con i parametri di collegamento letti dalla stazione master. In alternativa, i parametri di collegamento non vengono scritti. Il numero totale di stazioni slave impostato è 0.	Scrivere di nuovo i parametri e controllare. Controllare l'impostazione dei numeri di stazione. Se l'errore compare nuovamente, si tratta di un guasto hardware. In questo caso consultare la rappresentanza Mitsubishi.
OPERATION ERROR ⁴ (controllato durante l'esecuzione di una istruzione)	50	Continua	Il risultato dell'operazione BCD ha superato il campo ammesso (9999 o 99999999). Operazione impossibile perché è stato superato il campo operando. Registri file usati in un programma senza impostazione di capacità. Errore di esecuzione durante una istruzione RTOP, RFRP, LWTP o LRDP.	Leggere il passo in errore con un terminale di programmazione, controllare il passo di programma e correggerlo.
MAIN CPU DOWN (2) ⁴ (errore su interruzione)	60	Stop	Istruzione INT eseguita nell'area programmi microcomputer. Malfunzionamento CPU dovuto a disturbi. Guasto hardware.	Eliminare l'istruzione INT. Eliminare i disturbi. Consultare una rappresentanza Mitsubishi.
BATTERY ERROR ⁴ (controllato continuamente)	70	RUN	Tensione batteria scarsa. Batteria non collegata.	Sostituire la batteria. Collegare la batteria se viene usata la memoria RAM o la funzione di rilevamento mancanza rete.

13.4 Tabella di codici di errore; CPU AnA e AnAS

La tabella seguente contiene una panoramica di tutti i possibili errori ed i messaggi di errore corrispondenti, le cause possibili e le azioni consigliate. I codici di errore vengono scritti nel registro speciale D9008, il codice errore dettagliato nel registro speciale D9091 ed il numero del passo di programma corrispondente in cui si è verificato l'errore viene inserito nei registri speciali D9010 e D9011. Questa tabella contiene solo i messaggi di errore delle CPU AnA e AnAS.

Messaggio di errore	Codice di errore in D9008	Dettagli o errore in D9091	Descrizione errore e causa	Rimedio
INSTRCT CODE ERR" (controllato durante Stop → RUN o esecuzione dell'istruzione)	10	101	Il programma contiene codici che non possono essere decodificati dalla CPU.	Leggere il passo in errore con un terminale di programmazione e correggere il passo di programma in errore. Controllare che la ROM non contenga codici di istruzione che non possono essere decodificati. In questo caso sostituire la ROM con una corretta,
		102	Specificata indicizzazione per una costante a 32 bit.	Leggere il passo in errore con un terminale di programmazione e correggere il passo di programma in errore.
		103	L'operando specificato da una istruzione di applicazione estesa non è corretto.	
		104	Una istruzione di applicazione estesa ha una struttura di programmazione non corretta.	
		105	Una istruzione di applicazione estesa ha un nome di comando non corretto.	
		106	Indicizzazione che usa Z o V è inclusa nel programma tra LEDA/B IX e LEDA/B IXEND.	
		107	Specificata indicizzazione per indirizzi operando e impostazione valori in una istruzione OUT per timer o contatori. Specificata indicizzazione come numero di label del puntatore (P) inserito all'inizio della destinazione di una istruzione CJ, SCJ, CALL, CALLP, JMP, LEDA/B FCALL e LEDA/B BREAK oppure come numero di label del puntatore di interruzione (I) inserito all'inizio di un programma di interruzione.	
		108	Altri errori diversi dai codici 101 e 107 sopra citati.	

Tabella di codici di errore; CPU AnA e AnAS

Codici di errore da 11 a 13

Messaggio di errore	Codice di errore in D9008	Dettagli o errore in D9091	Descrizione errore e causa	Rimedio
PARAMETER ERROR“ (controllato a accensione/reset) Stop → RUN, and PAUSE → RUN)	11	111	Le impostazioni di capacità di programma principale e sottoprogrammi, programma microcomputer, commenti file registri, memoria di stato, campionamento tracce e file registri estesi non sono compatibili con quelle della CPU.	Leggere i parametri nella memoria della CPU, controllarne il contenuto, apportare le correzioni necessarie e riscriverli nella memoria.
		112	La capacità totale impostata per programma principale e sottoprogrammi, commenti file registri, memoria di stato, campionamento tracce e file registri estesi supera la capacità della cartuccia di memoria.	
		113	Il campo retentivo impostato dai parametri o l'impostazione di M, L o S non corretti.	
		114	Errore somma di controllo	
		115	Parametri di configurazione per contatto remoto RUN/PAUSE, modo operativo in caso di errore, modo segnalazione spie di errore o modo di segnalazione STOP → RUN non corretti.	
		116	I parametri di configurazione del rinfresco automatico MNET-MINI non sono corretti.	
		117	Parametri di configurazione timer non corretti.	
	118	Parametri di configurazione contatti non corretti.		
MISSING END INS.“ (controllato durante Stop → RUN)	12	121	Istruzione END (FEND) mancante nel programma principale.	Inserire l'istruzione END alla fine del programma principale.
		122	Istruzione END (FEND) mancante nel sottoprogramma, se il sottoprogramma è impostato da parametri.	Inserire l'istruzione END alla fine del sottoprogramma.
CAN'T EXECUTE (P)“ (controllato durante l'esecuzione dell'istruzione)	13	131	Lo stesso numero di operando è usato in due o più passi per i puntatori (P) e i puntatori di interruzione (I) usati come label all'inizio di una destinazione di salto.	Eliminare gli indirizzi puntatori duplicati dalle destinazioni di salto.
		132	La label del puntatore (P) specificata con le istruzioni CJ, SCJ, CALL, CALLP, JMP, LEDA/B FCALL o LEDA/B BREAK non compare prima dell'istruzione END.	Leggere il passo in errore con un terminale di programmazione, controllarne il contenuto e inserire un puntatore di destinazione di salto (P).

Tabella di codici di errore; CPU AnA e AnAS

Codici di errore da 13 a 14

Messaggio di errore	Codice di errore in D9008	Dettagli o errore in D9091	Descrizione errore e causa	Rimedio
CAN'T EXECUTE (P) (controllato durante l'esecuzione dell'istruzione)	13	133	<p>1. Una istruzione RET è stata incontrata nel programma ed eseguita, senza la corrispondente istruzione CALL.</p> <p>1. Le istruzioni NEXT e LEDA/B BREAK sono state incontrate nel programma ed eseguite, senza la corrispondente istruzione FOR.</p> <p>1. Il livello di annidamento per le istruzioni CALL, CALLP e FOR supera i cinque livelli ammessi.</p> <p>1. Nessuna istruzione RET o NEXT corrispondente alle istruzioni CALL o FOR.</p>	<p>1. Leggere il passo in errore con un terminale di programmazione e correggere il passo di programma in errore.</p> <p>1. Ridurre il numero di livelli annidati per le istruzioni CALL, CALLP e FOR a 5 o meno.</p>
		134	L'istruzione CHG è stata incontrata nel programma ed eseguita, ma nessun sottoprogramma è presente.	Leggere il passo in errore usando un terminale di programmazione e cancellare il blocco dell'istruzione CHG.
		135	<p>1. Le istruzioni LEDA/B IX e LEDA IXEND non sono accoppiate.</p> <p>1. Presenti più di 33 coppie di istruzioni LEDA/B IX e LEDA IXEND.</p>	<p>1. Leggere il passo in errore con un terminale di programmazione e correggere il passo di programma in errore.</p> <p>1. Ridurre il numero di coppie di istruzioni LEDA/B IX e LEDA IXEND a 32 o meno.</p>
CHK FORMAT ERR ^o (controllato durante Stop → RUN, e PAUSE → RUN)	14	141	Istruzioni (compreso NOP) diverse da LD, LDIX, ANDX, e ANIX sono contenute nel blocco dell'istruzione CHK.	Controllare la programmazione dell'istruzione CHK e correggerla in base al contenuto del codice di errore dettagliato.
		142	Inserite più istruzioni CHK.	
		143	Il numero di contatti del blocco dell'istruzione CHK supera 150.	
		144	Le istruzioni LEDA/CHK non sono appaiate con le istruzioni LEDA/CHKEND, oppure esistono 2 o più coppie di queste istruzioni.	
		145	Nessuna istruzione CJ con condizione di ingresso programmata prima del blocco dell'istruzione CHK.	
		146	L'indirizzo dell'operando D1 di una istruzione CHK D1 D2 non corrisponde all'operando prima dell'istruzione CJ.	
		147	Indicizzazione usata nel circuito di controllo.	
148	<p>1. Presenti più circuiti di controllo con istruzioni LEDA/CHK - LEDA/CHKEND.</p> <p>1. Presenti 7 o più condizioni di controllo nelle istruzioni LEDA/CHK - LEDA/CHKEND.</p> <p>1. I circuiti di controllo delle istruzioni LEDA/CHK - LEDA/CHKEND sono programmati senza utilizzare contatti X e Y o istruzioni di confronto.</p> <p>1. O circuiti di controllo delle istruzioni LEDA/CHK - LEDA/CHKEND occupano 257 passi o più.</p>			

Tabella di codici di errore; CPU AnA e AnAS

Codici di errore da 15 a 24

Messaggio di errore	Codice di errore in D9008	Dettagli o errore in D9091	Descrizione errore e causa	Rimedio
CAN'T EXECUTE (1) (controllato durante una interruzione)	15	151	L'istruzione IRET è stata incontrata ed eseguita fuori dal programma di interruzione.	Leggere il passo in errore usando un terminale di programmazione e cancellare il blocco dell'istruzione IRET.
		152	Nessuna istruzione IRET nel programma di interruzione.	Controllare la presenza dell'istruzione IRET nel programma di interruzione. Inserire l'istruzione IRET se assente.
		153	Sebbene sia usato un modulo di interrupt, nessun puntatore di interruzione (I) corrispondente al modulo è inserito nel programma. Al verificarsi dell'errore, l'indirizzo del puntatore errato (I) viene memorizzato in D9011.	Controllare il registro speciale D9011 con un terminale di programmazione e controllare se il programma di interruzione che corrisponde ai dati memorizzati è presente, oppure se esistono due o più puntatori di interruzione (I) con lo stesso indirizzo. Apportare le correzioni necessarie.
CASSETTE ERROR*	16	—	La cartuccia di memoria non è caricata.	Togliere l'alimentazione e caricare la cartuccia di memoria.
RAM ERROR (controllato a accensione)	20	201	La RAM che memorizza il programma di sequenza nella CPU ha provocato un errore.	Dato che si tratta di errore hardware della CPU, consultare una rappresentanza Mitsubishi.
		202	La RAM che memorizza l'area di lavoro nel modulo CPU ha provocato un errore.	
		203	La RAM che memorizza l'area operandi nel modulo CPU ha provocato un errore.	
		204	La RAM che memorizza l'area di indirizzamento nel modulo CPU ha provocato un errore.	
OPE. CIRCUIT ERR* (controllato a accensione/reset)	21	211	La circuiteria di indicizzazione della CPU non funziona correttamente.	Dato che si tratta di errore hardware della CPU, consultare una rappresentanza Mitsubishi.
		212	La logica (hardware) della CPU non funziona correttamente.	
		213	La circuiteria per l'elaborazione sequenziale della CPU non funziona correttamente.	
WDT ERROR* (controllato durante l'esecuzione di una istruzione END)	22	-	<p>Tempo di scansione superiore al tempo WDT.</p> <p>2. Il tempo di scansione del programma utente si è allungato per determinate condizioni.</p> <p>3. Il tempo di scansione è stato esteso per una mancanza rete momentanea durante la scansione.</p>	<p>4. Controllare il tempo di scansione del programma utente ed abbreviarlo usando le istruzioni CJ.</p> <p>5. Sorvegliare il contenuto del registro speciale D9005 con un terminale di programmazione. Se il contenuto è diverso da 0, la tensione di alimentazione non è stabile. Controllare l'alimentatore e ridurre le variazioni di tensione.</p>
END NOT EXECUTE* (controllato durante l'esecuzione di una istruzione END)	24	241	<p>L'intero programma è stato eseguito senza incontrare l'istruzione END.</p> <p>6. Durante l'esecuzione dell'istruzione END, il codice operativo è stato letto erroneamente a causa di disturbi.</p> <p>7. Il codice dell'istruzione END è stato modificato in un codice diverso per una ragione sconosciuta.</p>	1. Resettare la CPU e riportarla in RUN. Se si verifica lo stesso errore si tratta di errore hardware della CPU, consultare una rappresentanza Mitsubishi.

Tabella di codici di errore; CPU AnA e AnAS

Codici di errore da 26 a 43

Messaggio di errore	Codice di errore in D9008	Dettagli o errore in D9091	Descrizione errore e causa	Rimedio
MAIN CPU DOWN (1) (controllato continuamente)	26	—	Malfunzionamento o guasto della CPU principale.	Dato che si tratta di errore hardware della CPU, consultare una rappresentanza Mitsubishi.
UNIT VERIFY ERR. (controllato continuamente)	31	—	Le informazioni attuali del modulo di I/O sono diverse da quelle rilevate durante l'accensione. 2. Il collegamento del modulo di I/O (o modulo funzione speciale) è intermittente o il modulo è stato scollegato durante il funzionamento, oppure è stato collegato un modulo errato.	Leggere le informazioni di errore dettagliate con un terminale di programmazione e controllare o sostituire il modulo che corrisponde al dato indicato (indirizzo iniziale di I/O). In alternativa, controllare i registri speciali da D9116 a D9123 con un terminale di programmazione e controllare o sostituire i moduli il cui bit corrispondente è „1“.
„FUSE BREAK OFF“ (controllato continuamente)	32	—	Presente un modulo di uscita con fusibile interrotto.	3. Controllare il LED FUSE BLOWN sul modulo di uscite e sostituire il fusibile. 4. Leggere le informazioni di errore dettagliate con un terminale di programmazione e controllare o sostituire il fusibile del modulo che corrisponde al dato indicato (indirizzo iniziale di I/O). In alternativa, controllare i registri speciali da D9100 a D9107 con un terminale di programmazione e sostituire il fusibile sul modulo di uscita il cui bit corrispondente è „1“.
CONTROL BUS ERR.“	40	401	Le istruzioni FROM/TO non possono essere eseguite a causa di un errore sul bus di controllo che collega i moduli funzione speciali.	Dato che si tratta di errore hardware su un modulo funzione speciale, modulo CPU o telaio di base, sostituire e controllare i moduli difettosi. Consultare la rappresentanza Mitsubishi per i moduli difettosi.
		402	Se viene eseguita l'assegnazione degli I/O, i moduli funzione speciali non sono accessibili per le comunicazioni iniziali. In caso di errore, l'indirizzo iniziale di I/O (le 2 cifre più pesanti di un numero a 3 cifre) del modulo funzione speciale che ha provocato l'errore viene memorizzato in D9011.	
SP.UNIT DOWN“	41	411	Mancata risposta durante l'accesso a un modulo funzione speciale con una istruzione FROM/TO.	Dato che l'errore riguarda l'accesso a un modulo funzione speciale, consultare la rappresentanza Mitsubishi.
		412	Mancata risposta da un modulo funzione speciale durante la parametrizzazione dell'assegnazione degli I/O. In caso di errore, l'indirizzo iniziale di I/O (le 2 cifre più pesanti di un numero a 3 cifre) del modulo funzione speciale che ha provocato l'errore viene memorizzato in D9011.	
LINK UNIT ERROR“	42	—	1. Modulo AJ71(A)R22 o AJ71(A)P22 caricato nella stazione master. 1. Esistono due moduli di comunicazione configurati nella stazione master (stazione 0).	1. Rimuovere il modulo AJ71(A)R22 o AJ71(A)P22 dalla stazione master. 1. Ridurre a 1 il numero di stazioni master. Ridurre a 1 il numero di moduli di comunicazione se non viene utilizzato un sistema 3-tier.
I/O INT. ERROR“	43	—	Ricevuto un segnale di interruzione senza nessun modulo di interrupt caricato.	Errore hardware di un modulo, controllare e sostituire il modulo difettoso. Per i moduli difettosi, consultare una rappresentanza Mitsubishi.

Tabella di codici di errore; CPU AnA e AnAS

Codici di errore da 44 a 70

Messaggio di errore	Codice di errore in D9008	Dettagli o errore in D9091	Descrizione errore e causa	Rimedio
SP.UNIT LAY ERROR“	44	441	Modulo funzione speciale assegnato invece di un modulo di I/O o viceversa, durante l'assegnazione dei parametri dal dispositivo periferico.	Eseguire nuovamente l'assegnazione di I/O usando i parametri dal dispositivo periferico, rispettando lo stato effettivo dei moduli funzione speciali.
		442	Esistono 9 moduli funzione speciali o più (tranne AI61(S1)) che possono inviare segnalazioni di interrupt alla CPU.	Ridurre a 8 o meno il numero dei moduli funzione speciali (tranne AI61(S1)) che possono inviare interrupt.
		443	Caricati 2 o più moduli AJ71AP21, AJ71AR21, AJ71P22 o AJ71R22.	Ridurre a 1 o meno il numero di moduli AJ71AP21, AJ71AR21, AJ71P22 o AJ71R22.
		444	Esistono 7 o più moduli di comunicazione con computer caricati su un modulo CPU.	Ridurre i moduli di comunicazione con computer a 6 o meno.
		445	Esistono 2 o più moduli AI61(S1) caricati.	Inserire un solo modulo AI61.
		446	I moduli definiti dai parametri per il rinfresco automatico MNT/MINI dal terminale di programmazione non coincidono con i tipi di moduli effettivamente collegati.	Eseguire nuovamente l'assegnazione dei moduli per il rinfresco automatico MNT/MINI con parametri che corrispondano ai moduli effettivamente collegati.
		447	Il numero di moduli funzione speciali assegnati ad un modulo CPU, che possono utilizzare il set di istruzioni dedicate supera il limite specificato. (il numero totale non può superare 1344). (AD59 X 5) (AD57(S1)/AD58 X 8) (AJ71C24(S3/S6/S8) X 10) (AJ71UC24 X 10) AJ71C21(S1) X 29 + (AJ71PT32(S3) X 125) ----- Totale > 1344	Ridurre il numero di moduli funzione speciali installati.
SP. UNIT ERROR“ (controllato durante l'esecuzione di una istruzione FROM/TO)	46	461	Il modulo specificato dalle istruzioni FROM/TO non è un modulo funzione speciale.	Leggere il passo in errore con un terminale di programmazione e correggere l'istruzione FROM/TO contenuta nel passo.
		462	Il modulo specificato dall'istruzione dedicata per moduli funzione speciali non è un modulo funzione speciale, oppure non è quello corretto.	Leggere il passo in errore con un terminale di programmazione e controllare e correggere il contenuto dell'istruzione dedicata ai moduli funzione speciali contenuta nel passo.
LINK PARA. ERROR“	47	-	2. Se viene usata una CPU di comunicazione per configurare una stazione master (numero di stazione „0“): I parametri di collegamento scritti dalla CPU di comunicazione non corrispondono con i parametri di collegamento letti dalla stazione master. In alternativa, i parametri di collegamento non vengono scritti. 3. Il numero totale di stazioni locali impostato è 0.	4. Scrivere di nuovo i parametri e controllare. 5. Controllare l'impostazione dei numeri di stazione. 6. Se compare la stessa segnalazione di errore, c'è un guasto hardware. Consultare una rappresentanza Mitsubishi.
BATTERY ERROR“ (controllato continuamente)	70	-	7. La tensione di batteria è scesa sotto a un livello specificato. 8. Il cavo della batteria non è collegato.	9. Sostituire la batteria. 10. Se si utilizza la memoria RAM o la funzione di gestione mancanza rete, collegare il cavo della batteria.

Tabella di codici di errore; CPU AnA e AnAS

Codici di errore da 50 a 60

Messaggio di errore	Codice di errore in D9008	Dettagli o errore in D9091	Descrizione errore e causa	Rimedio
OPERATION ERROR“ (controllato durante l'esecuzione dell'istruzione)	50	501	11.Se viene usato il file registri (R), l'operazione è stata eseguita oltre i campi specificati degli indirizzi operandi e numeri di blocco del file registri (R). 12.File registri utilizzati nel programma senza averne definito la capacità.	Leggere il passo in errore con un terminale di programmazione e correggere il passo di programma in errore.
		502	La combinazione di operandi specificata dall'istruzione non è corretta.	
		503	Costante o dato contenuto nell'operando specificato non è nel campo utile.	
		504	Numero di dati impostato oltre il campo ammesso.	
		505	13.Il numero di stazione specificat dalle istruzioni LEDA/B LRDP, LCDA/B LWTP, LRDP, LWTP n corrisponde a una stazione locale. 14.L'indirizzo iniziale di I/O specificat dalle istruzioni LEDA/B RFRP, LEDA/B RTOP, RFRP, RTOP n corrisponde a una stazione remota.	
		506	L'indirizzo iniziale di I/O specificato dalle istruzioni LEDA/B RFRP, LEDA/B RTOP, RFRP, RTOP non corrisponde a un modulo funzione speciale.	
	50	507	15.Accesso dalla rete a un modulo AD57(S1) o AD58 durante l'esecuzione di istruzioni in "divided processing mode". 16.Durante l'esecuzione di istruzioni in "divided processing mode" di un modulo AD57(S1) o AD58, si accede contemporaneamente ad un altro modulo AD57(S1) o AD58.	Leggere il passo in errore con un terminale di programmazione e interbloccare con il relé speciale M9066, oppure modificare la struttura del programma in modo che mentre lo AD57(S1) o AD68 sta eseguendo istruzioni in "divided processing mode", non si possano eseguire altre istruzioni nello stesso modo nello stesso o in un altro modulo AD57(S1) o AD68.
		509	17.Una istruzione non consentita da moduli terminale remoti è stata richiesta per un modulo terminal collegato a MNET/MINI-S3. 18.Durante l'esecuzione di una istruzione PRC verso un terminale remoto, le aree di registrazione della richiesta di comunicazione risultano sovrapposte. 19.È stata eseguita l'istruzione PDCONT senza aver prima eseguito l'istruzione PIDINIT. È stata eseguita l'istruzione PID57 senza aver prima eseguito l'istruzione PIDINIT o PIDCONT.	20.Leggere il passo in errore con un terminale di programmazione e correggere il programma adeguandosi alle capacità dei moduli terminale remoti. 21.Prevedere un interblocco con il relé M9081 (segnale di BUSY delle aree di definizione della richiesta di comunicazione) o il registro D9081 (numero di aree vacanti nell'area di definizione della richiesta di comunicazione) quando viene eseguita l'istruzione PRC verso un terminale remoto. 22.Eseguire l'istruzione PIDCONT dopo aver eseguito l'istruzione PIDINIT. Eseguire l'istruzione PID57 dopo aver eseguito le istruzioni PIDINIT e PIDCONT.
MAIN CPU DOWN (2)“ (errore durante interruzione)	60	-	23.Malfunzionamento CPU dovuto a disturbi. 24.Guasto hardware.	25.Adottare le misure appropriate contro i disturbi. 26.Guasto hardware.

A Appendice A

A.1 Definizione dei tempi di elaborazione

Il tempo di elaborazione di una operazione è la sommatoria di quanto segue:

- Sommatoria dei tempi di esecuzione di ciascuna istruzione.
- Tempo di elaborazione END. Questo tempo è composto dal tempo di esecuzione dell'istruzione END, dal tempo di rinfresco della rete MELSECNET, dal tempo di elaborazione per la comunicazione con i dispositivi periferici e dal tempo per la comunicazione seriale.
- Il tempo di rinfresco I/O può essere calcolato con la formula seguente:

$$\text{Tempo rinfresco I/O} = \frac{\text{Numero punti di ingresso}}{16} \times N1 + \frac{\text{Numero punti di uscita}}{16} \times N2$$

La tabella seguente indica i due tempi N1 e N2 per le CPU System Q e QnA:

Tipo di CPU	N1 (µs)			N2 (µs)		
	System Q Rack principale	System Q Rack di espansione	QnA Rack di espansione	System Q Rack principale	System Q Rack di espansione	QnA Rack di espansione
Q00JCPU	2,5	3,3	—	1,3	2,3	—
Q00CPU	2,4	3,2	—		2,3	—
Q01CPU	2,3	3,1	—		2,3	—
Q02CPU)	2,2	2,9	4,3		2,1	3,5
Q02HCPU Q06HCPU Q12HCPU Q12PHCPU Q25HCPU Q25PHCPU	1,7	2,4	3,7		2,1	3,5
Q2ASCPU (S1) Q2ACPU	5,2			5,0		
Q3ACPU	4,8			4,65		
Q2ASHCPU (S1) Q4ACPU Q4ARCPU	4,34			4,26		

A.1 Tempi di elaborazione

La tabella riportata alle pagine seguenti contiene i tempi di elaborazione di tutte le istruzioni. I tempi effettivi di elaborazione dipendono dai valori degli operandi sorgente e destinazione. I valori di tempo riportati servono come riferimento per il calcolo del tempo totale di elaborazione di un programma.

Il tempo di elaborazione di una istruzione non comprende il tempo per l'indicizzazione.

Se l'istruzione non viene eseguita, il tempo di elaborazione viene calcolato come segue:

Tipo di CPU	Tempo di elaborazione se l'istruzione non viene eseguita (µs)
Q00JCPU	0,20 x (numero di passi per ciascuna istruzione +1)
Q00CPU	0.16 x (numero di passi per ciascuna istruzione +1)
Q01CPU	0.10 x (numero di passi per ciascuna istruzione +1)
Q02CPU)	0.079 x (numero di passi per ciascuna istruzione +1)
Q02HCPU Q06HCPU Q12HCPU Q12PHCPU Q25HCPU Q25PHCPU	0.034 x (numero di passi per ciascuna istruzione +1)
Q2ASCPU (S1) Q2ACPU	0,20 x (numero di passi per ciascuna istruzione +1)
Q3ACPU	0.15 x (numero di passi per ciascuna istruzione +1)
Q2ASHCPU (S1) Q4ACPU Q4ARCPU	0.075 x (numero di passi per ciascuna istruzione +1)

A.2.1 Tabella dei tempi di elaborazione (serie QnA e System Q)

Istruzione	Elaborazione (operando)		Tempo di elaborazione (µs)									
			Moduli CPU serie QnA				Moduli CPU System Q					
			Q2A Q2AS	Q3A	Q4A QnASH	Q4AR	Q00J	Q00	Q01	Q02	QnH	
LD	ad es. X0	ad es. D0.0	0.20	0.15	0.075	0.075	0.20	0.16	0.10	0.079	0.034	
LDI												
AND												
ANI												
OR												
ORI												
LDP	6.6	5.0	2.5	2.5	0.30	0.24	0.15	0.158	0.068			
LDF												
ANDP												
ANDF												
ORP												
ORF												
ANB	0.20	0.15	0.075	0.075	0.20	0.16	0.10	0.079	0.034			
ORB												
MPS												
MRD												
MPP												
INV										non eseguita	2.4	1.8
	eseguita											
MEP	non eseguita	2.0	1.5	0.75	0.75	0.30	0.24	0.15	0.173	0.073		
MEF	eseguita											
EGP	non modificata	0.6	0.3	0.15	0.15	0.20	0.16	0.10	0.158	0.068		
	modificata (OFF/ON o ON/OFF)											
EGF	non modificata	0.6	0.3	0.15	0.15	17	9.5	9.4	0.158	0.068		
	modificata (OFF/ON o ON/OFF)					18	14	14				
OUT	tranne F, T e C	non modificata	0.40	0.30	0.15	0.15	0.20	0.16	0.10	0.158	0.068	
		modificata (OFF/ON o ON/OFF)										
	D0.0	non modificata	0.40	0.30	0.15	0.15	0.40	0.32	0.20	0.158	0.068	
		modificata (OFF/ON o ON/OFF)										
	F	non eseguita	7.0	5.3	2.7	2.7	24	20	19	2.8	1.2	
		eseguita	visualizzato	167	126	63	63	260	210	200	162	69.7
			visualizzazione terminata	166	125	62	62	205	165	155	126	54
	T	non eseguita	1.6	1.2	0.6	0.6	1.1	0.88	0.55	0.63	0.27	
		dopo timeout										
		eseguita										sommato
	D											
	C	non eseguita	1.6	1.2	0.6	0.6	1.1	0.88	0.55	0.63	0.27	
dopo timeout												
eseguita		sommato										K
	D											
OUTH	non eseguita	1.6	1.2	0.6	0.6	1.1	0.88	0.55	0.63	0.27		
	dopo timeout											
	eseguita										sommato	K
D												
SET	tutti gli operandi tranne Fe D0.0	non eseguita	0.40	0.30	0.15	0.15	0.20	0.16	0.10	0.158	0.068	
		eseguita										non modificata
		modificato										
	D0.0	non eseguita	0.40	0.30	0.15	0.15	0.40	0.32	0.20	0.158	0.068	
		eseguita										non modificata
		modificato										
F	non eseguita	1.2	0.90	0.45	0.45	0.50	0.44	0.25	0.47	0.20		
	eseguita	visualizzato	277	208	104	104	255	205	195	161	69	
		visualizzazione terminata	1.2	0.90	0.45	0.45	195	160	150	0.47	0.20	

Istruzione	Elaborazione (operando)		Tempo di elaborazione (µs)																	
			Moduli CPU serie QnA				Moduli CPU System Q													
			Q2A Q2AS	Q3A	Q4A QnASH	Q4AR	Q00J	Q00	Q01	Q02	QnH									
RST	Tutti gli operandi tranne quelli sotto elencati	non eseguita	0.40	0.30	0.15	0.15	0.20	0.16	0.10	0.158	0.068									
		non modificata																		
		eseguita modificato																		
	D0.0	non eseguita	0.40	0.30	0.15	0.15	0.40	0.32	0.20	0.158	0.068									
		non modificata																		
		eseguita modificato																		
	SM	non eseguita	0.40	0.30	0.15	0.15	0.20	0.16	0.10	0.158	0.068									
		eseguita																		
	F	non eseguita	1.2	0.90	0.45	0.45	0.48	0.44	0.25	0.47	0.20									
		eseguita visualizzato										148	112	56	56	75	69	65	90	38
		eseguita visualizzazione terminata										1.2	0.90	0.45	0.45	43	35	33	0.47	0.20
	T, C	non eseguita	1.4	1.1	0.6	0.6	0.80	0.64	0.40	0.63	0.27									
		eseguita					1.0	0.80	0.50											
	D	non eseguita	0.60	0.45	0.23	0.23	0.40	0.32	0.20	0.24	0.10									
		eseguita					0.60	0.48	0.30											
	Z	non eseguita	1.2	0.90	0.45	0.45	0.50	0.40	0.25	0.47	0.20									
		eseguita										10.8	8.1	4.1	4.1	9.4	7.9	7.4	4.3	1.9
	R	non eseguita	1.0	0.75	0.38	0.38	-	0.32	0.20	0.40	0.17									
eseguita		-					0.48	0.30												
PLS			2.6	2.0	0.98	0.98	12	9.5	9.2	1.0	0.44									
PLF			2.6	2.0	0.98	0.98	11	9.5	8.9	1.0	0.44									
FF	Y	non eseguita	1.2	0.90	0.45	0.45	0.68	0.40	0.25	0.47	0.20									
		eseguita					7.5	6.2	5.7											
DELTA	DY0	non eseguita	1.2	0.90	0.45	0.45	0.50	0.40	0.25	0.47	0.20									
		eseguita										16.8	14.1	11.1	11.1	26	21	21	5.9	2.6
DELTAP	DY0	non eseguita	1.2	0.90	0.45	0.45	0.48	0.40	0.25	0.47	0.20									
		eseguita										16.8	14.1	11.1	11.1	58	45	43	5.9	2.6
SFT SFTP	non eseguita		1.2	0.90	0.45	0.45	0.50	0.34	0.25	0.47	0.20									
	eseguita											4.2	3.2	1.6	1.6	12	8.7	8.3	1.66	0.71
MC	M0.0		0.60	0.45	0.23	0.23	0.40	0.32	0.20	0.24	0.10									
	D0.0						3.3	2.9	2.8											
MCR			0.20	0.15	0.075	0.075	0.20	0.16	0.10	0.079	0.034									
FEND END	eseguito controllo errore		1643	1236	618	618	660	530	480	348	150									
	senza controllo errore: - Controllo batteria - Controllo fusibili interrotti - Verifica modulo di I/O		1106	832	416	416	660	530	480	359	150									
NOP			0.2	0.15	0.075	0.075	0.20	0.16	0.10	0.079	0.034									
NOPLF PAGE			0.2	0.15	0.075	0.075	0.20	0.16	0.10	0.79	0.034									
LD=	continuità		3.8	2.9	1.5	1.5	0.80	0.64	0.40	0.24	0.10									
	non continuità		3.6	2.7	1.4	1.4														
AND=	non eseguita		1.4	1.1	0.55	0.55	0.70	0.56	0.35	0.24	0.10									
	eseguita	continuità	2.8	2.1	1.1	1.1	0.80	0.64	0.40											
		non continuità	3.2	2.4	1.2	1.2														
OR=	non eseguita		1.4	1.1	0.55	0.55	0.70	0.56	0.35	0.24	0.10									
	eseguita	continuità	3.8	2.9	1.5	1.5	0.80	0.64	0.40											
		non continuità	2.8	2.1	1.1	1.1														
LD<>	continuità		4.4	3.3	1.7	1.7	0.80	0.64	0.40	0.24	0.10									
	non continuità		3.6	2.7	1.4	1.4														
AND<>	non eseguita		1.4	1.1	0.55	0.55	0.70	0.56	0.35	0.24	0.10									
	eseguita	continuità	2.8	2.1	1.1	1.1	0.80	0.64	0.40											
		non continuità	3.2	2.4	1.2	1.2														

Istruzione	Elaborazione (operando)		Tempo di elaborazione (µs)								
			Moduli CPU serie QnA				Moduli CPU System Q				
			Q2A Q2AS	Q3A	Q4A QnASH	Q4AR	Q00J	Q00	Q01	Q02	QnH
OR<>	non eseguita		1.4	1.1	0.55	0.55	0.70	0.56	0.35	0.24	0.10
	eseguita	continuità	3.8	2.9	1.5	1.5	0.80	0.64	0.40		
		non continuità	2.8	2.1	1.1	1.1					
LD>	continuità		4.4	3.3	1.7	1.7	0.80	0.64	0.40	0.24	0.10
	non continuità		3.6	2.7	1.4	1.4					
AND>	non eseguita		1.4	1.1	0.55	0.55	0.70	0.56	0.35	0.24	0.10
	eseguita	continuità	2.8	2.1	1.1	1.1	0.80	0.64	0.40		
		non continuità	3.2	2.4	1.2	1.2					
OR>	non eseguita		1.4	1.1	0.55	0.55	0.70	0.56	0.35	0.24	0.10
	eseguita	continuità	3.8	2.9	1.5	1.5	0.80	0.64	0.40		
		non continuità	2.8	2.1	1.1	1.1					
LD<=	continuità		4.4	3.3	1.7	1.7	0.80	0.64	0.40	0.24	0.10
	non continuità		3.6	2.7	1.4	1.4					
AND<=	non eseguita		1.4	1.1	0.55	0.55	0.70	0.56	0.35	0.24	0.10
	eseguita	continuità	2.8	2.1	1.1	1.1	0.80	0.64	0.40		
		non continuità	3.2	2.4	1.2	1.2					
OR<=	non eseguita		1.4	1.1	0.55	0.55	0.70	0.56	0.35	0.24	0.10
	eseguita	continuità	3.8	2.9	1.5	1.5	0.80	0.64	0.40		
		non continuità	2.8	2.1	1.1	1.1					
LD<	continuità		4.4	3.3	1.7	1.7	0.80	0.64	0.40	0.24	0.10
	non continuità		3.6	2.7	1.4	1.4					
AND<	non eseguita		1.4	1.1	0.55	0.55	0.70	0.56	0.35	0.24	0.10
	eseguita	continuità	2.8	2.1	1.1	1.1	0.80	0.64	0.40		
		non continuità	3.2	2.4	1.2	1.2					
OR<	non eseguita		1.4	1.1	0.55	0.55	0.70	0.56	0.35	0.24	0.10
	eseguita	continuità	3.8	2.9	1.5	1.5	0.80	0.64	0.40		
		non continuità	2.8	2.1	1.1	1.1					
LD>=	continuità		4.4	3.3	1.7	1.7	0.80	0.64	0.40	0.24	0.10
	non continuità		3.6	2.7	1.4	1.4					
AND>=	non eseguita		1.4	1.1	0.55	0.55	0.70	0.56	0.35	0.24	0.10
	eseguita	continuità	2.8	2.1	1.1	1.1	0.80	0.64	0.40		
		non continuità	3.2	2.4	1.2	1.2					
OR>=	non eseguita		1.4	1.1	0.55	0.55	0.70	0.56	0.35	0.24	0.10
	eseguita	continuità	3.8	2.9	1.5	1.5	0.80	0.64	0.40		
		non continuità	2.8	2.1	1.1	1.1					
LDD=	continuità		5.0	3.8	1.9	1.9	1.0	0.80	0.50	0.55	0.24
	non continuità		4.2	3.2	1.6	1.6				0.39	0.17
ANDD=	non eseguita		1.4	1.1	0.55	0.55	0.80	0.64	0.40	0.39	0.17
	eseguita	continuità	3.4	2.6	1.3	1.3	1.0	0.80	0.50	0.55	0.24
		non continuità	3.8	2.9	1.5	1.5				0.39	0.17
ORD=	non eseguita		1.4	1.1	0.55	0.55	0.80	0.64	0.40	0.39	0.17
	eseguita	continuità	4.4	3.3	1.7	1.7	1.0	0.80	0.50	0.55	0.24
		non continuità	3.4	2.6	1.3	1.3					
LDD<>	continuità		5.0	3.8	1.9	1.9	1.0	0.80	0.50	0.55	0.24
	non continuità		4.2	3.2	1.6	1.6					
ANDD<>	non eseguita		1.4	1.1	0.55	0.55	0.80	0.64	0.40	0.39	0.17
	eseguita	continuità	3.4	2.6	1.3	1.3	1.0	0.80	0.50	0.55	0.24
		non continuità	3.8	2.9	1.5	1.5					
ORD<>	non eseguita		1.4	1.1	0.55	0.55	0.80	0.64	0.40	0.39	0.17
	eseguita	continuità	4.4	3.3	1.7	1.7	1.0	0.80	0.50	0.55	0.24
		non continuità	3.4	2.6	1.3	1.3					
LDD>	continuità		3.8	2.9	1.5	1.5	1.0	0.80	0.50	0.55	0.24
	non continuità		4.2	3.2	1.6	1.6					
ANDD>	non eseguita		1.4	1.1	0.55	0.55	0.80	0.64	0.40	0.39	0.17
	eseguita	continuità	2.8	2.1	1.1	1.1	1.0	0.80	0.50	0.55	0.24
		non continuità	3.8	2.9	1.5	1.5					
ORD>	non eseguita		1.4	1.1	0.55	0.55	0.80	0.64	0.40	0.39	0.17
	eseguita	continuità	3.8	2.9	1.5	1.5	1.0	0.80	0.50	0.55	0.24
		non continuità	3.4	2.6	1.3	1.3					
LDD<=	continuità		4.4	3.3	1.7	1.7	1.0	0.80	0.50	0.55	0.24
	non continuità		3.6	2.7	1.4	1.4					

Istruzione	Elaborazione (operando)		Tempo di elaborazione (µs)									
			Moduli CPU serie QnA				Moduli CPU System Q					
			Q2A Q2AS	Q3A	Q4A QnASH	Q4AR	Q00J	Q00	Q01	Q02	QnH	
ANDD<=	non eseguita		1.4	1.1	0.55	0.55	0.80	0.64	0.40	0.39	0.17	
	eseguita	continuità	3.4	2.6	1.3	1.3	1.0	0.80	0.50	0.55	0.24	
		non continuità	3.2	2.4	1.2	1.2						
ORD<=	non eseguita		1.4	1.1	0.55	0.55	0.80	0.64	0.40	0.39	0.17	
	eseguita	continuità	4.4	3.3	1.7	1.7	1.0	0.80	0.50	0.55	0.24	
		non continuità	2.8	2.1	1.1	1.1						
LDD<	continuità		3.8	2.9	1.5	1.5	1.0	0.80	0.50	0.55	0.24	
	non continuità		4.2	3.2	1.6	1.6						
ANDD<	non eseguita		1.4	1.1	0.55	0.55	0.80	0.64	0.40	0.39	0.17	
	eseguita	continuità	2.8	2.1	1.1	1.1	1.0	0.80	0.50	0.55	0.24	
		non continuità	3.8	2.9	1.5	1.5						
ORD<	non eseguita		1.4	1.1	0.55	0.55	0.80	0.64	0.40	0.39	0.17	
	eseguita	continuità	3.8	2.9	1.5	1.5	1.0	0.80	0.50	0.55	0.24	
		non continuità	3.4	2.6	1.3	1.3						
LDD>=	continuità		4.4	3.3	1.7	1.7	1.0	0.80	0.50	0.55	0.24	
	non continuità		3.6	2.7	1.4	1.4						
ANDD>=	non eseguita		1.4	1.1	0.55	0.55	0.80	0.64	0.40	0.39	0.17	
	eseguita	continuità	3.4	2.6	1.3	1.3	1.0	0.80	0.50	0.55	0.24	
		non continuità	3.2	2.4	1.2	1.2						
ORD>=	non eseguita		1.4	1.1	0.55	0.55	0.80	0.64	0.40	0.39	0.17	
	eseguita	continuità	4.4	3.3	1.7	1.7	1.0	0.80	0.50	0.55	0.24	
		non continuità	2.8	2.1	1.1	1.1						
LDE=	singola precisione	continuità	235	177	89	35	—	—	—	93	40	
		non continuità	231	174	87	87	—	—	—	92		
	doppia precisione	continuità	—	—	—	—	—	—	—	93	40	
		non continuità	—	—	—	—	—	—	—	92		
ANDE=	singola precisione	non eseguita		1.4	1.1	0.55	0.55	—	—	—	0.55	0.24
		eseguita	continuità	234	176	88	35	—	—	—	93	
	non continuità		230	172	86	86	—	—	—	92		
	doppia precisione	non eseguita		—	—	—	—	—	—	—	—	—
		eseguita	continuità	—	—	—	—	—	—	—	93	40
	non continuità		—	—	—	—	—	—	—	92		
ORE=	singola precisione	non eseguita		1.4	1.1	0.55	0.55	—	—	—	0.55	0.24
		eseguita	continuità	234	176	88	35	—	—	—	93	
	non continuità		230	172	86	86	—	—	—	92		
	doppia precisione	non eseguita		—	—	—	—	—	—	—	—	—
		eseguita	continuità	—	—	—	—	—	—	—	93	40
	non continuità		—	—	—	—	—	—	—	92		
LDE<>	singola precisione	continuità		231	174	87	35	—	—	—	92	40
		non continuità		234	176	88	88	—	—	—		
	doppia precisione	continuità		—	—	—	—	—	—	—	92	40
		non continuità		—	—	—	—	—	—	—		
ANDE<>	singola precisione	non eseguita		1.4	1.1	0.55	0.55	—	—	—	0.55	0.24
		eseguita	continuità	230	172	86	34	—	—	—	92	
	non continuità		234	176	88	88	—	—	—	93		
	doppia precisione	non eseguita		—	—	—	—	—	—	—	—	—
		eseguita	continuità	—	—	—	—	—	—	—	92	40
	non continuità		—	—	—	—	—	—	—	92		
ORE<>	singola precisione	non eseguita		1.4	1.1	0.55	0.55	—	—	—	0.55	0.24
		eseguita	continuità	231	174	87	35	—	—	—	93	
	non continuità		234	176	88	88	—	—	—	92		
	doppia precisione	non eseguita		—	—	—	—	—	—	—	—	—
		eseguita	continuità	—	—	—	—	—	—	—	93	40
	non continuità		—	—	—	—	—	—	—	92		
LDE>	singola precisione	continuità		231	174	87	35	—	—	—	92	40
		non continuità		234	176	88	88	—	—	—		
	doppia precisione	continuità		—	—	—	—	—	—	—	92	40
		non continuità		—	—	—	—	—	—	—		

Istruzione	Elaborazione (operando)		Tempo di elaborazione (µs)									
			Moduli CPU serie QnA				Moduli CPU System Q					
			Q2A Q2AS	Q3A	Q4A QnASH	Q4AR	Q00J	Q00	Q01	Q02	QnH	
ANDE>	singola precisione	non eseguita	1.4	1.1	0.55	0.55	—	—	—	0.55	0.24	
		eseguita	continuità	230	172	86	34	—	—	—	92	40
			non continuità	234	176	88	88	—	—	—	93	
	doppia precisione	non eseguita	—	—	—	—	—	—	—	—	—	
		eseguita	continuità	—	—	—	—	—	—	—	92	40
			non continuità	—	—	—	—	—	—	—	93	
ORE>	singola precisione	non eseguita	1.4	1.1	0.55	0.55	—	—	—	0.55	0.24	
		eseguita	continuità	231	174	87	34	—	—	—	93	40
			non continuità	234	176	88	88	—	—	—	92	
	doppia precisione	non eseguita	—	—	—	—	—	—	—	—	—	
		eseguita	continuità	—	—	—	—	—	—	—	93	40
			non continuità	—	—	—	—	—	—	—	92	
LDE<=	singola precisione	continuità	235	177	89	34	—	—	—	93	40	
		non continuità	231	174	87	88	—	—	—	92		
	doppia precisione	continuità	—	—	—	—	—	—	—	93	40	
		non continuità	—	—	—	—	—	—	—	92		
ANDE<=	singola precisione	non eseguita	1.4	1.1	0.55	0.55	—	—	—	0.55	0.24	
		eseguita	continuità	234	176	88	34	—	—	—	92	40
			non continuità	230	172	86	86	—	—	—	93	
	doppia precisione	non eseguita	—	—	—	—	—	—	—	—	—	
		eseguita	continuità	—	—	—	—	—	—	—	92	40
			non continuità	—	—	—	—	—	—	—	93	
ORE<=	singola precisione	non eseguita	1.4	1.1	0.55	0.55	—	—	—	0.55	0.24	
		eseguita	continuità	234	176	88	34	—	—	—	92	40
			non continuità	230	172	86	86	—	—	—	93	
	doppia precisione	non eseguita	—	—	—	—	—	—	—	—	—	
		eseguita	continuità	—	—	—	—	—	—	—	92	40
			non continuità	—	—	—	—	—	—	—	93	
LDE<	singola precisione	continuità	231	174	87	35	—	—	—	92	40	
		non continuità	234	176	88	88	—	—	—	93		
	doppia precisione	continuità	—	—	—	—	—	—	—	92	92	
		non continuità	—	—	—	—	—	—	—	93		
ANDE<	singola precisione	non eseguita	1.4	1.1	0.55	0.55	—	—	—	0.55	0.24	
		eseguita	continuità	230	172	86	34	—	—	—	92	40
			non continuità	234	176	88	88	—	—	—	93	
	doppia precisione	non eseguita	—	—	—	—	—	—	—	—	—	
		eseguita	continuità	—	—	—	—	—	—	—	92	40
			non continuità	—	—	—	—	—	—	—	93	
ORE<	singola precisione	non eseguita	1.4	1.1	0.55	0.55	—	—	—	0.55	0.24	
		eseguita	continuità	231	174	87	34	—	—	—	93	40
			non continuità	234	176	88	88	—	—	—	92	
	doppia precisione	non eseguita	—	—	—	—	—	—	—	—	—	
		eseguita	continuità	—	—	—	—	—	—	—	93	40
			non continuità	—	—	—	—	—	—	—	92	
LDE>=	singola precisione	continuità	235	177	89	35	—	—	—	93	40	
		non continuità	231	174	87	87	—	—	—	92		
	doppia precisione	continuità	—	—	—	—	—	—	—	93	40	
		non continuità	—	—	—	—	—	—	—	92		
ANDE>=	singola precisione	non eseguita	1.4	1.1	0.55	0.55	—	—	—	0.55	0.24	
		eseguita	continuità	234	176	88	34	—	—	—	92	40
			non continuità	231	174	87	87	—	—	—	93	
	doppia precisione	non eseguita	—	—	—	—	—	—	—	—	—	
		eseguita	continuità	—	—	—	—	—	—	—	92	40
			non continuità	—	—	—	—	—	—	—	93	

Istruzione	Elaborazione (operando)		Tempo di elaborazione (μs)								
			Moduli CPU serie QnA				Moduli CPU System Q				
			Q2A Q2AS	Q3A	Q4A QnASH	Q4AR	Q00J	Q00	Q01	Q02	QnH
ORE>=	singola precisione	non eseguita	1.4	1.1	0.55	0.55	—	—	—	0.55	0.24
		eseguita	continuità	234	176	88	34	—	—	—	92
	non continuità		230	172	86	86	—	—	—		
	doppia precisione	non eseguita	—	—	—	—	—	—	—	—	—
eseguita		continuità	—	—	—	—	—	—	—	92	40
	non continuità	—	—	—	—	—	—	—			
LD\$=	continuità		97	73	37	37	—	—	—	38	16
	non continuità		81	61	31	31	—	—	—	34	15
AND\$=	non eseguita		1.4	1.1	0.55	0.55	—	—	—	0.56	0.23
	eseguita	continuità	96	72	36	36	—	—	—	39	17
		non continuità	81	61	31	31	—	—	—	32	14
OR\$=	non eseguita		1.4	1.1	0.55	0.55	—	—	—	0.56	0.24
	eseguita	continuità	97	73	37	37	—	—	—	40	17
		non continuità	80	60	30	30	—	—	—	33	14
LD\$<>	continuità		83	62	31	31	—	—	—	32	14
	non continuità		97	73	37	37	—	—	—	40	17
AND\$<>	non eseguita		1.4	1.1	0.55	0.55	—	—	—	0.56	0.23
	eseguita	continuità	80	60	30	30	—	—	—	33	14
		non continuità	96	72	36	36	—	—	—	39	17
OR\$<>	non eseguita		1.4	1.1	0.55	0.55	—	—	—	0.56	0.24
	eseguita	continuità	81	61	31	31	—	—	—	32	14
		non continuità	96	72	36	36	—	—	—	39	17
LD\$>	continuità		83	62	31	31	—	—	—	32	14
	non continuità		97	73	37	37	—	—	—	40	17
AND\$>	non eseguita		1.4	1.1	0.55	0.55	—	—	—	0.56	0.23
	eseguita	continuità	80	60	30	30	—	—	—	33	14
		non continuità	96	72	36	36	—	—	—	39	17
OR\$>	non eseguita		1.4	1.1	0.55	0.55	—	—	—	0.56	0.24
	eseguita	continuità	81	61	31	31	—	—	—	32	14
		non continuità	96	72	36	36	—	—	—	39	17
LD\$<=	continuità		97	73	37	37	—	—	—	40	17
	non continuità		81	61	31	31	—	—	—	32	14
AND\$<=	non eseguita		1.4	1.1	0.55	0.55	—	—	—	0.56	0.23
	eseguita	continuità	96	72	36	36	—	—	—	39	17
		non continuità	81	61	31	31	—	—	—	32	14
OR\$<=	non eseguita		1.4	1.1	0.55	0.55	—	—	—	0.56	0.24
	eseguita	continuità	97	73	37	37	—	—	—	40	17
		non continuità	80	60	30	30	—	—	—	33	14

Istruzione	Elaborazione (operando)	Tempo di elaborazione (µs)									
		Moduli CPU serie QnA				Moduli CPU System Q					
		Q2A Q2AS	Q3A	Q4A QnASH	Q4AR	Q00J	Q00	Q01	Q02	QnH	
LD\$<	continuitr	81	61	31	31	—	—	—	32	14	
	non continuitr	97	73	37	37	—	—	—	40	17	
AND\$<	non eseguita	1.4	1.1	0.55	0.55	—	—	—	0.56	0.23	
	eseguita	continuitr	80	60	30	30	—	—	—	32	14
		non continuitr	96	72	36	36	—	—	—	39	16
OR\$<	non eseguita	1.4	1.1	0.55	0.55	—	—	—	0.56	0.24	
	eseguita	continuitr	81	61	31	31	—	—	—	32	14
		non continuitr	96	72	36	36	—	—	—	39	16
LD\$>=	continuitr	97	73	37	37	—	—	—	40	17	
	non continuitr	81	61	31	31	—	—	—	32	14	
AND\$>=	non eseguita	1.4	1.1	0.55	0.55	—	—	—	0.56	0.23	
	eseguita	continuitr	96	72	36	36	—	—	—	39	16
		non continuitr	81	61	31	31	—	—	—	32	14
OR\$>=	non eseguita	1.4	1.1	0.55	0.55	—	—	—	0.56	0.24	
	eseguita	continuitr	97	73	37	37	—	—	—	39	17
		non continuitr	80	60	30	30	—	—	—	32	14
BKCMP= BKCMP=P	n = 1	120	90	45	45	130	105	97	48	21	
	n = 96	367	276	138	138	205	175	165	142	61	
BKCMP<> BKCMP<>P	n = 1	123	92	46	46	130	105	98	48	21	
	n = 96	346	260	130	130	210	180	165	150	65	
BKCMP> BKCMP>P	n = 1	123	92	96	96	130	105	97	48	21	
	n = 96	366	275	138	138	210	180	165	142	61	
BKCMP>= BKCMP>=P	n = 1	121	91	46	46	130	105	98	48	21	
	n = 96	386	290	145	145	205	175	165	150	65	
BKCMP< BKCMP<P	n = 1	121	91	96	96	130	105	98	48	21	
	n = 96	366	275	138	138	210	180	165	158	68	
BKCMP<= BKCMP<=P	n = 1	121	91	46	46	130	105	97	48	21	
	n = 96	348	261	131	131	205	175	165	150	65	
+ (s,d) +P (s,d)		2.4	1.8	0.9	0.9	1.0	0.80	0.50	0.39	0.17	
+ (s1,s2,d) +P (s1,s2,d)		2.7	2.0	1.0	1.0	1.2	0.96	0.60	0.47	0.20	
- (s,d) -P (s,d)		2.4	1.8	0.9	0.9	1.0	0.80	0.50	0.39	0.17	
- (s1,s2,d) -P (s1,s2,d)		2.6	2.0	1.0	1.0	1.2	0.96	0.60	0.47	0.20	
D+ (s,d) D+P (s,d)		2.8	2.1	1.1	1.1	1.3	1.04	0.65	0.71	.031	
D+ (s1,s2,d) D+P (s1,s2,d)		3.2	2.4	1.2	1.2	1.5	1.2	0.75	0.79	0.34	
D-(s,d) D-P(s,d)		2.8	2.1	1.1	1.1	1.3	1.04	0.65	0.71	0.30	
D- (s1,s2,d) D-P (s1,s2,d)		3.2	2.4	1.2	1.2	1.5	1.2	0.75	0.79	0.34	
x (s1,s2,d) xP (s1,s2,d)		2.8	2.1	1.1	1.1	1.1	0.88	0.55	0.47	0.20	
/ (s1,s2,d) /P (s1,s2,d)		6.8	5.1	2.6	2.6	19	16	15	2.7	1.2	

Istruzione	Elaborazione (operando)		Tempo di elaborazione (µs)								
			Moduli CPU serie QnA				Moduli CPU System Q				
			Q2A Q2AS	Q3A	Q4A QnASH	Q4AR	Q00J	Q00	Q01	Q02	QnH
Dx (s1,s2,d) DxP (s1,s2,d)			20	15	7.5	7.5	41	34	31	7.9	3.4
D/ (s1,s2,d) D/P (s1,s2,d)			36	27	13.5	13.5	28	23	21	14	6.1
B+ (s,d) B+P (s,d)			5.5	4.1	2.1	2.1	34	28	26	2.2	1.0
B+ (s1,s2,d) B+P (s1,s2,d)			13	9.6	4.8	4.8	47	39	37	5.0	2.2
B- (s,d) B-P (s,d)			5.2	3.9	2.0	2.0	34	28	26	2.0	0.9
B- (s1,s2,d) B-P (s1,s2,d)			13	9.4	4.7	4.7	48	40	38	4.9	2.1
DB+ (s,d) DB+P (s,d)			29	22	11	11	58	48	44	12	5.0
DB+ (s1,s2,d) DB+P (s1,s2,d)			32	24	12	12	60	49	46	12	5.3
DB- (s,d) DB-P (s,d)			29	22	11	11	59	48	45	11	4.8
DB- (s1,s2,d) DB-P (s1,s2,d)			32	24	12	12	60	51	45	12	5.2
Bx (s1, s2, d) BxP (s1, s2, d)			9.4	7.1	3.6	3.6	42	35	33	3.7	1.6
B/ (s1, s2, d) B/P (s1, s2, d)			9.4	7.1	3.6	3.6	48	40	37	3.8	1.6
DBx (s1, s2, d) DBxP (s1, s2, d)			62	46	23	23	140	120	110	24	10
DB/ (s1, s2, d) DB/P (s1,s2,d)			69	52	26	26	83	69	65	27	12
E+ (s, d) E+P (s, d)	singola precisione	s = 0, d = 0	54	40	20	35	—	—	—	1.8	0.78
		$s = 2^{127}, d = 2^{127}$	524	394	197						
	doppia precisione	s = 0, d = 0	—	—	—	—	—	—	203	87	
		$s = 2^{127}, d = 2^{127}$	—	—	—						
E+ (s1, s2, d) E+P (s1, s2, d)	singola precisione	s1 = 0, s2 = 0	54	40	20	35	—	—	—	2.4	1.1
		$s1 = 2^{127}, s2 = 2^{127}$	524	394	197						
	doppia precisione	s = 0, d = 0	—	—	—	—	—	—	209	90	
		$s = 2^{127}, d = 2^{127}$	—	—	—						
E- (s, d) E-P (s, d)	singola precisione	s = 0, d = 0	54	40	20	35	—	—	—	1.8	0.78
		$s = 2^{127}, d = 2^{127}$	515	387	194						
	doppia precisione	s = 0, d = 0	—	—	—	—	—	—	202	87	
		$s = 2^{127}, d = 2^{127}$	—	—	—						

Istruzione	Elaborazione (operando)		Tempo di elaborazione (µs)								
			Moduli CPU serie QnA				Moduli CPU System Q				
			Q2A Q2AS	Q3A	Q4A QnASH	Q4AR	Q00J	Q00	Q01	Q02	QnH
E- (s1, s2, d) E-P (s1, s2, d)	singola precisione	s1 = 0, s2 = 0	55	41	21	36	—	—	—	2.4	1.1
		s1 = 2 ¹²⁷ , s2 = 2 ¹²⁷	520	391	146		—	—	—	210	90
	doppia precisione	s = 0, d = 0	—	—	—	—	—	—	—	210	90
		s = 2 ¹²⁷ , d = 2 ¹²⁷	—	—	—	—	—	—	—	210	90
Ex (s1, s2, d) Exp (s1, s2, d)	singola precisione	s1 = 0, s2 = 0	55	41	21	36	—	—	—	2.4	1.1
		s1 = 2 ¹²⁷ , s2 = 2 ¹²⁷	567	426	218		—	—	—	222	96
	doppia precisione	s = 0, d = 0	—	—	—	—	—	—	—	222	96
		s = 2 ¹²⁷ , d = 2 ¹²⁷	—	—	—	—	—	—	—	222	96
E/ (s1, s2, d) E/P (s1, s2, d)	singola precisione	s1 = 0, s2 = 1	149	112	56	37	—	—	—	12	5.2
		s1 = 2 ¹²⁷ , s2 = -2 ¹²⁶	1109	834	417		—	—	—	369	159
	doppia precisione	s = 0, d = 0	—	—	—	—	—	—	—	369	159
		s1 = 2 ¹²⁷ , s2 = -2 ¹²⁶	—	—	—	—	—	—	—	369	159
\$+ (s, d) \$+P (s, d)		179	134	67	67	—	—	—	68	29	
\$+ (s1, s2, d) \$+P (s1, s2, d)		206	155	78	78	—	—	—	81	35	
INC INCP		1.9	1.4	0.7	0.7	0.70	0.56	0.35	0.32	0.14	
DINC DINCP		2.3	1.7	0.9	0.9	0.90	0.72	0.45	0.47	0.20	
DEC DECP		1.9	1.4	0.7	0.7	0.70	0.56	0.35	0.32	0.14	
DDEC DDECP		2.3	1.7	0.9	0.9	0.90	0.72	0.45	0.47	0.20	
BCD BCDP		2.7	2.0	1.0	1.0	20	16	15	1.1	0.48	
DBCD DBCDP		7.9	5.9	3.0	3.0	26	21	20	3.2	1.4	
BIN BINP		2.7	2.0	1.0	1.0	19	16	15	1.0	0.44	
DBIN DBINP		4.8	3.6	1.8	1.8	22	18	17	1.9	0.82	
INT INTP	singola precisione	s = 0	20	15	7.5	7.5	—	—	—	3.2	1.4
		s = 32766.5	54	40	20	20					
	doppia precisione	s = 0	—	—	—	—	—	—	—	22	9.3
		s = 32766.5	—	—	—	—					
DINT DINTP	singola precisione	s = 0	20	15	7.5	7.5	—	—	—	2.5	1.1
		s = 1234567890.3	59	44	22	22					
	doppia precisione	s = 0	—	—	—	—	—	—	—	24	10
		s = 1234567890.3	—	—	—	—					
FLT FLTP	singola precisione	s = 0	27	20	10	10	—	—	—	2.1	0.92
		s = 7FFF _H	55	41	21	21					
	doppia precisione	s = 0	—	—	—	—	—	—	—	22	9.6
		s = 7FFF _H	—	—	—	—					
DFLT DFLTP	singola precisione	s = 0	28	21	11	11	—	—	—	2.1	0.88
		s = 7FFFFFFF _H	56	42	21	21					
	doppia precisione	s = 0	—	—	—	—	—	—	—	26	11
		s = 7FFFFFFF _H	—	—	—	—					
DBL DBLP		12	8.6	4.3	4.3	19	16	15	4.5	1.9	

Istruzione	Elaborazione (operando)	Tempo di elaborazione (µs)								
		Moduli CPU serie QnA				Moduli CPU System Q				
		Q2A Q2AS	Q3A	Q4A QnASH	Q4AR	Q00J	Q00	Q01	Q02	QnH
WORD WORDP		12	9.0	4.5	4.5	23	19	17	4.7	2.0
GRY GRYP		12	9.0	4.5	4.5	19	16	15	4.7	2.0
DGRY DGRYP		14	10	5.0	5.0	23	19	17	5.3	2.3
GBIN GBINP		46	34	17	17	52	42	40	18	7.7
DGBIN DGBINP		83	62	31	31	110	88	84	32	14
NEG NEGP		9.3	7	3.5	3.5	16	13	12	3.6	1.6
DNEG DNEGP		11	8.2	4.1	4.1	19	17	15	4.3	1.8
ENEG ENEGP		9.8	7.4	3.7	3.7	—	—	—	3.9	1.7
BKBCD (s, d, n) BKBCDP (s, d, n)	n = 1	102	76	38	38	78	63	57	38	17
	n = 96	272	204	102	102	315	275	250	99	43
BKBIN (s, d, n) BKBINP (s, d, n)	n = 1	102	76	38	38	74	61	57	38	17
	n = 96	272	204	102	102	285	255	230	99	43
MOV MOVVP	s = D0, d = D1	0.7	0.5	0.3	0.3	0.70	0.56	0.35	0.24	0.10
	s = D0, d = J1/W1	392 ¹	305 ¹	176 ¹	176 ¹	155	130	120	140	160
		391 ²	299 ²	165 ²	165 ²					
DMOV DMOVVP	s = K4X0, d = D1	2.4	1.8	0.9	0.9	0.90	0.72	0.45	0.47	0.20
	s = K4X0, d = J1/W1	400 ¹	313 ¹	183 ¹	183 ¹	165	135	120	147	64
		395 ²	301 ²	167 ²	167 ²					
EMOV EMOVVP		12	8.6	4.3	4.3	—	—	—	0.63	0.27
\$MOV \$MOVVP		100	75	38	38	46	38	35	40	17
						98	80	73		
CML CMLP		2.0	1.5	0.8	0.8	0.70	0.56	0.35	0.40	0.17
DCML DCMLP		2.4	1.8	0.9	0.9	0.90	0.72	0.45	0.55	0.24
BMOV (s, d, n) BMOVVP (s, d, n)	n = 1	43	32	16	16	27	21	20	17	7.1
	n = 96	81	61	31	31	72	62	53	32	14
FMOV (s, d, n) FMOVVP (s, d, n)	n = 1	18	13	6.5	6.5	23	19	17	6.7	2.9
	n = 96	36	27	14	14	48	41	36	14	6.1
XCH XCHP		3.1	2.3	1.2	1.2	7.6	6.3	5.7	1.3	0.54
DXCH DXCHP		3.1	2.3	1.2	1.2	9.5	8.0	7.1	1.3	0.54
BXCH (d1, d2, n) BXCHP (d1, d2, n)	n = 1	77	58	29	29	62	51	48	31	13
	n = 96	213	160	80	80	165	140	125	84	36
SWAP SWAPP		9.2	6.9	3.5	3.5	17	14	13	3.7	1.6

¹ Questi tempi di elaborazione sono validi quando viene usato un telaio principale A38B/A1S38B con un telaio di espansione.

² Questi tempi di elaborazione sono validi per un telaio principale A38HB/A1S38HB.

Istruzione	Elaborazione (operando)	Tempo di elaborazione (µs)								
		Moduli CPU serie QnA				Moduli CPU System Q				
		Q2A Q2AS	Q3A	Q4A QnASH	Q4AR	Q00J	Q00	Q01	Q02	QnH
CJ		7.8	5.8	2.9	2.9	10	8.5	8.1	3.2	1.4
SCJ		7.8	5.8	2.9	2.9	10	8.5	8.1	3.2	1.4
JMP		8.0	6.0	3.0	3.0	11	8.5	8.1	3.2	1.4
GOEND		2.0	1.5	0.75	0.75	3.3	2.9	2.8	0.39	0.34
EI		3.1	2.3	1.2	1.2	14	11	11	1.3	0.54
DI		2.3	1.7	0.9	0.9	13	12	11	0.95	0.41
IMASK		8.1	6.5	3.3	3.3	41	34	35	11	4.6
IRET		4.0	3.0	1.5	1.5	205	170	155	1.6	0.68
RFS RFSP	s = X, n = 1	31.3	23.4	11.7	11.7	55	46	43	6.7	4.7
	s = Y, n = 1					54	45	41		
	s = X, n = 96	97.6	72.8	36.4	36.4	79	64	59	19	13
	s = Y, n = 96					73	61	56		
UDCNT1		42.6	31.8	15.9	15.9	—	—	—	15	6.5
UDCNT2		44.6	33.3	16.7	16	—	—	—	16	6.8
TTMR		25.9	19.3	9.7	9.7	—	—	—	10	4.4
STMR		41.7	31.1	15.6	15.6	—	—	—	20	7.1
ROTC		66.1	49.3	24.7	24.7	—	—	—	26	11
RAMP		45.4	33.9	17.0	17.0	—	—	—	18	7.7
SPD		48.9	36.5	18.3	18.3	—	—	—	19	8.3
PLSY		26.9	20.1	10.1	10.1	—	—	—	10	4.5
PWM		32.8	24.5	12.3	12.3	—	—	—	9.1	3.9
MTR		29.2	21.8	10.9	10.9	—	—	—	11	4.9
WAND (s, d) WANDP (s, d)		2.4	1.8	0.9	0.9	1.0	0.80	0.50	0.39	0.17
WAND (s1, s2, d) WANDP (s1, s2, d)		9.5	7.1	3.6	3.6	1.2	0.96	0.60	0.47	0.20
DAND (s, d) DANDP (s, d)		3.0	2.3	1.2	1.2	1.3	1.04	0.65	0.71	0.31
DAND (s1, s2, d) DANDP (s1, s2, d)		19	14	7.0	7.0	1.5	1.2	0.75	0.79	0.34
BKAND (s1, s2, d, n) BKANDP (s1, s2, d, n)	n = 1	89	67	34	34	110	87	79	36	16
	n = 96	184	138	69	69	185	155	140	74	32
WOR (s, d) WORP (s, d)		2.4	1.8	0.9	0.9	1.0	0.80	0.50	0.40	0.17
WOR (s1, s2, d) WORP (s1, s2, d)		9.5	7.1	3.6	3.6	1.2	0.96	0.60	0.47	0.20
DOR (s, d) DORP (s, d)		3.0	2.3	1.2	1.2	1.3	1.04	0.65	0.71	0.31
DOR (s1, s2, d) DORP (s1, s2, d)		19	14	7.0	7.0	1.5	1.2	0.75	0.79	0.34
BKOR (s1, s2, d, n) BKORP (s1, s2, d, n)	n = 1	89	67	34	34	110	87	81	36	16
	n = 96	184	138	69	69	185	155	140	74	32

Istruzione	Elaborazione (operando)	Tempo di elaborazione (µs)								
		Moduli CPU serie QnA				Moduli CPU System Q				
		Q2A Q2AS	Q3A	Q4A QnASH	Q4AR	Q00J	Q00	Q01	Q02	QnH
WXOR (s, d) WXORP (s, d)		2.4	1.8	0.9	0.9	1.0	0.80	0.50	0.39	0.17
WXOR (s1, s2, d) WXORP (s1, s2, d)		17.2	7.1	3.6	3.6	1.2	0.96	0.60	0.47	0.20
DXOR (s, d) DXORP (s, d)		3.0	2.3	1.2	1.2	1.3	1.04	0.65	0.71	0.31
DXOR (s1, s2, d) DXORP (s1, s2, d)		19	14	7.0	7.0	1.5	1.2	0.75	0.79	0.34
BKXOR (s1, s2, d, n)	n = 1	89	67	34	34	110	87	81	36	16
BKXORP (s1, s2, d, n)	n = 96	184	138	69	69	183	155	140	74	32
WXNR (s, d) WXNRP (s, d)		2.4	1.8	0.9	0.9	1.0	0.80	0.50	0.40	0.17
WXNR (s1, s2, d) WXNRP (s1, s2, d)		9.5	7.1	3.6	3.6	1.2	0.96	0.60	0.47	0.20
DXNR (s,d) DXNRP (s,d)		3.0	2.3	1.2	1.2	1.3	1.04	0.65	0.71	0.31
DXNR (s1,s2,d) DXNRP (s1,s2,d)		24	18	9	9	1.5	1.2	0.75	0.79	0.34
BKXNR (s1, s2, d, n)	n = 1	89	67	34	34	110	87	82	36	16
BKXNR (s1, s2, d, n)	n = 96	184	138	69	69	185	155	140	74	32
ROR (d, n)	n = 1	5.0	3.8	1.9	1.9	13	11	9.7	2.0	0.85
RORP (d, n)	n = 15	5.0	3.8	1.9	1.9	13	11	9.7	2.0	0.85
RCR (d, n)	n = 1	4.0	3.0	1.5	1.5	15	12	12	1.6	0.68
RCRP (d, n)	n = 15	4.0	3.0	1.5	1.5	15	13	12	1.6	0.68
ROL (d, n)	n = 1	5.0	3.8	1.9	1.9	13	11	10	2.0	0.85
ROLP (d, n)	n = 15	5.0	3.8	1.9	1.9	13	11	10	2.0	0.85
RCL (d, n)	n = 1	4.0	3.0	1.5	1.5	15	13	12	1.6	0.68
RCLP (d, n)	n = 15	4.0	3.0	1.5	1.5	16	13	12	1.6	0.68
DROR (d, n)	n = 1	9.8	7.4	3.7	3.7	15	12	12	3.9	1.7
DRORP (d, n)	n = 31	10	7.8	3.9	3.9	15	13	12	4.0	1.7
DRCR (d, n)	n = 1	11	8.1	4.1	4.1	17	14	14	4.3	1.8
DRCRP (d, n)	n = 31	11	8.3	4.2	4.2	18	16	15	4.3	1.9

Istruzione	Elaborazione (operando)		Tempo di elaborazione (µs)								
			Moduli CPU serie QnA				Moduli CPU System Q				
			Q2A Q2AS	Q3A	Q4A QnASH	Q4AR	Q00J	Q00	Q01	Q02	QnH
DROL (d, n) DROLP (d, n)	n = 1		9.8	7.4	3.7	3.7	14	13	12	3.9	1.7
	n = 31		10	7.8	3.9	3.9	14	13	12	4.0	1.7
DRCL (d, n) DRCLP (d, n)	n = 1		11	8.1	4.1	4.1	18	15	14	4.3	1.8
	n = 31		11	8.3	4.2	4.2	20	17	16	4.3	1.9
SFR (d, n) SFRP (d, n)	n = 1		4.4	3.3	1.7	1.7	13	10	9.7	1.7	0.75
	n = 15		5.0	3.8	1.9	1.9	13	11	9.5	2.0	0.85
SFL (d, n) SFLP (d, n)	n = 1		4.4	3.3	1.7	1.7	12	10	9.5	1.7	0.75
	n = 15		5.0	3.8	1.9	1.9	12	9.8	9.5	2.0	0.85
BSFLR (d, n) BSFLRP (d, n)	n = 1		51	38	19	19	42	35	33	20	8.6
	n = 96		60	45	23	23	69	58	54	24	10
BSFL (d, n) BSFLP (d, n)	n = 1		49	37	19	19	41	34	32	20	8.5
	n = 96		58	44	22	22	63	53	50	23	10
DSFR (d, n) DSFRP (d, n)	n = 1		3.6	2.6	1.3	1.3	19	16	15	1.3	0.58
	n = 96		63	47	24	24	71	61	53	25	11
DSFL (d, n) DSFLP (d, n)	n = 1		3.6	2.6	1.3	1.3	19	16	15	1.3	0.58
	n = 96		65	49	25	25	70	60	52	26	11
BSET (d, n) BSETP (d, n)	n = 1		20	15	7.5	7.5	27	22	20	7.6	3.3
	n = 15		20	15	7.5	7.5	27	22	20	7.6	3.3
BRST (d, n) BRSTP (d, n)	n = 1		20	15	7.5	7.5	27	22	21	7.6	3.3
	n = 15		20	15	7.5	7.5	27	22	21	7.6	3.3
TEST (s1, s2, d) TESTP (s1, s2, d)			21	16	8.0	8.0	35	30	27	8.2	3.5
DTEST (s1, s2, d) DTESTP (s1, s2, d)			24	18	9.0	9.0	37	31	28	9.2	3.9
BKRST (s, n) BKRST (s, n)	n = 1		45	34	17	17	49	41	38	18	7.8
	n = 96		49	37	19	19	64	54	50	19	8.2
SER (s1, s2, d, n) SERP (s1, s2, d, n)	n = 1	corrispondenza	58	44	22	22	56	54	42	22	9.6
		nessuna corrispondenza	57	43	21	21	56	54	42	21	8.9
	n = 96	corrispondenza	293	220	110	110	280	240	220	115	49
		nessuna corrispondenza	340	256	128	128	280	240	220	133	57
DSER (s1, s2, d, n) DSERP (s1, s2, d, n)	n = 1	corrispondenza	61	46	23	23	71	67	53	23	9.9
		nessuna corrispondenza	58	44	22	22	71	67	54	23	9.7
	n = 96	corrispondenza	354	266	133	133	495	415	375	142	61
		nessuna corrispondenza	354	266	133	133	500	415	375	132	57
SUM SUMP	s = 0		9.8	7.4	3.7	3.7	32	26	25	3.9	1.7
							27	22	21		
DSUM DSUMP	s = 0		12	9.0	4.5	4.5	54	44	42	4.7	2.0
							31	23	12		

Istruzione	Elaborazione (operando)		Tempo di elaborazione (µs)								
			Moduli CPU serie QnA				Moduli CPU System Q				
			Q2A Q2AS	Q3A	Q4A QnASH	Q4AR	Q00J	Q00	Q01	Q02	QnH
DECO (s, d, n) DECOP (s, d, n)	n = 2		48	36	18	18	60	50	46	20	8.6
	n = 8		62	47	24	24	80	65	61	27	12
ENCO (s, d, n) ENCOP (s, d, n)	n = 2	M1 = ON	52	39	20	20	66	55	51	21	9.1
		M4 = ON	52	39	20	20	66	54	51	21	9.1
	n = 8	M1 = ON	65	49	25	25	90	76	71	28	12
		M256 = ON	65	49	25	25	76	74	71	26	11
SEG SEGP			3.2	2.4	1.2	1.2	8.0	6.8	6.1	1.3	0.54
DIS (s, d, n) DISP (s, d, n)	n = 1		46	34	17	17	47	39	36	18	7.7
	n = 4		51	38	19	19	53	43	40	19	8.3
UNI (s, d, n) UNIP (s, d, n)	n = 1		53	40	20	20	54	44	41	21	8.9
	n = 4		57	43	22	22	60	49	46	23	9.7
NDIS (s1, d, s2) NDISP (s1, d, s2)			104	78	39	39	92	76	38	41	18
NUNI (s1, d, s2) NUNIP (s1, d, s2)			105	79	40	40	47	39	36	42	18
WTOB (s, d, n) WTOBP (s, d, n)	n = 1		125	94	47	47	56	46	42	47	20
	n = 96		257	193	97	97	190	155	145	99	43
BTOW (s, d, n) BTOWP (s, d, n)	n = 1		121	91	46	46	56	46	42	45	19
	n = 96		233	175	88	88	190	155	145	89	38
MAX (s, d, n) MAXP (s, d, n)	n = 1		43	32	16	16	48	40	36	17	7.1
	n = 96		318	239	120	120	300	240	235	136	59
MIN (s, d, n) MINP (s, d, n)	n = 1		43	32	16	16	48	40	36	17	7.1
	n = 96		436	326	163	163	300	240	235	159	69
DMAX (s, d, n) DMAXP (s, d, n)	n = 1		71	53	27	27	52	43	39	27	12
	n = 96		427	321	161	161	600	490	460	181	78
DMIN (s, d, n) DMINP (s, d, n)	n = 1		71	53	27	27	52	43	39	27	12
	n = 96		268	201	101	101	585	475	445	112	48
SORT (s1, n, s2, d1, d2)	n = 1		43	32	16	16	66	55	50	16	7.1
	n = 96		40*	30*	15*	15*	105	86	80	14	6.2
DSORT (s1, n, s2, d1, d2)	n = 1		44	33	17	17	98	57	52	17	7.1
	n = 96		43*	32*	16*	16*	115	96	88	16	6.8
* Indica allungamento del tempo di scansione al completamento dell'istruzione											
WSUM (s, d, n) WSUMP (s, d, n)	n = 1		41.5	31.1	15.6	15.6	52	43	40	16.4	7.1
	n = 96		173.2	129.9	65	65	175	140	135	68.4	29.5
DWSUM (s, d, n) DWSUMP (s, d, n)	n = 1		47.9	35.9	18	18	61	51	46	18.9	8.2
	n = 96		330	247.5	123.8	123.8	515	420	395	130.4	56.1
FOR	n = 0		5.2	3.9	2.0	2.0	11	8.9	8.1	2.3	1.0
NEXT			8.0	6.0	3.0	3.0	8.8	7.3	6.8	3.3	1.4

Istruzione	Elaborazione (operando)	Tempo di elaborazione (µs)									
		Moduli CPU serie QnA				Moduli CPU System Q					
		Q2A Q2AS	Q3A	Q4A QnASH	Q4AR	Q00J	Q00	Q01	Q02	QnH	
BREAK BREAKP		26	19	9.5	9.5	37	30	28	11	4.6	
CALL (pn) CALLP (pn)	puntatore file interno	5.1	3.8	1.9	1.9	17	14	13	2.1	0.88	
	puntatore file comune	85	64	32	32				33	14	
CALL (pn s1 - s5) CALLP (pn s1 - s5)		348	261	131	131	245	200	190	135	58	
RET	Ritorno al programma originale	7.5	5.6	2.8	2.8	16	13	12	2.9	1.3	
	Ritorno all'altro programma	51	38	19	19	—	—	—	20	8.5	
FCALL (pn) FCALLP (pn)	puntatore file interno	8.8	6.6	3.3	3.3	29	24	22	3.6	1.6	
	puntatore file comune	48	36	18	18				20	8.7	
FCALL (pn S1 - S5) FCALLP (pn S1 - S5)		388	254	127	127	250	205	190	134	57	
ECALL (pn) ECALLP (pn)		187	140	70	70	—	—	—	77	33	
ECALL (pn S1 - S5) ECALLP (pn S1 - S5)		515	387	144	144	—	—	—	162	70	
EFCALL (pn) EFCALLP (pn)		188	141	71	71	—	—	—	78	34	
EFCALL (pn S1 - S5) EFCALLP (pn S1 - S5)		516	388	194	194	—	—	—	200	86	
COM		137	103	52	52	110	77	72	55	16	
IX		31	23	12	12	65	54	51	12	5.2	
IXEND		12	8.9	4.5	4.5	30	26	25	4.7	2.0	
IXDEV	numero di contatti: 1	127	95	46	46	145	120	110	48	21	
	numero di contatti: 14	238	179	85	85	770	630	585	93	40	
IXSET	numero di contatti: 1	127	95	46	46	145	120	110	48	21	
	numero di contatti: 14	238	179	85	85	770	630	585	93	40	
FIFW FIFWP	numero dati: 1	27	20	10	10	36	32	28	11	4.5	
	numero dati: 96										
FIFR FIFRP	numero dati: 1	34	25	13	13	45	41	36	13	5.6	
	numero dati: 96	79	59	30	30	93	82	70	32	14	
FPOP FPOPP	numero dati: 1	46	34	17	17	40	37	32	16	7.0	
	numero dati: 96										
FINS FINSP	numero dati: 1	48	36	18	18	53	44	38	20	8.4	
	numero dati: 96	96	72	36	36	100	89	76	36	15	
FDEL FDELP	numero dati: 1	47	35	18	18	60	50	43	19	7.5	
	numero dati: 96	97	73	37	37	110	95	82	39	15	
PR	SM701 ON	1 carattere	83	62	31	31	—	—	—	33	11
		32 caratteri	123	92	46	46				48	18
	SM701 OFF	54	40	20	20	21				7.8	
PRC		400	301	151	151	—	—	—	181	16	
LED	visualizzato	223	167	84	84	—	—	—	—	—	
	visualizzazione terminata	79	59	30	30						
LEDC	visualizzato	559	420	210	210	—	—	—	—	—	
	visualizzazione terminata	413	310	155	155						

Istruzione	Elaborazione (operando)		Tempo di elaborazione (µs)								
			Moduli CPU serie QnA				Moduli CPU System Q				
			Q2A Q2AS	Q3A	Q4A QnASH	Q4AR	Q00J	Q00	Q01	Q02	QnH
LEDR	nessuna visualizzazione → nessuna visualizzazione		18	13	6.5	6.5	—	—	—	0.40	0.17
	Esecuzione istruzione LED → nessuna visualizzazione		205	154	77	77	—	—	—	103	44
CHKST			15	11	5.5	5.5	—	—	—	5.8	2.5
CHK (controllo errori)	1 contatto d'ingresso	nessun errore su contatto 1	61	46	23	23	—	—	—	24	10
	150 contatti d'ingresso	nessun errore su contatto 150	4232	3182	1591	1591				1676	721
		nessun errore su contatto 1	224	168	84	84				88	38
CHKCIR	10 circuiti di controllo errore		15	11	5.5	5.5	—	—	—	5.8	2.5
SLT	tutti operandi interni		2399	1804	902	902	—	—	—	—	—
	file registri 8 kByte		7254	5454	2727	2727					
	completamento istruzione SLT		15	11	5.5	5.5					
SLTR			1.1	0.8	0.4	0.4	—	—	—	—	—
STRA	Start		47	35	18	18	—	—	—	—	—
	completamento istruzione STRA		15	11	5.5	5.5					
STRAR			1.1	0.8	0.4	0.4	—	—	—	—	—
PTRA			15	11	5.5	5.5	—	—	—	—	—
PTRAR			15	11	5.5	5.5	—	—	—	—	—
PTRAXE PTRAXEP	esecuzione istruzione		1.6	1.2	0.6	0.6	—	—	—	—	—
	traccia programma		169	127	64	64					
BINDA BINDAP	s = 1		40	30	15	15	—	—	—	15	6.7
	s = -32768		60	45	23	23				24	10
DBINDA DBINDAP	s = 1		63	47	44	44	—	—	—	43	18
	s = -2147483648		217	163	82	82				86	37
BINHA BINHAP	s = 1		46	34	17	17	—	—	—	18	7.7
	s = FFFF _H		48	36	18	18				19	8.2
DBINHA DBINHAP	s = 1		59	44	22	22	—	—	—	23	10
	s = FFFFFFFF _H		62	46	23	23				24	10
BCDDA BCDDAP	s = 1		58	43	22	22	—	—	—	23	9.8
	s = 9999		54	40	20	20				21	8.9
DBCDDA DBCDDAP	s = 1		61	46	23	23	—	—	—	22	9.5
	s = 99999999		75	56	28	28				29	13
DABIN DABINP	s = 1		133	100	50	50	—	—	—	57	25
	s = -32768		145	109	55	55				58	28
DDABIN DDABINP	s = 1		241	181	91	91	—	—	—	92	40
	s = -2147483648		268	201	101	101				106	46
HABIN HABINP	s = 1		32	24	12	12	—	—	—	13	5.8
	s = FFFF _H		38	28	14	14				15	6.4
DHABIN DHABINP	s = 1		54	40	20	20	—	—	—	22	9.5
	s = FFFFFFFF _H		63	47	24	24				25	11
DABCD DABCDP	s = 1		36	27	14	14	—	—	—	16	6.9
	s = 9999		42	31	16	16				17	7.2
DDABCD DDABCDP	s = 1		63	47	24	24	—	—	—	25	11
	s = 99999999		75	56	28	28				29	13
COMRD COMRDP			36	27	14	14	—	—	—	40	16
LEN LENP	1 carattere		48	36	18	18	—	—	—	18	8.0
	96 caratteri		229	172	86	86				86	37
STR STRP			132	99	50	50	—	—	—	53	23
DSTR DSTRP			285	214	107	107	—	—	—	123	53
VAL VALP			258	194	97	97	—	—	—	95	41
DVAL DVALP			402	302	151	151	—	—	—	166	72

Istruzione	Elaborazione (operando)		Tempo di elaborazione (µs)								
			Moduli CPU serie QnA				Moduli CPU System Q				
			Q2A Q2AS	Q3A	Q4A QnASH	Q4AR	Q00J	Q00	Q01	Q02	QnH
ESTR ESTRP			1337	1005	503	503	—	—	—	564	243
EVAL EVALP	formato virgola mobile		242	182	91	91	—	—	—	100	43
	formato esponenziale		306	230	115	115	—	—	—	127	55
ASC (s, d, n) ASCP (s, d, n)	n = 1		164	123	62	62	—	—	—	64	28
	n = 96		780	586	293	293				289	125
HEX (s, d, n) HEXP (s, d, n)	n = 1		161	121	61	61	—	—	—	60	26
	n = 96		826	621	311	311				343	148
RIGHT (s, d, n) RIGHTP (s, d, n)	n = 1		131	98	49	49	—	—	—	49	21
	n = 96		354	266	133	133				131	56
LEFT (s, d, n) LEFTP (s, d, n)	n = 1		129	97	49	49	—	—	—	50	21
	n = 96		354	266	133	133				131	56
MIDR MIDRP			141	106	53	53	—	—	—	53	23
MIDW MIDWP			341	256	128	128	—	—	—	128	55
INSTR INSTRP	nessuna corrispondenza		156	117	59	59	—	—	—	58	25
	corrispondenza	primo	141	106	53	53				55	24
		ultimi	155	116	58	58				58	25
EMOD EMODP			1313	987	494	494	—	—	—	527	227
EREXP EREXPP			4423	3325	1663	1663	—	—	—	1656	713
SIN SINP	singola precisione		4921	3700	1850	35	—	—	—	115	50
	doppia precisione		—	—	—	—				1945	837
COS COSP	singola precisione		6462	4858	2429	35	—	—	—	122	53
	doppia precisione		—	—	—	—				2618	1127
TAN TANP	singola precisione		6515	4898	2449	38	—	—	—	123	53
	doppia precisione		—	—	—	—				2618	1127
ASIN ASINP	singola precisione		890	669	335	44	—	—	—	111	48
	doppia precisione		—	—	—	—				2491	1072
ACOS ACOSP	singola precisione		801	602	301	44	—	—	—	115	49
	doppia precisione		—	—	—	—				2367	1019
ATAN ATANP	singola precisione		7818	5878	2939	39	—	—	—	157	68
	doppia precisione		—	—	—	—				3140	1352
RAD RADP	singola precisione		465	349	175	31	—	—	—	17	7.2
	doppia precisione		—	—	—	—				24	10
DEG DEGP	singola precisione		492	369	185	31	—	—	—	17	7.2
	doppia precisione		—	—	—	—				23	9.9
SQR SQRP	singola precisione		4520	3398	1699	39	—	—	—	28	12
	doppia precisione		—	—	—	—				1812	780
EXP EXPP	singola precisione	s = -10	5871	4414	2207	37	—	—	—	—	—
		s = 1	5950	4474	2237					129	56
	doppia precisione	s = -10	—	—	—	—				—	—
		s = 1	—	—	—	—				2386	1026
LOG LOGP	singola precisione	s = 1	1191	896	448	37	—	—	—	113	49
		s = 10	6839	5142	2571					—	—
	doppia precisione	s = 1	—	—	—	—				2146	924
		s = 10	—	—	—	—				—	—
RND RNDP			10	7.5	3.8	3.8	—	—	—	3.9	1.7

Istruzione	Elaborazione (operando)	Tempo di elaborazione (µs)								
		Moduli CPU serie QnA				Moduli CPU System Q				
		Q2A Q2AS	Q3A	Q4A QnASH	Q4AR	Q00J	Q00	Q01	Q02	QnH
SRND SRNDP		8.8	6.6	3.3	3.3	—	—	—	3.5	1.5
BSQR BSQRP	s = 0	16	12	6.0	6.0	—	—	—	6.2	2.7
	s = 9999	97	73	37	37				38	16
BDSQR BDSQRP	s = 0	17	13	6.5	6.5	—	—	—	6.2	2.7
	s = 99999999	88	66	33	33				38	16
BSIN BSINP		30	22	11	11	—	—	—	12	5.1
BCOS BCOSP		32	24	12	12	—	—	—	12	5.2
BTAN BTANP		30	22	11	11	—	—	—	12	5.2
BASIN BASINP		52	39	20	20	—	—	—	20	8.7
BACOS BACOSP		53	40	20	20	—	—	—	21	9.0
BATAN BATANP		56	42	21	21	—	—	—	22	9.6
LIMIT LIMITP		24	18	9.0	9.0	34	28	26	10	4.3
DLIMIT DLIMITP		28	21	11	11	41	34	30	11	4.7
BAND BANDP		24	18	9.0	9.0	33	28	25	9.8	4.2
DBAND DBANDP		28	21	11	11	40	34	30	11	4.9
ZONE ZONEP		24	18	9.0	9.0	31	25	24	9.1	3.9
DZONE DZONEP		28	21	11	11	37	29	28	11	4.6
RSET RSETP		19	14	7.0	7.0	—	18	16	6.8	2.9
QDRSET QDRSETP		322	242	121	121	—	—	—	205	88
QCDSSET QCDSSETP		218	164	82	82	—	—	—	147	63
DATERD DATERDP		36	27	14	14	30	25	23	13	5.5
DATEWR DATEWRP		42	31	16	16	69	57	54	15	6.4
DATE+ DATE+P	senza incremento cifra	60	45	23	23	47	39	36	13	5.4
	incremento cifra	60	45	23	23	50	42	38	13	5.4
DATE- DATE-P	senza incremento cifra	59	44	22	22	47	40	36	12	5.2
	incremento cifra	60	45	23	23	50	42	38	12	5.2
SECOND SECONDP		27	20	10	10	28	24	22	10	4.5
HOUR HOURP		31	23	12	12	38	32	29	12	5.2
MSG	1 carattere	7.2	5.4	2.7	2.7	—	—	—	3.0	1.3
	32 caratteri	7.4	5.6	2.8	2.8					
PKEY	tempo iniziale	51	38	19	19	—	—	—	20	8.6
	nessun tasto premuto	48	36	18	18				19	8.2
PSTOP PSTOPP		122	92	46	46	—	—	—	79	34
POFF POFFP		120	90	45	45	—	—	—	79	34
PSCAN PSCANP		122	92	46	46	—	—	—	75	32

Istruzione	Elaborazione (operando)	Tempo di elaborazione (µs)								
		Moduli CPU serie QnA				Moduli CPU System Q				
		Q2A Q2AS	Q3A	Q4A QnASH	Q4AR	Q00J	Q00	Q01	Q02	QnH
PLOW PLOWP		124	93	47	47	—	—	—	80	34
WDT WDTP		12	8.7	4.4	4.4	18	15	14	5.9	2.6
DUTY		1.6	1.2	0.6	0.6	41	36	32	9.3	4.0
ZRRDB ZRRDBP		19	14	6.9	6.9	—	24	22	7.9	3.4
ZRWRB ZRWRBP		21	16	7.8	7.8	—	27	24	9.4	4.0
ADRSET ADRSETP		13	9.3	4.7	4.7	23	19	18	4.9	2.1
KEY		43.4	32.4	16.2	16.2	—	—	—	17	7.3
ZPUSH ZPUSHP		27.6	20.6	10.3	10.3	38	33	30	11	4.7
ZPOP ZPOPP		12.7	9.5	4.8	4.8	37	31	29	5.1	2.2
EPROMWR EPROMWRP		62.6	46.7	23.4	23.4	—	—	—	—	—
ZCOM		4296.6	3206.4	1603.2	1603.2	105	82	80	691	289
READ		770.6	575.1	287.6	287.6	—	—	—	554	260
SREAD		858.9	641.0	320.5	320.5	—	—	—	588	278
WRITE		791.9	591.0	295.5	295.5	—	—	—	582	273
SWRITE		848.6	633.3	316.6	316.6	—	—	—	625	295
SEND		575.7	429.6	214.8	214.8	—	—	—	—	—
RECV		375.9	280.5	140.3	140.3	—	—	—	—	—
REQ		527.4	393.6	196.8	196.8	—	—	—	—	—
ZNFR		982.1	732.9	366.5	366.5	—	—	—	—	—
ZNTO		989.3	738.3	369.2	369.2	—	—	—	—	—
ZNRD	MELSECNET/10	598.6	446.7	223.4	223.4	—	—	—	—	—
	MELSECNET (II)	649.2	484.5	242.3	242.3	—	—	—	—	—
ZNWR	MELSECNET/10	614.3	458.4	229.2	229.2	—	—	—	—	—
	MELSECNET (II)	665.6	496.7	248.4	248.4	—	—	—	—	—
RFRP		590.9	441.0	220.5	220.5	—	—	—	—	—
RTOP		588.8	439.4	219.7	219.7	—	—	—	—	—
S.STMODE		—	—	—	22.6	—	—	—	—	—
S.CGMODE		—	—	—	19.4	—	—	—	—	—
S.TRUCK		—	—	—	43.7	—	—	—	—	—
S.SPREF	Indirizzo iniziale buffer di memoria = 0 (con A68AD)	—	—	—	407.1	—	—	—	—	—
	Indirizzo iniziale buffer di memoria = 0 (con A68DA)	—	—	—	331.4	—	—	—	—	—
UNIRD	n = 1	—	—	—	—	96	80	74	79	34
	n = 16	—	—	—	—	440	370	340		
TRACE	Start	—	—	—	—	—	—	—	176	76
	Completamento istruzione TRACE	—	—	—	—	—	—	—	6.3	2.7
TRACER		—	—	—	—	—	—	—	19	8.2
FWRITE		—	—	—	—	—	—	—	84	36
FREAD		—	—	—	—	—	—	—	79	34
PLOADP		—	—	—	—	—	—	—	58	25
PUNLOADP		—	—	—	—	—	—	—	272	117
PSWAPP		—	—	—	—	—	—	—	308	133
RBMOV	Durata trasferimento	1 word	—	—	—	—	—	—	69	29
		1000 word	—	—	—	—	—	—	580	308

Istruzione	Elaborazione (operando)		Tempo di elaborazione (µs)								
			Moduli CPU serie QnA				Moduli CPU System Q				
			Q2A Q2AS	Q3A	Q4A QnASH	Q4AR	Q00J	Q00	Q01	Q02	QnH
COM ¹	Con rinfresco automatico della memoria condivisa della CPU	Campo di rinfresco = 2 k word (0.5 k word per ogni CPU)	—	—	—	—	—	—	720	660	
		Campo di rinfresco = 4 k word (1 k word per ogni CPU)	—	—	—	—	—	—	860	730	
	Senza rinfresco automatico della memoria condivisa della CPU		—	—	—	—	—	—	43	20	
FROM ¹	Letture dalla memoria condivisa di una diversa CPU	n3 = 0		—	—	—	—	—	—	59	29
		n3 = 1000								530	500
	Letture dal buffer di memoria di un modulo funzione speciale ²	n3 = 1	telaio base principale							51	24
			telaio di espansione							54	27
	n3 = 1000	telaio base principale	540							480	
		telaio di espansione	1100							1050	
S.TO	s2 = 1		—	—	—	—	—	—	74	33	
	s2 = 256		—	—	—	—	—	—	126	54	

¹ Se l'istruzione viene eseguita contemporaneamente da più CPU di un sistema multi CPU, il tempo di esecuzione aumenta. Per calcolare l'aumento del tempo di elaborazione dell'istruzione, usare la seguente formula.
 Per un sistema con solo telaio principale:
 Aumento del tempo di esecuzione [µs] = 0,54 x (numero di punti processati) x (numero di moduli CPU)

Per un sistema con telaio principale e telai di espansione:
 Aumento del tempo di esecuzione [µs] = 1.30 x (numero di punti processati) x (numero di moduli CPU)

² Il tempo di elaborazione per i moduli funzione speciali controllati dalla CPU che sta eseguendo l'istruzione è identico al tempo di elaborazione dell'istruzione per i moduli funzione speciali controllati da una diversa CPU del sistema multi CPU.

Istruzione	Elaborazione (operando)		Tempo di elaborazione (µs)							
			Moduli CPU serie QnA				Moduli CPU System Q			
			Q2A	Q3A	Q4A QnASH	Q4AR	Q00J	Q00	Q01	Q02
FROM (n1, n2, d, n3) FROMP (n1, n2, d, n3)	n3 = 1	253 ¹	217 ¹	160 ¹	160 ¹	—	—	—	—	—
		252 ²	210 ²	154 ²	154 ²	—	—	—	—	—
		—	—	—	—	125 ³	105 ³	93 ³	47 ⁴	22 ⁴
	n3 = 1000	4514 ¹	4286 ¹	4150 ¹	4150 ¹	—	—	—	—	—
		2855 ²	2127 ²	2038 ²	2038 ²	—	—	—	—	—
		—	—	—	—	740 ³	695 ³	685 ³	476 ⁴	437 ⁴
DFRO (n1, n2, d, n3) DFROP (n1, n2, d, n3)	n3 = 1	260 ¹	221 ¹	165 ¹	165 ¹	—	—	—	—	—
		257 ²	214 ²	156 ²	156 ²	—	—	—	—	—
		—	—	—	—	130 ³	110 ³	100 ³	51 ⁴	24 ⁴
	n3 = 500	4543 ¹	4271 ¹	4082 ¹	4082 ¹	—	—	—	—	—
		2883 ²	2129 ²	2064 ²	2064 ²	—	—	—	—	—
		—	—	—	—	745 ³	695 ³	675 ³	478 ⁴	437 ⁴
TO (n1, n2, d, n3) TOP (n1, n2, d, n3)	n3 = 1	276 ¹	217 ¹	162 ¹	162 ¹	—	—	—	—	—
		254 ²	211 ²	154 ²	154 ²	—	—	—	—	—
		—	—	—	—	120 ³	105 ³	92 ³	48 ⁴	20 ⁴
	n3 = 1000	4500 ¹	4319 ¹	4188 ¹	4188 ¹	—	—	—	—	—
		2878 ²	2155 ²	2043 ²	2043 ²	—	—	—	—	—
		—	—	—	—	735 ³	680 ³	645 ³	479 ⁴	412 ⁴
DTO (n1, n2, d, n3) DTOP (n1, n2, d, n3)	n3 = 1	260 ¹	221 ¹	165 ¹	165 ¹	—	—	—	—	—
		257 ²	216 ²	157 ²	157 ²	—	—	—	—	—
		—	—	—	—	130 ³	110 ³	99 ³	50 ⁴	23 ⁴
	n3 = 500	4471 ¹	4315 ¹	4198 ¹	4198 ¹	—	—	—	—	—
		2819 ²	2172 ²	2062 ²	2062 ²	—	—	—	—	—
		—	—	—	—	740 ³	680 ³	640 ³	457 ⁴	416 ⁴

¹ Questi tempi di elaborazione sono validi quando viene usato un telaio principale A38B/A1S38B con un telaio di espansione.

² Questi tempi di elaborazione sono validi per un telaio principale A38HB/A1S38HB.

³ Il tempo di elaborazione dell'istruzione dipende dal tipo di telaio di espansione utilizzato, dal numero di slot del telaio di base e dal numero di moduli effettivamente installati.

⁴ I tempi riportati sono relativi a un telaio base principale Q312B con il modulo QJ71C24 installato nello slot 0.

A.2.2 Tempi di elaborazione delle CPU serie A

Il tempo di elaborazione di una istruzione dipende dal modo di elaborazione selezionato per i segnali di ingresso e di uscita:

- Modalità comando diretto I/O = ○
- Modalità comando I/O con rinfresco = ○○

Istruzione	Elaborazione (Operandi)		Tempo di elaborazione (μs)								
			A	AnN, AnS		A2A, A2AS	A3A				
				○	○○			○			
				nessun X, Y	con X, Y						
LD	X		2.3	1.0	2.3	0.2	0.15				
LDI											
AND											
ANI	Y, M, L, B, F, T, C		1.3	1.0	1.0	0.2	0.15				
OR											
ORI											
ANB											
ORB			1.3	1.0	1.0	0.2	0.15				
MPS											
MRD											
MPP											
OUT	Y	non modificata	2.3	1.0	2.3	0.4	0.3				
		variato (OFF/ON)	2.3	1.0	2.3	0.4	0.3				
	M (tranne M speciali)	non modificata	1.3	1.0	1.0	0.4	0.3				
		variato (OFF/ON)	1.3	1.0	1.0	0.4	0.3				
	M speciali			37	37	0.8	0.6				
	F	non eseguita		66	AnN: 61 AnS: 62	61	6.6	5.0			
		eseguita	visualizzato	700	AnN: 633 AnS: 270	AnN: 633 AnS: 267	99	74			
	visualizzazione terminata										
	T	Tempo esecuzione istruzione		1.3	1.0	1.0	0.4	0.3			
		Elaborazione END	non eseguita		1.3	0	0	0.23	0.18		
			eseguita	dopo timeout		15	11	11	4.5	3.3	
				sommato	K	30	24	24	7.7	5.7	
			D		36	30	30	8.3	6.2		
		C	Tempo esecuzione istruzione		1.3	1.0	1.0	0.4	0.3		
			Elaborazione END	non eseguita		1.3	0	0	0.27	0.2	
				eseguita	senza conteggio		14	0	0	0.27	0.2
					dopo timeout		14	0	0	0.27	0.2
					sommato	K	28	25	25	4.2	3.1
	D			33		30	30	4.8	3.6		
	SET	Y	non eseguita		2.3	1.0	2.3	0.4	0.3		
eseguita			non modificata		2.3	1.0	2.3	0.4	0.3		
			modificato		2.3	1.0	2.3	0.4	0.3		
M, L, S, B		non eseguita		3.7	1.0	1.0	0.4	0.3			
		eseguita	non modificata		41	1.0	1.0	0.4	0.3		
			modificato		41	1.0	1.0	0.4	0.3		
M, B speciali		non eseguita			2.0	3.0	0.8	0.6			
		eseguita			32	32	0.8	0.6			
F		non eseguita		3.7	AnN: 3.0 AnS: 2.7	AnN: 3.0 AnS: 3.2	2.0	1.5			
		eseguita	visualizzato	730	AnN: 638 AnS: 232	AnN: 638 AnS: 237	99	74			
visualizzazione terminata											

Istruzione	Elaborazione (Operandi)		Tempo di elaborazione (µs)						
			A	AnN, AnS		A2A, A2AS	A3A		
				○	○○			○ nessun X, Y	○ con X, Y
RST	Y	non eseguita	2.3	1.0	2.3		0.4	0.3	
		eseguita	non modificata	2.3	1.0	2.3		0.4	0.3
			modificato (OFF/ON)	2.3	1.0	2.3		0.4	0.3
	M, L, S, B	non eseguita	3.7	1.0	1.0		0.4	0.3	
		eseguita	non modificata	41	1.0	1.0		0.4	0.3
			modificato (OFF/ON)	41	1.0	1.0		0.4	0.3
	Modulo M, B	non eseguita		3.0	3.0		0.8	0.6	
		eseguita		32	32		0.8	0.6	
	F	non eseguita	3.7	AnN: 3.0 AnS: 3.6	3.0		2.0	1.5	
		eseguita	visualizzato	680	AnN: 477 AnS: 296	AnN: 477 AnS: 283		150	112
			visualizzazione terminata						
	T, C	non eseguita	3.7	3.0	3.0		1.4	1.1	
		eseguita	57	43	43		5.6	4.2	
	D, W, A0, A1, V, Z	non eseguita	3.7	3.0	3.0		1.4	1.1	
eseguita		34	28	28		8.4	6.3		
R	non eseguita	3.7	3.0	3.0		1.4	1.1		
	eseguita	41	35	35		4.6	3.5		
PLS PLF	Y	non eseguita	65	59	61		2.2	1.7	
		eseguita	ON	68	62	63		2.2	1.7
			OFF	64	60	62		2.2	1.7
	M, L, B, F	non eseguita	64	59	59		2.2	1.7	
		eseguita	ON	67	62	62		2.2	1.7
			OFF	63	61	61		2.2	1.7
SFT SFTP	Y	non eseguita	3.7	3.0	3.0		1.4	1.1	
		eseguita	49	38	39		4.4	3.3	
	M, L, B, F	non eseguita	3.7	3.0	3.0		1.4	1.1	
		eseguita	48	38	38		4.4	3.3	
MC	Y	non eseguita	85	43	44		1.2	0.9	
		eseguita	50	39	41		1.2	0.9	
	M, L, S, B, F	non eseguita	84	43	43		1.2	0.9	
		eseguita	49	39	39		1.2	0.9	
MCR		35	26	26		0.6	0.45		
FEND	M9084 = OFF	2400	2150	2150		435	327		
END	M9084 = ON	2400	2060	2060		285	214		
NOP		1.3	1.0	1.0		0.2	0.15		
LD=		95	70	70	87	3.8	2.9		
AND=		96	61	62	81	2.6	2.0		
OR=		94	67	66	85	2.8	2.1		
LD<>		98	69	69	86	4.1	3.1		
AND<>		92	60	60	79	2.6	2.0		
OR<>		96	66	66	84	2.8	2.1		
LD>		96	67	67	84	4.1	3.1		
AND>		92	60	60	79	2.6	2.0		
OR>		98	66	65	83	2.8	2.1		
LD<=		100	71	71	88	4.1	3.1		
AND<=		94	61	61	80	2.6	2.0		
OR<=		100	69	68	86	2.8	2.1		
LD<		96	69	69	86	4.1	3.1		
AND<		92	59	60	79	2.6	2.0		
OR<		96	66	65	84	2.8	2.1		
LD>=		100	71	71	88	4.1	3.1		

Istruzione	Elaborazione (Operandi)	Tempo di elaborazione (μs)					
		A	AnN, AnS		A2A, A2AS	A3A	
		○	○○	○ nessun X, Y	○ con X, Y	○○	○○
AND>=		94	61	61	81	2.6	2.0
OR>=		100	69	68	86	2.8	2.1
LDD=		238	133	134	119	10	7.7
ANDD=		231	124	125	210	5.9	4.4
ORD=		236	133	133	218	6.3	4.7
LDD<>		235	131	132	217	10	7.7
ANDD<>		239	129	129	215	5.9	4.4
ORD<>		234	129	129	214	6.1	4.6
LDD>		238	133	133	219	9.7	7.3
ANDD>		240	131	131	217	5.8	4.4
ORD>		236	131	130	219	6.0	4.5
LDD<=		244	137	136	222	9.7	7.3
ANDD<=		238	127	128	213	5.8	4.4
ORD<=		246	137	136	221	6.0	4.5
LDD<		238	133	133	219	9.7	7.3
ANDD<		241	131	131	217	5.8	4.4
ORD<		236	131	130	215	6.0	4.5
LDD>=		243	137	137	222	9.7	7.3
ANDD>=		238	127	128	213	5.8	4.4
ORD>=		246	137	136	221	6.0	4.5
+ (s,d) +P (s,d)		72	44	45	59	2.8	2.1
+ (s1,s2,d) +P (s1,s2,d)		112	77	77	103	3.2	2.4
-(s,d) -P (s,d)		74	45	45	59	2.8	2.1
-(s1,s2,d) -P (s1,s2,d)		123	79	79	107	3.2	2.4
D+ (s,d) D+P (s,d)		110	69	69	90	4.0	3.0
D+ (s1,s2,d) D+P (s1,s2,d)		140	99	99	246	4.6	3.5
D-(s,d) D-P (s,d)		110	69	69	90	4.0	3.0
D-(s1,s2,d) D-P (s1,s2,d)		141	99	99	130	4.6	3.5
x (s1,s2,d) xP (s1,s2,d)		135	94	95	168	3.4	2.6
/ (s1,s2,d) /P (s1,s2,d)		144	102	103	99	11	8.6
Dx (s1,s2,d) DxP (s1,s2,d)		429	341	340	370	20	15
D/ (s1,s2,d) D/P (s1,s2,d)		289	393	394	412	36	27

Istruzione	Elaborazione (Operandi)	Tempo di elaborazione (µs)					
		A	AnN, AnS		A2A, A2AS	A3A	
			○	○○			○
				nessun X, Y	con X, Y		
B+ (s,d) B+P (s,d)		210	123	123	183	6.4	4.8
B+ (s1,s2,d) B+P (s1,s2,d)		217	129	129	192	6.2	4.7
B- (s,d) B-P (s,d)		210	125	125	185	34	25
B- (s1,s2,d) B-P (s1,s2,d)		212	133	133	203	32	23
DB+ (s,d) DB+P (s,d)		320	175	176	280	14	11
DB+ (s1,s2,d) DB+P (s1,s2,d)		321	187	186	294	14	11
DB- (s,d) DB-P (s,d)		318	175	175	280	31	23
DB- (s1,s2,d) DB-P (s1,s2,d)		322	185	186	294	29	22
Bx (s1, s2, d) BxP (s1, s2, d)		410	299	300	358	14	11
B/ (s1, s2, d) B/P (s1, s2, d)		422	235	236	274	89	67
DBx (s1, s2, d) DBxP (s1, s2, d)		1158	941	939	1044	11	8.0
DB/ (s1, s2, d) DB/P (s1,s2,d)		998	896	894	954	62	47
INC INCP		46	29	29	38	2.0	1.5
DINC DINCP		66	42	42	132	2.4	1.8
DEC DECP		48	31	31	39	2.0	1.5
DDEC DDECP		66	42	42	54	2.4	1.8
BCD BCDP		110	82	83	90	3.0	2.3
DBCD DBC DP		329	219	220	284	13	9.5
		329	219	220	284	13	9.5
BIN BINP		104	79	78	86	3.0	2.3
DBIN DBINP		311	215	216	280	6.0	4.5
NEG NEGP		105	50	49	86	8.6	6.5

Istruzione	Elaborazione (Operandi)	Tempo di elaborazione (µs)					
		A	AnN, AnS		A2A, A2AS	A3A	
			○	○○			○
				nessun X, Y	con X, Y		
MOV MOV P		72	47	47	57	17	13
DMOV DMOV P		104	67	67	87	20	15
CML CML P		68	43	43	57	2.4	1.8
DCML DCML P		130	74	75	108	3.2	2.4
BMOV (s1, s2, n) BMOV P (s1, s2, n)		7498	399	400	7144	1444	1083
FMOV (s1, s2, n) FMOV P (s1, s2, n)		1118	229	228	1029	1427	1070
XCH XCH P		102	60	61	84	2.8	2.1
DXCH DXCH P		170	107	107	141	4.2	3.2
CJ	senza indicizzazione	49	39	39		6.6	5.0
	con indicizzazione		48	48		6.6	5.0
SCJ	senza indicizzazione	54	71	71		6.6	5.0
	con indicizzazione		81	81		6.6	5.0
JMP		50	39	39		6.6	5.0
EI		195	38	38		3.0	2.3
DI		46	66	66		3.2	2.4
IRET		249	120	120		3.4	2.6
WAND (s, d) WAND P (s, d)		90	60	59	72	2.8	2.1
WAND (s1, s2, d) WAND P (s1, s2, d)		179	96	96	152	7.6	5.7
DAND (s, d) DAND P (s, d)		276	140	139	240	13	9.5
DAND (s1, s2, d) DAND P (s1, s2, d)							
WOR (s, d) WOR P (s, d)		90	61	60	72	2.8	2.1
WOR (s1, s2, d) WOR P (s1, s2, d)		176	97	96	152	7.6	5.7
DOR (s, d) DOR P (s, d)		276	140	139	240	13	9.5
DOR (s1, s2, d) DOR P (s1, s2, d)							

Istruzione	Elaborazione (Operandi)	Tempo di elaborazione (µs)					
		A	AnN, AnS		A2A, A2AS	A3A	
			nessun X, Y	con X, Y			
WXOR (s, d) WXORP (s, d)		91	60	59	72	2.8	2.1
WXOR (s1, s2, d) WXORP (s1, s2, d)		178	97	96	152	7.6	5.7
DXOR (s, d) DXORP (s, d)		274	140	139	240	13	9.5
DXOR (s1, s2, d) DXORP (s1, s2, d)							
WXNR (s, d) WXNRP (s, d)		89	64	62	74	3.0	2.3
WXNR (s1, s2, d) WXNRP (s1, s2, d)		177	98	96	152	7.8	5.9
DXNR (s, d) DXNRP (s, d)		277	142	140	241	15	11
DXNR (s1, s2, d) DXNRP (s1, s2, d)							
ROR (n) RORP (n)	n = 3/5 (n = 5 per tutte le CPU AnN, n = 3 per tutte le CPU esclusa AnN)	66	52	51	51	5.8	4.4
RCR (n) RCRP (n)	n = 3/5 (n = 5 per tutte le CPU AnN, n = 3 per tutte le CPU esclusa AnN)	74	59	59	59	6.4	4.8
ROL (n) ROLP (n)	n = 3/5 (n = 5 per tutte le CPU AnN, n = 3 per tutte le CPU esclusa AnN)	68	54	53	53	5.8	4.4
RCL (n) RCLP (n)	n = 3/5 (n = 5 per tutte le CPU AnN, n = 3 per tutte le CPU esclusa AnN)	74	57	57	57	6.4	4.8
DROR (n) DRORP (n)	n = 3/5 (n = 5 per tutte le CPU AnN, n = 3 per tutte le CPU esclusa AnN)	97	70	70	69	11	8.3
DRCR (n) DRCRP (n)	n = 3/5 (n = 5 per tutte le CPU AnN, n = 3 per tutte le CPU esclusa AnN)	95	72	72	72	12	9.2
DROL (n) DROLP (n)	n = 3/5 (n = 5 per tutte le CPU AnN, n = 3 per tutte le CPU esclusa AnN)	101	70	69	69	10	7.8
DRCL (n) DRCLP (n)	n = 3/5 (n = 5 per tutte le CPU AnN, n = 3 per tutte le CPU esclusa AnN)	98	68	68	68	12	8.7

Istruzione	Elaborazione (Operandi)	Tempo di elaborazione (µs)					
		A	AnN, AnS		A2A, A2AS	A3A	
			nessun X, Y	con X, Y			
SFR (s, n) SFRP (s, n)	n = 3/5 (n = 5 per tutte le CPU AnN, n = 3 per tutte le CPU esclusa AnN)	102	74	72	83	5.0	3.8
SFL (d, n) SFLP (d, n)	n = 5	106	74	73	84	4.8	3.6
BSFR (d, n) BSFRP (d, n)	n = 5	145	124	123	124	29	22
BSFL (d, n) BSFLP (d, n)	n = 5	158	134	133	134	28	21
DSFR (d, n) DSFRP (d, n)	n = 5	133	118	116	—	18.8	14.1
DSFL (d, n) DSFLP (d, n)	n = 5	134	118	17	—	22	17
BSET (d, n) BSETP (d, n)	n = 5	107	90	90	—	9.6	7.2
BRST (d, n) BRSTP (d, n)	n = 5	114	97	96	—	9.6	7.2
SER (s1, s2, d, n) SERP (s1, s2, d, n)	n = 5	230	200	200	—	33	25
SUM SUMP		164	115	114	131	15	11
DSUM DSUMP		267	200	199	231	34	25
DECO (s, d, n) DECOP (s, d, n)	n = 2	249	164	163	216	32	24
ENCO (s, d, n) ENCOP (s, d, n)	n = 2	478	164	163	195	41	31
SEG		170	91	91	155	6.4	4.8
DIS (s, d, n)	n = 4	180	154	153	??	25	19
DISP (s, d, n)	n = 4	180	154	153	120	25	19
UNI (s, d, n)	n = 4	159	131	131	—	31	24
UNIP (s, d, n)	n = 4	159	131	131	—	31	24
FOR		64	53	53		5.8	4.4
NEXT		2532	41	41		8.0	6.0
CALL (pn)	senza indicizzazione	74	74	74		10	7.8
	con indicizzazione		78	78		10	7.8
CALLP (pn)	senza indicizzazione	74	70	70		10	7.8
	con indicizzazione		78	78		10	7.8
RET		249	50	50		7.0	5.3

Istruzione	Elaborazione (Operandi)	Tempo di elaborazione (μs)					
		A	AnN, AnS		A2A, A2AS	A3A	
			○	○○			○ nessun X, Y
CHG	M9084 = OFF	8546	2420	2420		—	—
	M9084 = ON	—	2340	2340		—	—
SUB SUBP	senza indicizzazione	90	79	79		—	—
	con indicizzazione	—	85	85		—	—
FIFW FIFWP		340	101	101	123	20	15
FIFR FIFRP		202	118	118	134	69	52
PR		—	226	226	226	74	56
PRC		—	141	141	141	37	28
LED		170	203	203	203	100	75
LEDC		210	265	265	265	142	106
LEDR		520	638	638	638	106	80
LEDA		170	202	202	202	—	—
LEDB		172	211	211	211	—	—
CHK (controllo errori)	1 contatto d'ingresso	—	—	AnN: 771 AnS: 240		33	25
	50 contatti d'ingresso	—	—	AnN: 3380 AnS: 3905		1257	943
	100 contatti d'ingresso	—	—	AnN: 6687 AnS: 7820		2503	1877
	150 contatti d'ingresso	—	—	AnA: 10137 AnS: 11470		3753	2815
CHK (generazione flip-flop)		—	121	121	121	—	—
SLT	memoria operandi	—	8448	8448	8448	2915	2186
	memoria operandi + R	—	24598	24598	24598	9996	7497
SLTR		—	29	29	29	6.6	5.0
STRA		—	30	30	30	5.0	3.8
STRAR		—	28	28	28	5.0	3.8
STC		—	28	28	28	2.4	1.8
CLC		—	31	31	31	2.4	1.8
ASC		140	120	120	120	3.4	2.6
WDT WDTP		—	64	64	64	5.0	3.8
DUTY		—	68	68	68	14	11
LRDP (n1, s, d, n2)	n2 = 1	—	190	190	190	42	32
	n2 = 32	—	190	190	190	42	32
LWTP (n1, d, s, n2)	n2 = 1	—	200	200	200	49	37
	n2 = 32	—	446	446	446	89	66
RFRP (n1, n2, d, n3)	n3 = 1	—	172	172	172	32	24
	n3 = 32	—	172	172	172	63	47
RTOP (n1, n2, s, n3)	n3 = 1	—	176	176	176	68	51
	n3 = 32	—	176	176	176	34	26

Istruzione	Elaborazione (Operandi)	Tempi di elaborazione (μs)						
		A	AnN, AnS		A2A, A2AS, A2U		A3A, A3U	
		○	○ + ○		○		○	
			nessun X, Y	con X, Y	nessun X, Y	con X, Y	nessun X, Y	con X, Y
FROM (n1, n2, d, n3)	n3 = 1	—	439	524	237	261	178	196
FROMP (n1, n2, d, n3)	n3 = 1000/112 ¹	—	6609	2358	5749	2789	4312	2092
DFRO (n1, n2, d, n3)	n3 = 1	—	449	529	244	266	183	199
DFROP (n1, n2, d, n3)	n3 = 500/56 ²	—	6609	2109	5669	1669	4252	1252
TO (n1, n2, d, n3)	n3 = 1	—	449	539	243	266	182	200
TOP (n1, n2, d, n3)	n3 = 1000/112 ¹	—	6609	3918	5773	2117	4330	1588
DTO (n1, n2, d, n3)	n3 = 1	—	454	544	240	266	180	199
DTOP (n1, n2, d, n3)	n3 = 500/56 ²	—	6609	1609	5747	1501	4310	1126

¹ CPU senza X e Y: n3 = 1000; altre CPU con X e Y: n3 = 112

² CPU senza X e Y: n3 = 500; altre CPU con X e Y: n3 = 56

A.1 Confronto fra CPU

La tabella seguente contiene le caratteristiche, cioè operandi disponibili, modi di funzionamento, relé speciali ecc. delle diverse CPU (System Q, Q4AR, QnA, AnU, AnA, AnN, AnS).

A.2.3 Operandi disponibili

Operando	System Q		Serie Q		Serie A			
	Q00J, Q00, Q01	Qn, QnH, QnPH	Q4AR	QnA	AnU	AnA	AnN	AnS
Numero ingressi/uscite	Q00J: 256 Q00: 1024 Q01: 1024	4096 per Q02, Q02H, Q06H, Q12H e Q25H	4096	Q2A: 512 Q2A-S1: 1024 Q3A: 2048 Q4A: 4096	A2U: 512 A2U-S1: 1024 A3U: 2048 A4U: 4096	A2A: 512 A2A-S1: 1024 A3A: 2048?	A1N: 256 A2N: 512 A2N-S: 1024 A3N: 2048?	A1S: 256 A1S-S1: 512 A2S: 512 A2S-S1: 1024
Relé interni	8192 ¹		8192	8192 ¹				256
Relé retentivi	2048 ¹	8192 ¹	8192	8192 ¹	Totale 8192	Totale 8192	Totale 2048	1048 ¹
Relé di passo	Programma di sequenza	—	8192	—				0 ¹
	SFC	2048	8192	8192	—	—	—	
Spie di errore	1024 ¹	2048 ¹	2048	2048 ¹	2048	2048	256	256
Relé impulsivi	1024 ¹	2048 ¹	2048	2048 ¹	—	—	—	—
Relé di comunicazione	2048 ¹	8192 ¹	8192	8192 ¹	8192	4096	1024	1024
Relé speciali di comunicazione	1024	2048 ¹	2048	2048 ¹	56	56	56	—
Timer	512 ¹	2048 ¹	2048	2048 ¹				256 ¹
Timer retentivi	0 ¹	0 ¹	0 ¹	0 ¹	Totale 2048	Totale 2048	Totale 256	0 ¹
Contatori	512 ¹	1024 ¹	1024	1024 ¹	1024	1024	256	256 ¹
Registri dati	11136 ¹	12288 ¹	12288	12288 ¹	8192	6144	1024	1024
Registri di comunicazione	2048 ¹	8192 ¹	8192	8192 ¹	8192	4096	1024	1024
Registri speciali di comunicazione	1024 ¹	2048 ¹	2048	2048 ¹	56	56	56	—
Ingressi funzione	16 (da FX0 a FXF)	16 (da FX0 a FXF)	16 (da FX0 a FXF)	16 (da FX0 a FXF)	—	—	—	—
Uscite funzione	16 (da FX0 a FXF)	16 (da FY0 a FYF)	16 (da FY0 a FYF)	16 (da FY0 a FYF)	—	—	—	—
Relé speciali	1024	2048	2048	2048	256	256	256	256
Registri funzione	5 (da FD0 a FD4)	16 (da FD0 a FD15)	16 (da FD0 a FD15)	16 (da FD0 a FD15)	—	—	—	—
Registri speciali	1024	2048	2048	2048	256	256	256	256
Operandi di comunicazione ad accesso diretto	Indicati da J□□		Indicati da J□□		—	—	—	—
Operandi speciali ad accesso diretto	Indicati da U□□G□		Indicati da U□□G□□		—	—	—	—

Operando		System Q		Serie Q		Serie A			
		Q00J, Q00, Q01	Qn, QnH, QnPH	Q4AR	QnA	AnU	AnA	AnN	AnS
Registri indice	Z	10 (da Z0 a Z15)	16 (da Z0 a Z15)	16 (da Z0 a Z15)	16 (da Z0 a Z15)	7 (Z, da Z1 a Z6)	7 (Z, da Z1 a Z6)	1 (Z)	1 (Z)
	V ²	—		—		7 (V, da V1 a V6)	7 (V, da V1 a V6)	1 (V)	1 (V)
File registri		Q00JCPU: 0 Q00 e Q01CPU: 32767	32767 per blocco (da R0 a R32767) 1042432 (da ZR0 a ZR1042432)	32767 per blocco (da R0 a R32767) 1042432 (da ZR0 a ZR1042432)	32767 per blocco (da R0 a R32767) 1042432 (da ZR0 a ZR1042432)	8192 per blocco (da R0 a R8191)	8192 per blocco (da R0 a R8191)	8192 per blocco (da R0 a R8191)	0 ¹
Accumulatori ³		—		—		2	2	2	2
Annidamento		15	15	15	15	8	8	8	8
Puntatore		300	4096	4096	4096	256	256	256	256
Puntatori di interruzione		128	256	48	48	32	32	32	32
Blocchi SFC		—	320	320	320	—	—	—	—
Operandi impulsivi SFC		—	512	512	512	—	—	—	—
Costanti decimali		da K-2147483648 a K2147483647		da K-2147483648 a K2147483647		da K-2147483648 a K2147483647			
Costanti esadecimali		da H0 a HFFFFFFF		da H0 a HFFFFFFF		da H0 a HFFFFFFF			
Costanti numero reale		—	da Ea1,17549-38 a Ea3,40282+38	da Ea1,17549-38 a Ea3,40282+38		—	—	—	—
Stringhe di caratteri		„CPU QnA“, „ABCD“ ⁴	„CPU QnA“, „ABCD“	„CPU QnA“, „ABCD“		—	—	—	—

¹ Il numero di operandi può essere modificato tramite parametrizzazione.

² Le CPU QnA usano V come relé impulsivi.

³ Le istruzioni che usano gli accumulatori con le CPU AnN, AnA e AnU hanno un formato diverso rispetto alle CPU QnA.

⁴ Può essere usato solo dalla istruzione \$MOV con Q00JCPU, Q00CPU e Q01CPU.

A.2.4 Modi di controllo I/O

Modo di controllo I/O		Tipo di CPU				
		System Q	QnA/ QnAR	AnU	AnA	AnN
Modo rinfresco		●		●	●	● ²
Modo ingresso/ uscita diretto	Istruzioni per rinfresco parziale	●		●	●	●
	Istruzioni dedicate ¹	—		●	●	—
	Accesso diretto ingressi	●		—	—	—
	Accesso diretto uscite	●		—	—	—
Modo diretto		—		—	—	● ²

¹ Le istruzioni DOUT, DSET e SRTS sono istruzioni dedicate per l'accesso diretto alle uscite. Non esistono istruzioni dedicate per l'accesso diretto agli ingressi..

² Nelle CPU AnN, il modo rinfresco o il modo diretto viene selezionato con un DIP switch.

A.2.5 Tipi di dati

Dati impostati		CPU System Q	CPU QnA/ QnAR	CPU AnU	CPU AnA	CPU AnN
Dati a bit	Operando a bit	●		●	●	●
	Operando a word	● (necessaria specifica del bit)		—	—	—
16-bit dati a word	Operando a bit	● (necessaria specifica della cifra)		● (necessaria specifica della cifra)	● (necessaria specifica della cifra)	● (necessaria specifica della cifra)
	Operando a word	●		●	●	●
32-bit dati a word	Operando a bit	● (necessaria specifica della cifra)		● (necessaria specifica della cifra)	● (necessaria specifica della cifra)	● (necessaria specifica della cifra)
	Word Operando	●		●	●	●
Dato in numero reale		● ¹		●	●	—
Dati stringa di caratteri		● ²		—	—	—

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

² Le stringhe di caratteri possono essere usate su Q00JCPU, Q00CPU e Q01CPU solo assieme all'istruzione \$MOV.

NOTA

Fare riferimento alla sezione 3.5 per informazioni dettagliate sui tipi di dati.

A.2.6 Confronto timer

Funzionitimer

Nome	Funzione			System Q CPU QnA/ QnAR	CPU AnU*	CPU AnA*	CPU AnN/AnS*
Timer lenti	Unità di misura			100 ms (default) Campo impostazione: da 10 a 100 ms Questo valore è il fattore moltiplicativo del valore di comando (TV).	Fisso a 100 ms		
	Programmazione (GX IEC Developer)	TIMER_M (timer normali/ dedicati)	Specifica valore impostato e avviamento timer	●	●	●	●
		TIMER_VALUE_M (solo timer dedicati)	Indicazione valore impostato	●	●	●	●
		TIMER_START_M (solo timer dedicati)	Avviamento timer	●	●	●	●
Programmazione (GX Developer)	Specifica valore impostato e avviamento timer		OUT Tn impostazione valore	OUT Tn impostazione valore			
Timer veloci	Unità di misura			10 ms (default) Campo impostazione: da 10 a 100 ms Questo valore è il fattore moltiplicativo del valore di comando (TV).	Fisso a 100 ms		
	Programmazione (GX IEC Developer)	TIMER_M (timer normali/ dedicati)	Specifica valore impostato e avviamento timer	●	●	●	●
		TIMER_VALUE_M (solo timer dedicati)	Indicazione valore impostato	●	●	●	●
		TIMER_START_M (solo timer dedicati)	Avviamento timer	●	●	●	●
Programmazione (GX Developer)	Specifica valore impostato e avviamento timer		OUT Tn impostazione valore	OUT Tn impostazione valore			
Timer lenti retentivi	Unità di misura			100 ms (default) Campo impostazione: da 10 a 100 ms Questo valore è il fattore moltiplicativo del valore di comando (TV).	Fisso a 100 ms		
	Programmazione (GX IEC Developer)	TIMER_H_M (timer normali/ dedicati)	Specifica valore impostato e avviamento timer	●	●	●	●
		TIMER_VALUE_M (solo timer dedicati)	Indicazione valore impostato	●	●	●	●
		TIMER_START_M (solo timer dedicati)	Avviamento timer	●	●	●	●
Programmazione (GX Developer)	Specifica valore impostato e avviamento timer		OUT Tn impostazione valore	OUT Tn impostazione valore			

Nome	Funzione		System Q CPU QnA/ QnAR	CPU AnU*	CPU AnA*	CPU AnN/AnS*	
Timer veloci retentivi	Unità di misura		10 ms (default) Campo impostazione: da 10 a 100 ms Questo valore è il fattore moltiplicativo del valore di comando (TV).	—			
	Programmazione (GX IEC Developer)	TIMER_H_M (timer normali/ dedicati)	Specifica valore impostato e avviamento timer	●	—	—	—
		TIMER_VALUE_M (solo timer dedicati)	Indicazione valore impostato	●	—	—	—
		TIMER_START_M (solo timer dedicati)	Avviamento timer	●	—	—	—
Programmazione (GX Developer)	Specifica valore impostato e avviamento timer		OUTH STn Impostazione valore	—	—	—	
Campo per impostazione valore			da 1 a 32767	da 1 a 32767			
Elaborazione valore di impostazione 0			ON transitorio	Nessun massimo (senza timeout)			
Indicizzazione	Contatto		Abilitata (usare solo Z0 e Z1)	Possibile		Non possibile	
	Bobina		Abilitata (usare solo Z0 e Z1)	Non possibile		Non possibile	
	Valore impostato		Non possibile	Non possibile		Non possibile	
	Valore attuale		Abilitato (da Z0 a Z15 ammessi, da Z0 a Z9 per Q00JCPU, Q00CPU e Q01CPU)	Possibile		Possibile	
Aggiornamento valore attuale			Con esecuzione istruzione OUT Tn	Dopo elaborazione END			
Elaborazione contatto ON/OFF							

* L'indirizzo iniziale dei diversi timer deve essere specificato in GX IEC Developer nella finestra di dialogo "PLC Parameter - T/C Range"

Nome	Blocco funzionale	Tipo di CPU					
		System Q	QnA/QnAR	AnU	AnA	AnN	AnS
Timer da 10 ms	Instance TIMER_10_FB_M - Coil ValueOut - Preset Status - ValueIn	—	●	●	●	●	●
Timer da 100 ms	Instance TIMER_100_FB_M - Coil ValueOut - Preset Status - ValueIn	—	●	●	●	●	●
Timer retentivo	Instance TIMER_CONT_FB_M - Coil ValueOut - Preset Status - ValueIn	—	●	●	●	●	●
Timer lenti	Instance TIMER_LOW_FB_M - Coil ValueOut - Preset Status - ValueIn	—	●	—	—	—	—
Timer veloci	Instance TIMER_HIGH_FB_M - Coil ValueOut - Preset Status - ValueIn	—	●	—	—	—	—
Timer retentivo veloce	Instance TIMER_CONTHFB_M - Coil ValueOut - Preset Status - ValueIn	—	●	—	—	—	—

Termine nel blocco funzionale	Significato		Indicazione di timer normali	Indicazione di timer retentivi
Coil	Bobina	Condizione di esecuzione per timer	TC	STC
Preset	Valore impostato	—	TValue	TValue
ValueIn	Valore iniziale	Default: 0	—	—
ValueOut	Valore attuale	—	TN	STN
Status	Contatto	Il contatto di uscita commuta allo scadere del tempo impostato	TS	STS

Assegna il blocco funzionale alla etichetta di istanza nella intestazione e assegna le variabili di ingresso e uscita.

NOTA Avvertenze per l'uso dei timer

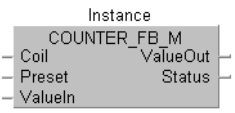
Durante l'esecuzione dell'istruzione OUT(H) T, il valore attuale del timer viene aggiornato e il contatto viene commutato ON o OFF. Se il valore attuale del timer è maggiore o uguale al valore impostato quando la bobina del timer è stata attivata, il contatto del timer viene attivato (ON).

In un programma in cui un timer viene avviato da un altro timer, l'istruzione relativa al timer al timer avviato per ultimo deve essere processata per prima. Ad esempio, se il contatto di T1 attiva la bobina di T2, l'istruzione relativa a T2 deve essere posta nel programma prima dell'istruzione per T1.

In questo modo si evita che entrambi i contatti dei timer si attivino nella stessa scansione. Questo si può verificare se l'istruzione per un timer che viene avviato da un altro timer viene elaborata per prima e il valore impostato per i timer veloci è minore del tempo di scansione, oppure se il valore impostato per i timer lenti è „1“.

A.2.7 Confronto dei contatori

Funzione			Tipo di CPU					
			System Q	QnA/QnAR	AnU	AnA	AnN	AnS
Programmazione (GX IEC Developer)	Counter_M	Valore impostato specifica e avviamento contatore	●		●	●	●	●
	Counter_Start_M	Valore impostato specifica	●		●	●	●	●
	Counter_Value_M	Avviamento contatore	●		●	●	●	●
Programmazione (GX Developer)	OUT Cn Impostazione valore	Valore impostato specifica e avviamento contatore	●		●	●	●	●
Indicizzazione	Contatto		Abilitata (usare solo Z0 e Z1)		Possibile		Non possibile	
	Coil		Abilitata (usare solo Z0 e Z1)		Possibile		Non possibile	
	Valore impostato		Non possibile		Non possibile		Non possibile	
	Valore attuale		Abilitato (da Z0 a Z15 ammessi, da Z0 a Z9 per Q00JCPU, Q00CPU e Q01CPU)		Possibile		Possibile	
Aggiornamento valore attuale			Con esecuzione istruzione OUT Tn		Dopo elaborazione END			
Elaborazione contatto ON/OFF								

Nome	Blocchi funzionali	Tipo di CPU					
		System Q	QnA/QnAR	AnU	AnA	AnN	AnS
Contatore		—	●	●	●	●	●

Termine nel blocco funzionale	Significato		Sigla di contatore
Coil	Bobina	Condizione di esecuzione contatore	CC
Preset	Valore impostato		CValue
ValueIn	Valore iniziale	Default: 0	—
ValueOut	Valore attuale		CN
Status	Contatto	Il contatto di uscita viene commutato al termine dell'elaborazione del blocco funzionale.	CS

A.2.8 Confronto istruzioni di visualizzazione

Istruzione	CPU System Q	CPU QnA/QnAR	CPU AnU	CPU AnA	CPU AnN	CPU AnS
PR ¹	Se SM701 è OFF: invio fino a quando 00 _H viene incontrato		Se M9049 è OFF: invio fino a quando 00 _H viene incontrato			
	Se SM701 è ON: emissione di 16 caratteri		Se M9049 è OFF: emissione di 16 caratteri			
PRC ¹	Se SM701 è OFF: emissione commento di 32 caratteri		emissione commento di 16 caratteri			
	Se SM701 è ON: emissione dei primi 16 caratteri					

¹ Queste istruzioni non sono disponibili su Q00JCPU, Q00CPU o Q01CPU.

A.2.9 Istruzioni serie Q e System Q equivalenti alle istruzioni per serie A

Dato che le CPU System Q e serie Q non usano accumulatori (A0, A1), il formato delle istruzioni per le CPU AnU, AnN e AnA che usano gli accumulatori è stato modificato.

Funzione	CPU System Q / QnA		CPU AnU / AnA / AnN	
	Formato istruzione	Nota	Formato istruzione	Nota
Rotazione 16 bit verso destra	ROR (d, n)	D: Dato da ruotare	ROR (n)	Il dato ruotato è inserito in A0
	RCR (d, n)	D: Dato da ruotare Il flag di riporto usa SM700	RRCR (n)	Il dato ruotato è inserito in A0 Segnale di riporto usa M9012
Rotazione 16 bit verso sinistra	ROL (d, n)	D: Dato da ruotare	ROL (n)	Il dato ruotato è inserito in A0
	RCL (d, n)	D: Dato da ruotare Il flag di riporto usa SM700	RCL (n)	Il dato ruotato è inserito in A0 Segnale di riporto usa M9012
Rotazione 32 bit verso destra	DROR (d, n)	D: Dato da ruotare	DROR (n)	Il dato ruotato è inserito in A0 e A1
	DRCR (d, n)	D: Dato da ruotare Il flag di riporto usa SM700	DRCCR (n)	Il dato ruotato è inserito in A0 e A1 Segnale di riporto usa M9012
Rotazione 32 bit verso sinistra	DROL (d, n)	D: Dato da ruotare	DROL (n)	Il dato ruotato è inserito in A0 e A1
	DRCL (d, n)	D: Dato da ruotare Il flag di riporto usa SM700	DRCL (n)	Il dato ruotato è inserito in A0 e A1 Segnale di riporto usa M9012
Ricerca dato 16-bit	SER (s1, s2, d, n)	Il risultato della ricerca viene memorizzato negli operandi D e D+1	SER (s1, s2, n)	Risultato della ricerca memorizzato in A0 e A1
Ricerca dato 32-bit	DSER (s1, s2, d, n)	Il risultato della ricerca viene memorizzato negli operandi D e D+1	DSER (s1, s2, n)	Risultato della ricerca memorizzato in A0 e A1 Segnale di riporto usa M9012
Controllo dati 16 bit	SUM (s, d)	Il risultato del controllo viene memorizzato nell'operando D	SUM (s)	Il risultato del controllo è memorizzato in A0
Controllo dati 32 bit	DSUM (s, d)	Il risultato del controllo viene memorizzato nell'operando D	DSUM (s)	Il risultato del controllo è memorizzato in A0
Rinfresco parziale	RFS (s, n)	Aggiunta istruzione dedicata	SEG (d, n)	Solo se M9052 è ON
Conversione 8 caratteri ASCII	\$MOV (s, d)		ASC (d)	
Segnale riporto attivo	SET (SM700)	Nessuna istruzione dedicata	STC	
Azzeramento segnale di riporto	RST (SM700)	Nessuna istruzione dedicata	CLC	
Salto a istruzione END	GOEND	Aggiunta istruzione dedicata	CJ (P255)	P255: richiamo istruzione END
Istruzione CHK	CHKST CHK	Aggiunta istruzione CHK	CJ (Pn) CHK (P255)	

¹ Non disponibile su Q00JCPU, Q00CPU e Q01CPU

A.2.10 Confronto fra CPU QnA/Q2AS e CPU MELSEC System Q

Le nuove istruzioni che seguono sono utilizzabili solo su CPU System Q.

Funzione	Istruzione
Lettura informazioni dei moduli	UNIRD
Insieme di tracce	TRACE
Reset traccia	TRACER
Scrittura dati nel file designato	S.FWRITE
Lettura dati dal file designato	S.FREAD
Caricamento di un programma dalla memoria	PLOAD
Scaricamento (cancellazione) di un programma dalla memoria programmi	PUNLOAD
Scaricamento (cancellazione) di un programma dalla memoria programmi e caricamento di un altro programma	PSWAP
Trasferimento blocchi o file registri ad alta velocità	PBMOV
Scrittura nella memoria condivisa della CPU	S.TO

La tabella seguente elenca le istruzioni QnA/Q2AS non utilizzabili sulle CPU System Q.

Funzione	Istruzione
Operazione di scrittura blocco su file registri EEPROM	EROMWR
Impostazione traccia di campionamento (può essere sostituita da TRACE)	STRA
Annullamento traccia di campionamento (può essere sostituita da TRACER)	STRAR
Attivazione stato retentivo	SLT
Azzeramento stato retentivo	SLTR
Impostazione traccia programma	PTRA
Annullamento traccia programma	PTRAR
Esecuzione traccia programma	PTRAEXE PTRAEXEP
Invio caratteri ASCII su display a LED	LED
Invio commenti su display a LED	LEDC

Notare che l'elaborazione delle istruzioni seguenti è diversa in una CPU QnA/Q2AS rispetto a una CPU del System Q.

Funzione	Istruzione
Comando, attivazione e disattivazione di operandi interni	OUT, SET, RST
Lettura commento operando	COMRD
Stampa commento	PRC
Cancellazione visualizzazione errore e spia di errore	LEDR
Conversione di dati binari	BIN
Conversione di dati binari	DBIN
Lettura dati orologio	DATERD
Lettura dati orologio	DATEWR
Maschera per programma di interruzione	IMASK
Istruzione di rinfresco	COM
Istruzioni per rinfresco rete	ZCOM
Lettura dati di instradamento	RTREAD
Scrittura dati di instradamento	RTWRITE
Configurazione di controllo in anello chiuso	PIDINT
Controllo in anello chiuso	PIDCONT
Contatore avanti/indietro con ingresso a 1 fase	UDCNT1
Contatore avanti/indietro con ingresso a 2 fase	UDCNT2
Treno di impulsi	SPD
Emissione impulso	PLSY
Modulazione larghezza impulso	PWM

NOTA

Se un programma che accede a un modulo funzione speciale scritto per una CPU QnA viene convertito per una CPU System Q, notare quanto segue:

- *La CPU System Q (modo Q) non è compatibile con i moduli funzione speciali e moduli di rete della serie A/AnS. Per leggere o scrivere dati in questi moduli, usare le istruzioni FROM/TO.*
- *Alcune istruzioni A/AnS possono invece essere utilizzate se i moduli funzione speciali serie QnA, Q2AS, A o AnS vengono sostituiti con moduli funzione speciali System Q. Fare riferimento al manuale relativo al modulo funzione speciale appropriato.*

A.4 Panoramica dei relé speciali

A.4.1 Tabella dei bit speciali diagnostici (MELSEC serie Q e System Q)

I relé speciali diagnostici (SM) sono relé interni il cui significato viene fissato dal PLC. I relé speciali non possono essere utilizzati come gli altri relé interni in un programma di sequenza. Tuttavia alcuni di essi possono essere pilotati ON o OFF per controllare la CPU.

NOTA

I relé speciali da SM1200 a SM1255 vengono usati nelle CPU QnA. Questi relé sono assenti nelle CPU System Q.

I relé speciali a partire da SM1500 sono dedicati per la CPU Q4AR.

Nella tabella seguente vengono spiegati i significati delle colonne delle tabelle alle pagine successive:

Caratteristica	Significato
Numero	Indica il numero dei relé speciali diagnostici.
Nome	Indica il nome del relé speciale diagnostico.
Significato	Contiene una breve descrizione della funzione del relé speciale.
Descrizione	Contiene una descrizione dettagliata del relé speciale diagnostico.
Impostato da (se impostato)	<p>Indica se il relé speciale viene impostato dal sistema o dall'utente.</p> <p><Impostato da></p> <p>S :Impostato dal sistema</p> <p>U :Impostato dall'utente (tramite programma di sequenza o terminale di programmazione in modo test)</p> <p>S/U :Impostato dal sistema o utente</p> <p>Viene indicato solo se l'impostazione viene effettuata dal sistema.</p> <p><se impostato></p> <p>Elaborazione END :Impostato durante l'elaborazione dell'istruzione END</p> <p>Inizializzazione :Impostato durante il processo di inizializzazione (accensione, STOP->RUN)</p> <p>Cambiamento di stato :Impostato dopo un cambiamento di stato</p> <p>Errore :Impostato dopo un errore</p> <p>Esecuzione istruzione :Impostato durante l'esecuzione di una istruzione</p> <p>Richiesta :Impostato per richiesta dell'utente (tramite SM, ecc.)</p>
CPU serie A M9 [] [] []	<p>Indica il relé speciale M9 [] [] [] corrispondente alla CPU serie A (modifica e annotazione se il contenuto viene modificato).</p> <p>Le voci contrassegnate con "Nuovo" sono state aggiunte per le CPU serie Q/System Q.</p>
Valido per:	<p>Indica la CPU corrispondente:</p> <p>●: Utilizzabile da tutte le CPU</p> <p>Q CPU: utilizzabile da CPU System Q</p> <p>CPU QnA: utilizzabile da CPU serie QnA e Q2AS</p> <p>Nome CPU: utilizzabile solo dalla CPU indicata (ad es. CPU Q4AR)</p> <p>Rem: utilizzabile da un modulo di I/O remoto MELSECNET/H</p>

Panoramica dei relé speciali

(1) Informazioni diagnostiche

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	CPU serie A M9[][][]	Valido per:
SM0	Errori diagnostici	OFF: Nessun errore ON: Errore	ON se il risultato della diagnostica indica presenza di errore (comprende diagnostica esterna) Rimane ON anche se viene ripristinato il funzionamento normale.	S (Errore)	Nuovo	● Rem
SM1	Errore di autodiagnosi	OFF: Nessun errore di autodiagnosi ON: Autodiagnosi	Si attiva se si verifica un errore dovuto ad autodiagnosi. Rimane ON anche se viene ripristinato il funzionamento normale.	S (Errore)	M9008	
SM5	Informazioni di errore generali	OFF: Nessuna informazione generale ON: Informazioni di errore generali	Se SM0 è ON, ON se esistono informazioni generali sull'errore.	S (Errore)	Nuovo	
SM16	Informazioni di errore specifiche	OFF: Nessuna informazioni di errore specifica ON: Informazioni di errore specifiche	Se SM0 è ON, ON se esistono informazioni specifiche sull'errore.	S (Errore)	Nuovo	
SM50	Cancellazione errore	OFF → ON: cancellazione errore	Esegue l'operazione di cancellazione errore. Vedi capitolo 5 per ulteriori informazioni.	U	Nuovo	
SM51	Memoria tensione batteria insufficiente	OFF: Normale ON: Tensione batteria insufficiente	On se la tensione di batteria sulla CPU o sulla memory card scende sotto al valore nominale. Rimane ON anche se viene ripristinato il funzionamento normale. Sincrono con il LED BAT. ALARM.	S (Errore)	M9007	●
SM52	Tensione batteria insufficiente	OFF: Normale ON: Tensione batteria insufficiente	Come SM51, ma torna OFF quando la tensione di batteria torna al livello normale.	S (Errore)	M9006	
SM53	Mancanza rete	OFF: Rilevata mancanza rete ON: Alimentazione normale	Commuta su ON quando si utilizza un modulo alimentatore CA e si verifica una interruzione momentanea di alimentazione, non superiore a 20 ms; cancellato spegnendo e riaccendendo il sistema.	S (Errore)	M9005	● Q CPU
			Commuta su ON quando si utilizza un modulo alimentatore CC e si verifica una interruzione momentanea di alimentazione, non superiore a 10 ms; cancellato spegnendo e riaccendendo il sistema.			
			Commuta su ON quando si utilizza un modulo alimentatore CC e si verifica una interruzione momentanea di alimentazione, non superiore a 1 ms; cancellato spegnendo e riaccendendo il sistema.			
SM54	Errori comunicazione MINI	OFF: Normale ON: Errore	Commuta su ON se viene rilevato un errore di comunicazione MINI (S3) su almeno uno dei moduli AJ71PT32 (S3) installati. Rimane ON anche se viene ripristinato il funzionamento normale.	S (Errore)	M9004	CPU QnA
SM56	Errori di esecuzione	OFF: Normale ON: Errore di esecuzione	ON se viene generato un errore di esecuzione. Rimane ON anche se viene ripristinato il funzionamento normale.	S (Errore)	M9011	●
SM60	Rilevamento fusibile interrotto	OFF: Normale ON: Modulo con fusibile interrotto	Commuta su ON quando anche un solo modulo di uscita presenta un fusibile interrotto e rimane ON anche dopo il ripristino. Lo stato di fusibile interrotto viene controllato anche sui moduli di uscita delle stazioni di I/O remoto.	S (Errore)	M9000	● Rem
SM61	Modulo di I/O Errore di verifica	OFF: Normale ON: Errore	Commuta su ON in caso di discrepanza fra gli effettivi moduli di I/O e le informazioni registrate durante l'accensione, La verifica dei moduli di I/O viene effettuata anche sui moduli delle stazioni di I/O remoto.	S (Errore)	M9002	
SM62	Rilevamento spia di errore	OFF: Non rilevata ON: Rilevata	Commuta su ON quando anche una sola spia di errore F si attiva.	S (esecuzione istruzione)	M9009	●

(1) Informazioni diagnostiche

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	CPU serie A M9[][][]	Valido per:
SM80	Rilevamento CHK	OFF: Non rilevata ON: Rilevata	Commuta su ON se viene rilevato un errore dall'istruzione CHK. Rimane ON anche se viene ripristinato il funzionamento normale.	S (esecuzione istruzione)	Nuovo	CPU QnA, CPU Q (tranne CPU Q00J, Q00 e Q01)
SM90	Avviamento timer di watchdog per transizione passo (abilitato solo in caso di programma SFC)	OFF: Non avviato (reset timer di watchdog) ON: Avviato (timer di watchdog avviato)	Corrisponde a SD90	U Commuta su ON quando viene avviata la misura del timer di watchdog delle transizioni di passo. Annulla il timer di watchdog quando commuta su OFF.	M9108	
SM91			Corrisponde a SD91		M9109	
SM92			Corrisponde a SD92		M9110	
SM93			Corrisponde a SD93		M9111	
SM94			Corrisponde a SD94		M9112	
SM95			Corrisponde a SD95		M9113	
SM96			Corrisponde a SD96		M9114	
SM97			Corrisponde a SD97		Nuovo	
SM98			Corrisponde a SD98		Nuovo	
SM99			Corrisponde a SD99		Nuovo	
SM100	Funzione comunicazione seriale in uso	OFF: Comunicazione seriale non usata ON: Comunicazione seriale in uso	Indica se è stata selezionata la funzione di comunicazione seriale nei parametri di configurazione relativi.	S (accensione o reset)	Nuovo	CPU Q00J e Q01
SM101	Flag di stato del protocollo di comunicazione	OFF: Protocollo per dispositivi di programmazione ON: Protocollo MC	Indica se il dispositivo che sta comunicando tramite l'interfaccia RS232 utilizza il protocollo per dispositivi di programmazione, oppure il protocollo MC.	S (comunicazione RS232)	Nuovo	
SM110	Errore di protocollo	OFF: Nessun errore ON: Errore	Commuta su ON quando viene utilizzato un protocollo anomalo per effettuare la comunicazione con la funzione di comunicazione seriale. Rimane ON anche se il protocollo viene successivamente ripristinato.	S (Errore)	Nuovo	
SM111	Stato della comunicazione	OFF: Nessun errore ON: Errore	Commuta su ON se il modo usato per effettuare la comunicazione è diverso da quello impostato con la funzione di comunicazione seriale. Rimane ON anche se il modo viene successivamente corretto.	S (Errore)	Nuovo	
SM112	Cancella informazioni di errore	ON: Cancella relé e registri speciali di diagnostica	Se commutato su ON, il relé speciali per diagnostica SM110 e SM111 vengono azzerati e il contenuto dei registri speciali SD110 e D111 viene azzerato.	U	Nuovo	
SM113	Errore di overrun	OFF: Nessun errore ON: Errore	Commuta su ON quando si verifica un errore di overrun (accavallamento dati) durante la comunicazione seriale.	S (Errore)		
SM114	Errore di parità	OFF: Nessun errore ON: Errore	Commuta su ON se si verifica un errore di parità durante la comunicazione seriale.	S (Errore)		
SM115	Errore di framing	OFF: Nessun errore ON: Errore	Commuta su ON se si verifica un errore di framing durante la comunicazione seriale.	S (Errore)		

Panoramica dei relé speciali

(2) Informazioni di sistema

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	CPU serie A M9[] [] []	Valido per:
SM202	Comando spegnimento LED	OFF → ON : LED spento	Nel passaggio da OFF a ON, i LED corrispondenti ai singoli bit di SD202 vengono spenti.	U	Nuovo	● (tranne CPU Q00J, Q00 e Q01)
SM203	Contatto STOP	Stato STOP	Attivato in stato STOP.	S (variazione di stato)	M9042	U
SM204	Contatto PAUSA	Stato PAUSE	Attivato in stato PAUSE.	S (variazione di stato)	M9041	
SM205	Contatto STEP-RUN	Stato STEP-RUN	Attivato in stato STEP-RUN.	S (variazione di stato)	M9054	● (tranne CPU Q00J, Q00 e Q01)
SM206	Bobina abilitazione PAUSE	OFF: PAUSE disabilitato ON: PAUSE abilitato	Lo stato PAUSE viene attivato se questo relé è ON quando il contatto remoto PAUSE commuta su ON.	U	M9040	U
	Stato del test operandi	OFF: Test operandi non ancora eseguito ON: Test operandi eseguito	Commuta su ON quando il software di programmazione attiva il modo test operandi.	S (Richiesta)	Nuovo	CPU Q00J Q00 e Q01
SM210	Richiesta impostazioni dati orologio	OFF: Ignorata ON: Richiesta accettata	Quando questo relé passa da OFF a ON, i dati dell'orologio contenuti da SD210 a SD213 vengono scritti nell'orologio dopo l'esecuzione dell'istruzione END della scansione in cui avviene la commutazione.	U	M9025	U
SM211	Dati orologio errati	OFF: Nessun errore ON: Errore	ON se viene riscontrato un errore nei dati orologio (da SD210 a SD213), OFF se non vengono rilevati errori.	S (Richiesta)	M9026	
SM212	Visualizzazione dati orologio	OFF: Ignorata ON: Visualizzazione	Visualizza i dati orologio come mese, giorno, ora, minuto e secondo sul display a LED del frontalino della CPU. (Abilitato solo su CPU Q3A e Q4A).	U	M9027	Q3A, Q4A Q4AR CPU
SM213	Richiesta lettura dati orologio	OFF: Ignorata ON: Richiesta di lettura	Quando questo relé è ON, i dati orologio vengono letti da SD210 a SD213 come valori BCD.	U	M9028	● Rem
SM240	Flag reset CPU 1	OFF: Nessun reset ON: La CPU 1 è stata resettata	Questo flag commuta ON quando la CPU numero 1 è stata resettata o è stata rimossa dal telaio. Anche le altre CPU del sistema multi CPU vengono portate in stato reset.	S (variazione di stato)	Nuovo	Q02, Q02H, Q06H, Q12H, Q25H CPU con versione funzionale B o successiva
SM241	Flag reset CPU 2	OFF: Nessun reset ON: La CPU 2 è stata resettata	Questo flag commuta ON quando la CPU numero 2 è stata resettata o è stata rimossa dal telaio. Nelle altre CPU del sistema multi CPU viene emesso il codice di errore 7000 ("MULTI CPU DOWN").	S (variazione di stato)	Nuovo	
SM242	Flag reset CPU 3	OFF: Nessun reset ON: La CPU 3 è stata resettata	Questo flag commuta ON quando la CPU numero 3 è stata resettata o è stata rimossa dal telaio. Nelle altre CPU del sistema multi CPU viene emesso il codice di errore 7000 ("MULTI CPU DOWN").	S (variazione di stato)	Nuovo	
SM243	Flag reset CPU 4	OFF: Nessun reset ON: La CPU 4 è stata resettata	Questo flag commuta ON quando la CPU numero 4 è stata resettata o è stata rimossa dal telaio. Nelle altre CPU del sistema multi CPU viene emesso il codice di errore 7000 ("MULTI CPU DOWN").	S (variazione di stato)	Nuovo	

(2) Informazioni di sistema

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	CPU serie A M9[] [] []	Valido per:
SM244	Flag errore CPU 1	OFF: Nessun errore ON: La CPU numero 1 è in stop a causa di errore	Il flag è attivo per indicare la presenza di un errore che ha arrestato la CPU. Il flag è in stato OFF se la CPU funziona normalmente o se si verifica un errore che non arresta la CPU.	S (variazione di stato)	Nuovo	Q02, Q02H, Q06H, Q12H, Q25H CPU con versione funzionale B o successiva
SM245	Flag errore CPU 2	OFF: Nessun errore ON: La CPU numero 2 è in stop a causa di errore		S (variazione di stato)	Nuovo	
SM246	Flag errore CPU 3	OFF: Nessun errore ON: La CPU numero 3 è in stop a causa di errore		S (variazione di stato)	Nuovo	
SM247	Flag errore CPU 4	OFF: Nessun errore ON: La CPU numero 4 è in stop a causa di errore		S (variazione di stato)	Nuovo	
SM250	Letture numero max. I/O caricati	OFF: Ignorata ON: Lettura	Il relé passa da OFF a ON quando viene letto l'indirizzo più alto degli I/O caricati in SD250.	U	Nuovo	● (tranne CPU Q00J, Q00 e Q01)
SM251	Flag variazione I/O	OFF: Nessuna sostituzione ON: Sostituzione	Dopo che l'indirizzo iniziale di I/O del modulo che viene sostituito è stato impostato online in SD251, la sostituzione del modulo di I/O è abilitata quando questo relé è ON. (Ad ogni impostazione può essere sostituito un solo modulo). Per sostituire un modulo di I/O in stato RUN, usare il programma o un dispositivo periferico per commutare su ON questo relé; per sostituire un modulo di I/O in stato STOP, commutare il relé su ON nel modo test di un dispositivo periferico. Non commutare fra gli stati RUN e STOP fino a quando non è stata completata la sostituzione del modulo di I/O.	S (END)	M9054	Q2A(S1) Q3A, Q4A Q4AR CPU
SM252	Sostituzione I/O abilitata	OFF: Sostituzione proibita ON: Sostituzione abilitata	Commuta su ON quando è abilitata la sostituzione dei moduli di I/O.	S (END)	Nuovo	
SM254	Comando di rinfresco di tutte le stazioni	OFF: Rinfresco della prima stazione ON: Rinfresco di tutte le stazioni	Usato per il rinfresco in blocco e per il ciclo lento. Se il relé è ON, viene eseguito il rinfresco di tutte le stazioni.	S (END)	Nuovo	Q CPU (tranne CPU Q00J, Q00 e Q01)
SM255	MELSECNET/10 informazioni modulo 1	OFF: Rete operativa ON: Rete in standby	In stato ON se la rete è in standby. (In assenza di una indicazione esplicita "active" o "standby", viene assunto "active").	S (Inizializzazione)	Nuovo	● (tranne CPU Q00J, Q00 e Q01)
SM256		OFF: Lettura ON: Nessuna lettura	Nel rinfresco da modulo di comunicazione a CPU (B, W, ecc.) indica lettura dal modulo di comunicazione.	U	Nuovo	
SM257		OFF: Scrittura ON: Nessuna scrittura	Nel rinfresco da CPU a modulo di comunicazione (B, W, ecc.) indica scrittura nel modulo di comunicazione.	U	Nuovo	
SM260	MELSECNET/10 informazioni modulo 2	OFF: Rete operativa ON: Rete in standby	In stato ON se la rete è in standby. (In assenza di una indicazione esplicita "active" o "standby", viene assunto "active").	S (Inizializzazione)	Nuovo	● (tranne CPU Q00J, Q00 e Q01)
SM261		OFF: Lettura ON: Nessuna lettura	Nel rinfresco da modulo di comunicazione a CPU (B, W, ecc.) indica lettura dal modulo di comunicazione.	U	Nuovo	
SM262		OFF: Scrittura ON: Nessuna scrittura	Nel rinfresco da CPU a modulo di comunicazione (B, W, ecc.) indica scrittura nel modulo di comunicazione.	U	Nuovo	

Panoramica dei relé speciali

(2) Informazioni di sistema






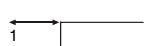




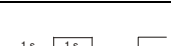
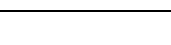

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	CPU serie A M9[] [] []	Valido per:
SM265	MELSECNET/10 informazioni modulo 3	OFF: Rete operativa ON: Rete in standby	In stato ON se la rete è in standby. (In assenza di una indicazioni esplicita "active" o "standby", viene assunto "active").	S (Inizializzazione)	Nuovo	● (tranne CPU Q00J, Q00 e Q01)
SM266		OFF: Lettura ON: Nessuna lettura	Nel rinfresco da modulo di comunicazione a CPU (B, W, ecc.) indica lettura dal modulo di comunicazione.	U	Nuovo	
SM267		OFF: Scrittura ON: Nessuna scrittura	Nel rinfresco da CPU a modulo di comunicazione (B, W, ecc.) indica scrittura nel modulo di comunicazione.	U	Nuovo	
SM270	MELSECNET/10 informazioni modulo 4	OFF: Rete operativa ON: Rete in standby	In stato ON se la rete è in standby. (In assenza di una indicazioni esplicita "active" o "standby", viene assunto "active").	S (Inizializzazione)	Nuovo	● (tranne CPU Q00J, Q00 e Q01)
SM271		OFF: Lettura ON: Nessuna lettura	Nel rinfresco da modulo di comunicazione a CPU (B, W, ecc.) indica lettura dal modulo di comunicazione.	U	Nuovo	
SM272		OFF: Scrittura ON: Nessuna scrittura	Nel rinfresco da CPU a modulo di comunicazione (B, W, ecc.) indica scrittura nel modulo di comunicazione.	U	Nuovo	
SM280	Errore CC-Link	OFF: Normale ON: Errore	Commuta su ON se viene riscontrato un errore CC-Link in uno dei moduli QJ61QBT11 installati. Commuta su OFF al ripristino del funzionamento normale.	S (Errore)	Nuovo	Q CPU (tranne CPU Q00J, Q00 e Q01)
			Commuta su ON se viene riscontrato un errore CC-Link in uno dei moduli A(1s)J61QBT11 installati. Rimane ON anche se viene ripristinato il funzionamento normale.	S (Errore)	Nuovo	CPU QnA
SM315	Flag abilitazione tempo di ritardo comunicazione	OFF: Senza ritardo ON: Con ritardo	Questo flag può essere utilizzato dopo aver impostato il tempo riservato per la comunicazione in SD315. Se il flag è commutato su ON, l'elaborazione di END viene prolungata del tempo impostato in SD315 in assenza di comunicazioni. Il tempo di scansione viene incrementato del tempo impostato in SD315. Se il flag è OFF, l'elaborazione di END viene eseguita senza ulteriore ritardo in assenza di comunicazioni.	U	New	CPU Q00J, Q00 e Q01
SM320	Presenza/assenza di programma SFC	OFF: programma SFC assente ON: programma SFC presente	ON se il programma SFC è stato registrato correttamente, OFF in caso contrario. Commuta su OFF se una istruzione dedicata SFC non è corretta.	S (Inizializzazione)	M9100	
SM321	Avviamento/arresto programma SFC	OFF: programma SFC arrestato ON: programma SFC in esecuzione	Il valore iniziale è lo stesso valore di SM900. (commuta automaticamente su ON se il programma SFC è presente). Il programma SFC non viene eseguito se il relé commuta su OFF prima dell'avviamento del programma SFC. Successivamente, avvia il programma SFC quando passa da OFF a ON. Successivamente, arresta il programma SFC quando passa da ON a OFF.	S/U (Inizializzazione)	M9101 modifica formato	● (tranne CPU Q00J, Q00 e Q01)
SM322	Stato avviamento programma SFC	OFF: Avviamento iniziale ON: Riavviamento	Il valore iniziale è impostato in funzione della parametrizzazione. Se OFF, tutti gli stati relativi al funzionamento vengono cancellati quando il programma SFC viene arrestato; inizia dal primo passo del blocco in cui viene effettuata la richiesta di avviamento. Se ON, inizia dal blocco e dal passo in esecuzione nel momento in cui il programma SFC viene arrestato. (ON viene abilitato solo se la parametrizzazione definisce un riavviamento). SM902 non è automaticamente assegnato come retentivo.	S/U (Inizializzazione)	M9102 modifica formato	

(2) Informazioni di sistema

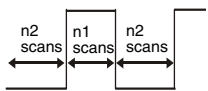
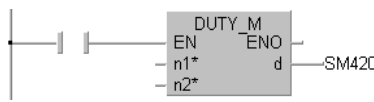
Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	CPU serie A M9[][][]	Valido per:
SM323	Presenza/assenza di transizione continua per l'intero blocco	OFF: Transizione continua non attiva ON: Transizione continua attiva	Se OFF, in ogni ciclo viene elaborata una sola delle transizioni attive del blocco. Se ON, in ogni ciclo vengono elaborate tutte le transizioni attive. Nella specifica dei singoli blocchi, viene assegnata una priorità al bit di transizione continua del blocco. (La specifica viene controllata all'avviamento del blocco),	U	M9103	● (tranne CPU Q00J, Q00 e Q01)
SM324	Flag di inibizione transizione continua	OFF: Se la transizione viene eseguita ON: Nessuna transizione	Se la transizione continua è attiva, il relé commuta su ON se la transizione continua non viene eseguita, commuta su OFF se la transizione continua viene eseguita. Normalmente ON se la transizione continua non è attiva.	S (esecuzione istruzione)	M9104	
SM325	Modo uscita su arresto del blocco	OFF: OFF ON: Inalterata	Se il blocco viene arrestato, seleziona l'azione da eseguire sulle uscite. Tutte le bobine di uscita vengono disattivate se il relé è OFF. Le bobine di uscita rimangono inalterate se il relé è ON.	S (variazione di stato)	M9196	
SM326	Modo azzeramento operandi SFC	OFF: Azzeramento operandi ON: Operandi inalterati	Seleziona lo stato degli operandi quando la CPU arrestata viene portata in RUN dopo che il programma di sequenza o il programma SFC è stato modificato (programma SFC esistente).	U	Nuovo	
SM327	Uscita durante l'esecuzione del passo finale	OFF: OFF ON: Inalterata	Seleziona l'azione sulle uscite quando viene eseguito il passo contenente l'istruzione END. Se il relé è OFF, tutte le bobine di uscita vengono disattivate. Se il relé è ON, tutte le bobine di uscita rimangono inalterate.	S (Inizializzazione) U	Nuovo	
SM330	Modo operativo per programmi a ciclo lento	OFF: Modo asincrono ON: Modo sincrono	Modo asincrono: Modo in cui prosegue l'esecuzione del programma a ciclo lento durante il tempo in eccesso. Modo sincrono: Modo nel quale le operazioni del programma a ciclo lento vengono avviate dalla scansione successiva, anche se esiste tempo in eccesso.	U (END)	Nuovo	

Panoramica dei relé speciali

(3) Orologio di sistema/contatori

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	CPU serie A M9[][][]	Valido per:
SM400	Sempre ON	ON  OFF	Questo flag è normalmente ON	S (ad ogni elaborazione END)	M9036	●
SM401	Sempre ON	ON  OFF	Questo flag è normalmente OFF	S (ad ogni elaborazione END)	M9037	
SM402	ON per 1 scansione dopo l'ingresso in RUN	ON  OFF	ON per 1 sola scansione dopo l'ingresso in RUN Questa connessione può essere usata solo se richiamata una sola volta per scansione.	S (ad ogni elaborazione END)	M9038	
SM403	OFF per 1 sola scansione dopo l'ingresso in RUN	ON  OFF	OFF per 1 sola scansione dopo l'ingresso in RUN Questa connessione può essere usata solo se richiamata una sola volta per scansione.	S (ad ogni elaborazione END)	M9039	
SM404	ON per 1 scansione dopo l'ingresso in RUN	ON  OFF	ON per 1 sola scansione dopo l'ingresso in RUN Questa connessione può essere usata solo se richiamata una sola volta per scansione.	S (ad ogni elaborazione END)	Nuovo	● (tranne CPU Q00J, Q00 e Q01)
SM405	OFF per 1 sola scansione dopo l'ingresso in RUN	ON  OFF	OFF per 1 sola scansione dopo l'ingresso in RUN Questa connessione può essere usata solo se richiamata una sola volta per scansione.	S (ad ogni elaborazione END)	Nuovo	
SM409	Impulso di clock da 0.01 s		Commuta continuamente fra ON e OFF con intervalli da 5 ms. Se il sistema viene spento o viene eseguito un reset, il segnale commuta automaticamente da OFF a ON.	S (variazione di stato)	Nuovo	Q CPU (tranne CPU Q00J, Q00 e Q01)
SM410	Impulso di clock da 0.1 s		Commuta continuamente fra ON e OFF con intervalli di tempo specificati. Questo funzionamento prosegue anche in STOP. Se il sistema viene spento o viene eseguito un reset, il segnale commuta automaticamente da OFF a ON.	S (variazione di stato)	M9030	●
SM411	Impulso di clock da 0.2 s				M9031	
SM412	Impulso di clock da 1 s				M9032	
SM413	Impulso di clock da 2 s				M9033	
SM414	Impulso di clock 2x n secondi				Commuta tra ON e OFF in base al numero di secondi indicato da SD414.	
SM415	Impulso di clock da 2 ms		Commuta tra ON e OFF in base al numero di millisecondi indicato da SD415.	S (variazione di stato)	Nuovo	Q CPU (tranne CPU Q00J, Q00 e Q01)

(3) Orologio di sistema/contatori

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	CPU serie A M9[][]	Valido per:			
SM420	Temporizzatore utente 0		<p>Il relé continua a commutare fra ON e OFF a intervalli di tempo fissi.</p> <p>Se il sistema viene spento o viene eseguito un reset, il segnale commuta automaticamente da OFF a ON.</p> <p>Gli intervalli ON/OFF sono definiti con l'istruzione DUTY.</p> 	S (ad ogni elaborazione END)	M9020	●			
SM421	Temporizzatore utente 1				M9021				
SM422	Temporizzatore utente 2				M9022				
SM423	Temporizzatore utente 3				M9023				
SM424	Temporizzatore utente 4				M9024				
SM430	Temporizzatore utente 5								
SM431	Temporizzatore utente 6								
SM432	Temporizzatore utente 7				Da usare con SM420 - SM424 in programmi a ciclo lento.				
SM433	Temporizzatore utente 8						S (ad ogni elaborazione END)	Nuovo	● (tranne CPU Q00J, Q00 e Q01)
SM434	Temporizzatore utente 9								

(4) Informazioni scansione

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	CPU serie A M9[][]	Valido per:
SM510	Flag di esecuzione programma a ciclo lento	OFF: Completato o non eseguito ON: Esecuzione in corso	ON durante l'esecuzione del programma a ciclo lento.	S (ad ogni elaborazione END)	Nuovo	● (tranne CPU Q00J, Q00 e Q01)
SM551	Legge intervallo di service del modulo	OFF: Ignorata ON: Lettura	Quando il relé passa da OFF a ON, l'intervallo di service del modulo, designato da SD550, viene letto in SD551 e SD552.	U	Nuovo	

Panoramica dei relé speciali

(5) Memory card

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	CPU serie A M9[][][]	Valido per:
SM600	Flag memory card A disponibile	OFF: Non disponibile ON: Utilizzabile	ON se la memory card A è pronta per l'uso.	S (Inizializzazione)	Nuovo	● (tranne CPU Q00J, Q00 e Q01)
SM601	Flag protezione memory card A	OFF: Non protetta ON: Protetta	Commuta su ON se l'interruttore di protezione della memory card A è su ON.	S (Inizializzazione)	Nuovo	
SM602	Flag drive 1	OFF: Nessun drive 1 ON: Drive 1 presente	Commuta su ON se il drive 1 (scheda 1 area RAM) è presente.	S (Inizializzazione)	Nuovo	
SM603	Flag drive 2	OFF: Nessun drive 2 ON: Drive 2 presente	Commuta su ON se il drive 2 (scheda 1 area ROM) è presente.	S (Inizializzazione)	Nuovo	
SM604	Flag memory card A in uso	OFF: non usata ON: in uso	Commuta su ON se la memory card A è in uso.	S (Inizializzazione)	Nuovo	
SM605	Flag inibizione rimozione/inserimento memory card A	OFF: Rimozione/inserimento abilitati ON: Rimozione/inserimento disabilitati	Commuta su ON quando la memory card A non può essere inserita o rimossa.	U	Nuovo	
SM620	Flag memory card B disponibile	OFF: Non disponibile ON: Utilizzabile	Sempre ON	S (Inizializzazione)	Nuovo	Q CPU
			ON se la memory card B è pronta per l'uso.	S (Inizializzazione)	Nuovo	Q2A(S1) Q3A Q4A Q4AR
SM621	Flag protezione memory card B	OFF: Non protetta ON: Protetta	Sempre ON	S (Inizializzazione)	Nuovo	Q CPU
			Commuta su ON se l'interruttore di protezione della memory card B è su ON.	S (Inizializzazione)	Nuovo	Q2A(S1) Q3A Q4A Q4AR
SM622	Flag drive 3	OFF: Nessun drive 3 ON: Drive 3 presente	Sempre ON	S (Inizializzazione)	Nuovo	Q CPU
			Commuta su ON se il drive 3 (scheda 2 area RAM) è presente.		Nuovo	Q2A(S1) Q3A Q4A Q4AR
SM623	Flag drive 4	OFF: Nessun drive 4 ON: Drive 4 presente	Sempre ON	S (Inizializzazione)	Nuovo	Q CPU
			Commuta su ON se il drive 4 (scheda 2 area ROM) è presente.	S (Inizializzazione)	Nuovo	Q2A(S1) Q3A Q4A Q4AR
SM624	Flag memory card B in uso	OFF: non usata ON: in uso	Commuta su ON se la memory card B è in uso.	S (Inizializzazione)	Nuovo	
SM625	Flag inibizione rimozione/inserimento memory card B	OFF: Rimozione/inserimento abilitati ON: Rimozione/inserimento disabilitati	Commuta su ON quando la memory card B non può essere inserita o rimossa.	U	Nuovo	

(5) Memory card (continua)

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	CPU serie A M9[] [] []	Valido per:
SM640	Uso file registri	OFF: File registri non usato ON: File registri in uso	Commuta su ON quando il file registri è in uso	S (variazione di stato)	Nuovo	●
SM650	Uso commenti	OFF: Commento non usato ON: Commento usato	Commuta su ON quando il file commenti è in uso	S (variazione di stato)	Nuovo	● (tranne CPU Q00J, Q00 e Q01)
SM660	Funzionamento boot	OFF: esecuzione da memoria interna ON: boot in corso	Commuta su ON mentre è attivo il processo di boot. Commuta su OFF se il commutatore di selezione boot è OFF.	S (variazione di stato)	Nuovo	●
SM672	Flag campo di accesso a file registri su memory card A	OFF: Nel campo ammesso ON: Oltre il campo ammesso	Commuta su ON quando si accede ad un'area al di fuori del campo del file registri R della memory card A (impostato durante elaborazione END). Azzerato dal programma utente.	S/U	Nuovo	● (tranne CPU Q00J, Q00 e Q01)
SM673	Flag campo di accesso a file registri su memory card B	OFF: Nel campo ammesso ON: Oltre il campo ammesso	Commuta su ON quando si accede ad un'area al di fuori del campo del file registri R della memory card B (impostato durante elaborazione END). Azzerato dal programma utente.	S/U	Nuovo	Q2A(S1) Q3A Q4A Q4AR

(6) Relé speciali diagnostici relativi alle istruzioni

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	CPU serie A M9[] [] []	Valido per:
SM700	Segnale di riporto	OFF: Riporto OFF ON: Riporto ON	Segnale di riporto utilizzato dalle istruzioni applicative.	S (esecuzione istruzione)	M9012	●
SM701	Scelta del numero di caratteri in uscita	OFF: invio di 16 caratteri ON: Invio fino a NUL	Se SM701 è OFF, vengono inviati in uscita 16 caratteri ASCII. Se SM701 è ON, l'invio continua fino ad incontrare il carattere NUL (00 _h).	U	M9049	● (tranne CPU Q00J, Q00 e Q01)
SM702	Metodo di ricerca	OFF: Ricerca successivo ON: ricerca in due fasi	Indica il metodo utilizzato dall'istruzione di ricerca. I dati devono essere disposti per una ricerca a due fasi.	U	Nuovo	●
SM703	Tipo di ordinamento	OFF: Ordine ascendente ON: Ordine discendente	L'istruzione di ordinamento viene usata per decidere se i dati devono essere ordinati in senso crescente o decrescente,	U	Nuovo	
SM704	Confronto blocchi	OFF: Nessuna corrispondenza ON: Corrispondenza	Commuta su ON se tutte le condizioni dell'istruzione BKCOMP sono state verificate.	S (esecuzione istruzione)	Nuovo	
SM707	Scelta del tipo di elaborazione per le istruzioni per numeri reali	OFF: Ottimizzazione velocità ON: Ottimizzazione precisione	Se SM707 è OFF, le istruzioni in virgola mobile vengono processate in modo veloce Se SM707 è ON, le istruzioni in virgola mobile vengono processate in modo preciso	U	Nuovo	Q4AR

Panoramica dei relé speciali

(6) Relé speciali diagnostici relativi alle istruzioni

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	CPU serie A M9[] [] []	Valido per:
SM710	Flag determinazione priorità istruzione CHK	OFF: priorità allo stato dei contatti ON: priorità alla catena di condizioni	Se OFF, rimane come impostato in origine. Se ON la priorità dell'istruzione CHK viene aggiornata.	S (esecuzione istruzione)	Nuovo	● (tranne CPU Q00J, Q00 e Q01)
SM711	Stato trasmissione suddivisa	OFF: Durante l'esecuzione di altre istruzioni ON: Durante l'elaborazione suddivisa	Unitamente al modulo AD57(S1), il relé commuta ON quando lo schermo viene suddiviso durante il trasferimento, torna OFF al termine.	S (esecuzione istruzione)	M9065	QnA
SM712	Scelta tipo trasmissione	OFF: Trasmissione in blocco ON: Trasmissione suddivisa	Unitamente a AD57(S1), commuta su ON quando lo schermo del display viene diviso per il trasferimento dati.	S (esecuzione istruzione)	M9066	
SM714	Segnale di occupato (BUSY) dell'area di registrazione della richiesta di comunicazione	OFF: Richiesta di comunicazione con modulo terminale abilitata ON: Richiesta di comunicazione con modulo terminale disabilitata	Usato per determinare se è possibile eseguire le richieste di comunicazione con i terminali remoti collegati con AJ71PT32-S3 o A2CCPU.	S (esecuzione istruzione)	M9081	
SM715	Flag EI	OFF: Durante DI ON: Durante EI	ON quando viene eseguita l'istruzione EI.	S (esecuzione istruzione)	Nuovo	
SM720	Flag di completamento lettura commento	OFF: Lettura commento non completata ON: lettura commento completata	SM720 viene attivato per una scansione al termine dell'esecuzione dell'istruzione COMRD o PRC.	S (variazione di stato)	Nuovo	Q CPU (tranne CPU Q00J, Q00 e Q01)
SM721	File a cui si accede	OFF: File non oggetto di accesso ON: File oggetto di accesso	Questo flag è ON durante l'accesso a un file con una istruzione S.FWRITE, S.FREAD, COMRD, PRC, o LEDC.	S (variazione di stato)	Nuovo	Q CPU
SM722	Flag disabilitazione errore istruzioni BIN/DBIN	OFF: Errore abilitato ON: Errore disabilitato	Se il flag è attivo, l'errore "OPERATION ERROR" non viene emesso per entrambe le istruzioni BIN e DBIN.	U	Nuovo	
SM730	Segnale BUSY (occupato) per l'area di registrazione della richiesta di comunicazione CC-Link	OFF: Richiesta di comunicazione con stazione intelligente abilitata ON: Richiesta di comunicazione con stazione intelligente disabilitata	Questo flag viene usato per determinare l'abilitazione o disabilitazione della richiesta di comunicazione per una stazione intelligente collegata con A(1S)J61QBT11.	S (esecuzione istruzione)	Nuovo	CPU QnA
SM736	Flag esecuzione istruzione PKEY in corso	OFF: Istruzione non in esecuzione ON: Istruzione eseguita	ON quando viene eseguita l'istruzione PKEY. Torna OFF quando si riceve CR o quando la stringa ricevuta raggiunge 32 caratteri.	S (esecuzione istruzione)	Nuovo	
SM737	Flag ricezione ingresso da tastiera per istruzione PKEY	OFF: Ricezione ingresso da tastiera abilitata ON: Ricezione ingresso da tastiera disabilitata	In stato ON se si riceve input da tastiera. Torna OFF quando i caratteri ricevuti dalla tastiera sono stati memorizzati nella CPU.	S (esecuzione istruzione)	Nuovo	● (tranne CPU Q00J, Q00 e Q01)
SM738	Flag ricezione istruzione MSG	OFF: Istruzione non in esecuzione ON: Istruzione eseguita	Commuta su ON quando viene eseguita l'istruzione MSG.	S (esecuzione istruzione)	Nuovo	

(6) Relé speciali diagnostici relativi alle istruzioni

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	CPU serie A M9[] [] []	Valido per:
SM774	Elaborazione PID tipo bumpless	OFF: Forza corrispondenza ON: Nessuna forzatura	In modo manuale, indica se si deve forzare o meno il valore SV per farlo corrispondere al valore PV.	U	Nuovo	● (tranne CPU Q00J, Q00 e Q01)
SM775	Selezione rinfresco del collegamento durante l'esecuzione dell'istruzione COM	OFF: Esegue rinfresco collegamento ON: Nessun rinfresco eseguito	Seleziona se eseguire o meno il rinfresco del collegamento nei casi in cui viene eseguita solo una elaborazione dati generale durante l'esecuzione dell'istruzione COM.	U	Nuovo	U
SM776	Abilita operandi locali con CALL	OFF: Operandi locali disabilitati ON: Operandi locali abilitati	Il flag specifica se abilitare o disabilitare gli operandi locali nei programmi richiamati con l'istruzione CALL.	U	Nuovo	● (tranne CPU Q00J, Q00 e Q01)
SM777	Abilitazione operandi locali nei programmi di interruzione	OFF: Operandi locali disabilitati ON: Operandi locali abilitati	Il flag specifica se abilitare o disabilitare gli operandi locali nei programmi richiamati di interruzione.	U	Nuovo	● (tranne CPU Q00J, Q00 e Q01)
SM780	Abilitazione istruzioni specifiche CC-Link	OFF: Istruzioni specifiche CC-Link abilitate ON: Istruzioni specifiche CC-Link non abilitate	Questo flag commuta su ON quando il numero di istruzioni specifiche per CC-Link che possono essere eseguite contemporaneamente raggiunge 32. Se il numero scende sotto 32, il flag viene azzerato.	S (variazione di stato)	Nuovo	CPU QnA

Panoramica dei relé speciali

(7) Debug

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	CPU serie A M9[] [] []	Valido per:
SM800	Preparazione traccia	OFF: Non pronta ON: Pronta	Commuta su ON al termine della preparazione della traccia.	S (variazione di stato)	Nuovo	Q CPU (tranne CPU Q00J, Q00 e Q01)
	Preparazione traccia campionamenti		Commuta su ON se la traccia di campionamento è pronta	S (variazione di stato)		CPU QnA
SM801	Avviamento traccia	OFF: Sospesa ON: Avviata	La traccia è attiva quando il flag è ON. Sospesa se OFF (i relé speciali M collegati sono tutti OFF).	U	M9047	Q CPU (tranne CPU Q00J, Q00 e Q01)
	Avviamento traccia campionamento		La traccia di campionamento è attiva quando il flag è ON. Sospesa se OFF (i relé speciali M collegati sono tutti OFF).	U		CPU QnA
SM802	Esecuzione traccia in corso	OFF: Sospesa ON: Avviata	ON durante l'esecuzione della traccia	S (variazione di stato)	M9046	Q CPU (tranne CPU Q00J, Q00 e Q01)
	Esecuzione traccia campionamento in corso		ON durante l'esecuzione della traccia di campionamento	S (variazione di stato)		CPU QnA
SM803	Trigger della traccia	OFF → ON: attivato	Il trigger della traccia commuta su ON quando il flag passa da OFF a ON (come la condizione di esecuzione dell'istruzione TRACE).	U	M9044	Q CPU (tranne CPU Q00J, Q00 e Q01)
	Trigger traccia di campionamento		Il trigger della traccia di campionamento commuta su ON quando il flag passa da OFF a ON (come la condizione di esecuzione dell'istruzione STRA).	U		CPU QnA
SM804	Trigger post traccia	OFF: Nessun trigger ON: Trigger post traccia	Commuta su ON quando il trigger della traccia è attivo.	S (variazione di stato)	Nuovo	Q CPU (tranne CPU Q00J, Q00 e Q01)
	Trigger post traccia di campionamento		Commuta su ON dopo l'attivazione del trigger della traccia di campionamento	S (variazione di stato)		CPU QnA
SM805	Traccia completata	OFF: Non completata ON: Completata	Commuta su ON al completamento della traccia	S (variazione di stato)	M9043	Q CPU (tranne CPU Q00J, Q00 e Q01)
	Traccia di campionamento completata		Commuta su ON al completamento della traccia di campionamento	S (variazione di stato)		CPU QnA
SM806	Preparazione memoria di stato	OFF: Non pronta ON: Pronta	Commuta su ON se la memoria di stato è pronta	S (variazione di stato)	Nuovo	CPU QnA
SM807	Comando memorizzazione stato	OFF → ON: memorizza	Esegue il comando di memorizzazione dello stato	U	Nuovo	
SM808	Completamento memorizzazione stato	OFF: Memorizzazione non completata ON: Memorizzazione completata	Commuta su ON al completamento della memorizzazione di stato	S (variazione di stato)	M9055	
SM809	Cancellazione memoria di stato	OFF → ON: cancella	Abilitazione memoria stato successivo	U	Nuovo	
SM810	Preparazione traccia di programma	OFF: Non pronta ON: Pronta	Commuta su ON se la traccia di programma è pronta	S (variazione di stato)	Nuovo	

(7) Debug

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	CPU serie A M9[][][]	Valido per:
SM811	Avviamento traccia di programma	OFF: Sospesa ON: Avviata	La traccia di programma è attiva quando il flag è ON. Sospesa se OFF (i relé speciali M collegati sono tutti OFF).	U	Nuovo	CPU QnA
SM812	Esecuzione traccia di programma in corso	OFF: Sospesa ON: Avviata	ON durante l'esecuzione della traccia di programma	S (variazione di stato)	Nuovo	
SM813	Trigger traccia di programma	OFF → ON: attivato	Il trigger della traccia di programma commuta su ON quando il flag passa da OFF a ON (come la condizione di esecuzione dell'istruzione PTRR).	U	Nuovo	
SM814	Trigger post traccia di programma	OFF: Nessun trigger ON: Post trigger	Commuta su ON sul post trigger della traccia di programma	S (variazione di stato)	Nuovo	
SM815	Completamento traccia di programma	OFF: Non completato ON: Completato	Commuta su ON al completamento della traccia di programma	S (variazione di stato)	Nuovo	
SM820	Preparazione traccia passi	OFF: Non pronta ON: Pronta	Commuta su ON dopo la registrazione della traccia passi	S (variazione di stato)	Nuovo	● (tranne CPU Q00J, Q00 e Q01)
SM821	Avviamento traccia passi	OFF: Sospesa ON: Avviata	Quando questo flag commuta su ON, viene avviata la traccia passi Sospesa se OFF (i relé speciali M collegati sono tutti OFF).	U	M9182 modifica formato	
SM822	Esecuzione traccia passi in corso	OFF: Sospesa ON: Start	ON durante l'esecuzione della traccia passi Commuta su OFF al completamento o sospensione	S (variazione di stato)	M9181	
SM823	Trigger post traccia passi	OFF: Nessun trigger ON: Post trigger	Commuta su ON se viene attivato il trigger di almeno 1 blocco della traccia passi in esecuzione Commuta su OFF quando la traccia passi si avvia.	S (variazione di stato)	Nuovo	
SM824	Traccia passi Post trigger	OFF: Non dopo tutti i trigger ON: Dopo tutti i trigger	Commuta su ON se viene attivato il trigger di tutti i blocchi della traccia passi in esecuzione Commuta su OFF quando la traccia passi si avvia.	S (variazione di stato)	Nuovo	
SM825	Traccia passi completata	OFF: Non completata ON: Completata	Commuta su ON al completamento della traccia passi Commuta su OFF quando la traccia passi si avvia.	S (variazione di stato)	M9180	Q CPU (tranne CPU Q00J, Q00 e Q01)
SM826	Errore traccia	OFF: Normale ON: Errore	Commuta su ON se si verificano errori durante l'esecuzione della traccia/traccia di campionamento	S (variazione di stato)	Nuovo	
	Errore traccia di campionamento			S (variazione di stato)		
SM827	Errore memoria di stato	OFF: Normale ON: Errore	Commuta su ON se si verificano errori durante l'esecuzione della memorizzazione stato	S (variazione di stato)	Nuovo	CPU QnA
SM828	Errore traccia di programma	OFF: Normale ON: Errore	Commuta su ON se si verificano errori durante l'esecuzione della traccia di programma	S (variazione di stato)	Nuovo	

(8) Area retentiva

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	CPU serie A M9[][][]	Valido per:
SM900	Accesso a file durante mancanza rete	OFF: Nessun accesso a file durante mancanza rete ON: Accesso a file durante mancanza rete	Commuta su ON se si verifica una mancanza rete durante l'accesso a file.	S/U (variazione di stato)	Nuovo	CPU QnA
SM910	Flag registrazione RKEY	OFF: Ingresso da tastiera non registrato ON: Ingresso da tastiera registrato	Commuta su ON con la registrazione dell'ingresso da tastiera. OFF se l'ingresso da tastiera non è registrato.	S (esecuzione istruzione)	Nuovo	● (tranne CPU Q00J, Q00 e Q01)

Panoramica dei relé speciali

(9) Corrispondenza conversione da serie A a System Q/serie QnA

Nella conversione da serie A MELSEC a serie Q o System Q MELSEC, occorre considerare che i relé speciali da M9000 a M9255 (serie A) corrispondono ai relé diagnostici da SM1000 a SM1255 (System Q/serie Q).

Questi relé speciali diagnostici sono tutti impostati dal sistema e non possono essere modificati dal programma utente, Gli utenti che necessitano di attivare o disattivare questi relé devono modificare i loro programmi in modo da utilizzare solo relé speciali diagnostici System Q/QnA. Una eccezione è costituita dai relé speciali M9084 e da M9200 a M9255. Se un utente può attivare o disattivare alcuni di questi relé speciali prima della conversione, potrà anche attivare o disattivare i relé corrispondenti tra SM1084 e SM1200 - SM1255 dopo la conversione. Fare riferimento ai manuali delle CPU e delle reti MELSECNET e MELSECNET/B per informazioni dettagliate sui relé speciali della serie A.

NOTA

Il tempo di elaborazione può risultare maggiore se i relé speciali convertiti vengono usati con una CPU System Q. Non selezionare "A-PLC: Use special relay/special register from SM/SD 1000" (usare relé/registri speciali da SM/SD1000) nella configurazione del sistema PC fra i parametri di GX Developer, se non si usano relé speciali convertiti.

Se viene specificato un relé speciale per la modifica, l'indirizzo dell'operando deve essere modificato con quello corrispondente ai relé speciali delle CPU System Q/QnA. Se non viene specificato un relé speciale per la modifica, il relé speciale convertito può essere utilizzato come indirizzo di operando.

Tabella relé speciali e relé diagnostici

Relé speciale CPU serie A	Relé speciale dopo la conversione	Relé speciale diagnostico equivalente System Q/QnA	Nome	Significato	Valido per:
M9000	SM1000	?—	Fusibile interrotto	OFF: Normale ON: Presente un modulo con fusibile interrotto	CPU System Q/ QnA
M9002	SM1002	?—	Errore verifica modulo di I/O	OFF: Normale ON: Errore	
M9004	SM1004	?—	Errori comunicazione MINI	OFF: Normale ON: Errore	
M9005	SM1005	?—	Mancanza rete	OFF: AC DOWN non rilevato ON: Rilevato AC DOWN	CPU QnA
M9006	SM1006	?—	Tensione batteria insufficiente	OFF: Normale ON: Tensione batteria insufficiente	
M9007	SM1007	?—	Memoria tensione batteria insufficiente	OFF: Normale ON: Tensione batteria insufficiente	
M9008	SM1008	SM1	Errore di autodiagnosi	OFF: Nessun errore ON: Errore	
M9009	SM1009	SM62	Rilevamento spia di errore	OFF: Nessun indirizzo F rilevato ON: Indirizzo F rilevato	
M9011	SM1011	SM56	Errore di esecuzione	OFF: Nessun errore ON: Errore	
M9012	SM1012	SM700	Segnale di riporto	OFF: Riporto OFF ON: Riporto ON	
M9016	SM1016	L'operando non è valido per una CPU System Q/QnA	Flag cancellazione memoria dati	OFF: Ignorata ON: Cancellazione	
M9017	SM1017	L'operando non è valido per una CPU System Q/QnA	Flag cancellazione memoria dati	OFF: Ignorata ON: Cancellazione	

Tabella relé speciali e relé diagnostici

Relé speciale CPU serie A	Relé speciale dopo la conversione	Relé speciale diagnostico equivalente System Q/QnA	Nome	Significato	Valido per:
M9020	SM1020	?—	Temporizzatore utente 0		
M9021	SM1021	?—	Temporizzatore utente 1		
M9022	SM1022	?—	Temporizzatore utente 2		
M9023	SM1023	?—	Temporizzatore utente 3		
M9024	SM1024	?—	Temporizzatore utente 4		
M9025	SM1025	?—	Richiesta impostazioni dati orologio	OFF: Ignorata ON: Richiesta impostazione presente	CPU System Q/QnA
M9026	SM1026	?—	Dati orologio errati	OFF: Nessun errore ON: Errore	
M9027	SM1027	?—	Visualizzazione dati orologio	OFF: Ignorata ON: Visualizzazione	
M9028	SM1028	—	Richiesta lettura dati orologio	OFF: Ignorata ON: Richiesta di lettura	
M9029	SM1029	L'operando non è valido per una CPU System Q/QnA	Elaborazione in blocco di richiesta di comunicazione	OFF: Elaborazione in gruppo non eseguita ON: Elaborazione in blocco eseguita	
M9030	SM1030	—	Impulso di clock da 0.1 s		
M9031	SM1031	—	Impulso di clock da 0.2 s		
M9032	SM1032	—	Impulso di clock da 1 s		
M9033	SM1033	—	Impulso di clock da 2 s		
M9034	SM1034	—	Impulso da 1 minuto		
M9036	SM1036	—	Sempre ON	ON ————— OFF	
M9037	SM1037	—	Sempre OFF	ON OFF —————	
M9038	SM1038	—	ON per 1 scansione dopo l'ingresso in RUN	ON OFF ———— 1	

Panoramica dei relé speciali

Tabella relé speciali e relé diagnostici speciali (continua)

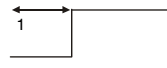
Relé speciale CPU serie A	Relé speciale dopo la conversione	Relé speciale diagnostico System Q/QnA equivalente	Nome	Significato	Valido per:	
M9039	SM1039	—	Flag RUN (OFF per 1 sola scansione dopo l'ingresso in RUN)	ON OFF 	CPU System Q/QnA	
M9040	SM1040	SM206	Bobina abilitazione PAUSE	OFF: PAUSE disabilitato ON: PAUSE abilitato		
M9041	SM1041	SM204	Contatto stato PAUSE	OFF: PAUSE non attiva ON: PAUSE attiva		
M9042	SM1042	SM203	Contatto stato STOP	OFF: STOP non attivo ON: STOP attivo		
M9043	SM1043	SM805	Traccia di campionamento completata	OFF: Traccia campionamento in corso ON: Traccia di campionamento completata		
M9044	SM1044	SM803	Traccia campionamento	0 → 1 STRA Stessa funzione 1 → 0 STRAR Stessa funzione		
M9045	SM1045	L'operando non è valido per una CPU System Q/QnA	Reset timer di watchdog (WDT)	OFF: Nessun reset WDT ON: Reset WDT		
M9046	SM1046	SM802	Traccia campionamento	OFF: Esecuzione traccia a riposo ON: Traccia in esecuzione		
M9047	SM1047	SM801	Preparazione traccia campionamenti	OFF: Traccia campionamenti sospesa ON: Traccia campionamenti avviata		
M9049	SM1049	SM701	Selezione numero caratteri inviati	OFF: Invio fino a NUL ON: emissione di 16 caratteri		
M9051	SM1051	L'operando non è valido per una CPU System Q/QnA	Disabilitazione esecuzione istruzione CHG	OFF: Abilitata ON: Disabilitata		
M9052	SM1052	L'operando non è valido per una CPU System Q/QnA	Commutatore istruzione SEG	OFF: Display a 7 segmenti ON: Rinfresco parziale I/O		
M9054	SM1054	SM205	Flag STEP RUN	OFF: STEP RUN non attivo ON: STEP RUN attivo		
M9055	SM1055	SM808	Flag completamento memorizzazione stato	OFF: Non completato ON: Completato		CPU QnA
M9056	SM1056	Questi operandi non sono validi per una CPU System Q/QnA	Richiesta impostazione P, I programma principale	OFF: Richiesta di altre istruzioni ON: Richiesta impostazione P, I		CPU System Q/QnA
M9057	SM1057		Richiesta impostazione P, I sottoprogramma			
M9058	SM1058		Completamento impostazione P, I programma principale	Temporaneamente ON al completamento impostazione P, I		
M9059	SM1059		Completamento impostazione P, I sottoprogramma	Temporaneamente ON al completamento impostazione P, I		
M9060	SM1060		Richiesta impostazione P, I sottoprogramma 2	OFF: Richiesta di altre istruzioni ON: Richiesta impostazione P, I		
M9061	SM1061		Richiesta impostazione P, I sottoprogramma 3			

Tabella relé speciali e relé diagnostici speciali (continua)

Relé speciale CPU serie A	Relé speciale dopo la conversione	Relé speciale diagnostico System Q/QnA equivalente	Nome	Significato	Valido per:
M9065	SM1065	SM711	Rilevamento esecuzione processo diviso	OFF: Elaborazione divisa non eseguita ON: Durante l'elaborazione suddivisa	CPU QnA
M9066	SM1066	SM712	Flag richiesta elaborazione suddivisa	OFF: Elaborazione in blocco ON: Elaborazione suddivisa	
M9070	SM1070	L'operando non è valido per una CPU System Q/QnA	Tempo di ricerca necessario A8UPU/A8PUJ	OFF: Tempo di lettura non abbreviato ON: Tempo di lettura abbreviato	CPU System Q/QnA
M9081	SM1081	SM714	Segnale di occupato (BUSY) dell'area di registrazione della richiesta di comunicazione	OFF: Spazio libero nell'area di registrazione della richiesta ON: Nessuno spazio libero nell'area di comunicazione della richiesta	CPU QnA
M9084	SM1084	L'operando non è valido per una CPU System Q/QnA	Controllo errori	OFF: Eseguito controllo errore ON: Nessun controllo errore	CPU System Q/QnA
M9091	SM1091	L'operando non è valido per una CPU System Q/QnA	Flag errore istruzione	OFF: Nessun errore ON: Errore	
M9094	SM1094	SM251	Flag variazione I/O	OFF: Sostituzione ON: Nessuna sostituzione	CPU QnA
M9100	SM1100	SM320	Presenza/assenza di programma SFC	OFF: Programma SFC non usato ON: Programma SFC in uso	CPU System Q/QnA
M9101	SM1101	SM321	Avviamento/arresto programma SFC	OFF: Programma SFC arrestato ON: Programma SFC in esecuzione	
M9102	SM1102	SM322	Stato avviamento programma SFC	OFF: Avviamento iniziale ON: Continua	
M9103	SM1103	SM323	Presenza/assenza di transizione continua	OFF: Transizione continua non attiva ON: Transizione continua attiva	
M9104	SM1104	SM324	Flag di sospensione transizione continua	OFF: Se la transizione è completata ON: Nessuna transizione	
M9108	SM1108	SM90	Avviamento timer di watchdog transizioni (equivalente a D9108)	OFF: Reset timer di watchdog ON: Avviamento timer di watchdog	
M9109	SM1109	SM91	Avviamento timer di watchdog transizioni (equivalente a D9109)		
M9110	SM1110	SM92	Avviamento timer di watchdog transizioni (equivalente a D9110)		
M9111	SM1111	SM93	Avviamento timer di watchdog transizioni (equivalente a D9111)		
M9112	SM1112	SM94	Avviamento timer di watchdog transizioni (equivalente a D9112)		
M9113	SM1113	SM95	Avviamento timer di watchdog transizioni (equivalente a D9113)		
M9114	SM1114	SM96	Avviamento timer di watchdog transizioni (equivalente a D9114)		
M9180	SM1180	SM825	Flag di esecuzione traccia campionamento passo attivo		
M9181	SM1181	SM822	Flag di esecuzione traccia campionamento passo attivo	OFF: Traccia non eseguita ON: Traccia in esecuzione	

Panoramica dei relé speciali

Tabella relé speciali e relé diagnostici speciali (continua)

Relé speciale CPU serie A	Relé speciale dopo la conversione	Relé speciale diagnostico System Q/QnA equivalente	Nome	Significato	Valido per:
M9182	SM1182	SM821	Abilitazione traccia campionamento passo attivo	OFF: Traccia disabilitata/sospesa ON: Traccia abilitata	CPU System Q/QnA
M9196	SM1196	SM325	Modo uscita su arresto del blocco	OFF: Uscita bobine OFF ON: Uscita bobine ON	
M9197 M9198	SM1197 SM1198	L'operando non è valido per una CPU System Q/QnA	Commuta fra visualizzazione fusibile interrotto e errore di verifica I/O	La visualizzazione varia in funzione della combinazione dello stato ON/OFF di M9197 e M9198.	
M9199	SM1199	L'operando non è valido per una CPU System Q/QnA	Ricupero on-line dei dati della memoria stato traccia di campionamento	OFF: Non esegue recupero dati ON: Esegue recupero dati	
M9200	SM1200	—	Ricezione istruzione LRDP	OFF: Non accettata ON: Accettata	CPU QnA
M9201	SM1201	—	Completamento istruzione LRDP	OFF: Non completato ON: Completata	
M9202	SM1202	—	Ricezione istruzione LWTP	OFF: Non accettata ON: Accettata	
M9203	SM1203	—	Completamento istruzione LWTP	OFF: Non completato ON: Completata	
M9204	SM1204	—	Completamento istruzione LRDP	OFF: Non completato ON: Completata	
M9205	SM1205	—	Completamento istruzione LWTP	OFF: Non completato ON: Completata	
M9206	SM1206	—	Errore nei parametri di comunicazione stazione host	OFF: Normale ON: Anomalo	
M9207	SM1207	—	Risultato controllo parametri di comunicazione	OFF: Sì ON: No	
M9208	SM1208	—	Imposta i campi di trasmissione B e W delle stazioni master di livello inferiore.	OFF: Trasmissione a livello 2 e 3 ON: Trasmissione solo a livello 2	
M9209	SM1209	—	Comando controllo parametri di comunicazione per stazioni master di livello inferiore.	OFF: Esegue la funzione di controllo ON: Controllo non eseguito	
M9210	SM1210	—	Errore scheda di comunicazione (per stazione locale)	OFF: Normale ON: Anomalo	
M9211	SM1211	—	Errore modulo di comunicazione (per stazione master)	OFF: Normale ON: Anomalo	
M9224	SM1224	—	Stato comunicazione	OFF: Online ON: Offline, test stazione-stazione, o test self-loopback	
M9225	SM1225	—	Errore anello avanti	OFF: Normale ON: Anomalo	
M9226	SM1226	—	Errore anello indietro	OFF: Normale ON: Anomalo	
M9227	SM1227	—	Test stato anello	OFF: Non eseguito ON: Test anello avanti o indietro in corso	

Tabella relé speciali e relé diagnostici speciali (continua)

Relé speciale CPU serie A	Relé speciale dopo la conversione	Relé speciale diagnostico System Q/QnA equivalente	Nome	Significato	Valido per:
M9232	SM1232	?	Stato operativo stazione locale	OFF: Stato RUN o STEP RUN ON: Stato STOP o PAUSE	CPU QnA
M9233	SM1233	?	Stato rilevamento errori stazione locale	OFF: Nessun errore ON: Rilevamento errori	
M9235	SM1235	?	Stato rilevamento errori di parametrizzazione stazione locale o stazione I/O remoto	OFF: Nessun errore ON: Rilevamento errori	
M9236	SM1236	?	Stato rilevamento errori di parametrizzazione stazione locale o stazione I/O remoto	OFF: Nessuna comunicazione ON: Comunicazione in corso	
M9237	SM1237	?	Errore stazione locale o stazione I/O remoto	OFF: Normale ON: Anomalo	
M9238	SM1238	?	Errore anello avanti o indietro stazione locale o stazione I/O remoto	OFF: Normale ON: Anomalo	
M9240	SM1240	?	Stato comunicazione	OFF: Online ON: Offline, test stazione-stazione, o test self-loopback	
M9241	SM1241	?	Errore linea anello avanti	OFF: Normale ON: Anomalo	
M9242	SM1242	?	Errore linea anello indietro	OFF: Normale ON: Anomalo	
M9243	SM1243	?	Implementazione loopback	OFF: Loopback non eseguito ON: Loopback in corso	
M9246	SM1246	?	Ricezione dati	OFF: Ricezione ON: Nessuna ricezione	
M9247	SM1247	?	Ricezione dati	OFF: Ricezione ON: Nessuna ricezione	
M9250	SM1250	?	Ricezione parametri	OFF: Ricezione ON: Nessuna ricezione	
M9251	SM1251	?	Relé di comunicazione	OFF: Normale ON: Abortito	
M9252	SM1252	?	Test stato anello	OFF: Non eseguito ON: Test anello avanti o indietro in corso	
M9253	SM1253	?	Stato operativo stazione master	OFF: Stato RUN o STEP RUN ON: Stato STOP o PAUSE	
M9254	SM1254	?	Stato operativo stazione locale diversa da stazione host	OFF: Stato RUN o STEP RUN ON: Stato STOP o PAUSE	
M9255	SM1255	?	Errore stazione locale diversa da stazione host	OFF: Normale ON: Anomalo	

A.4.2 Tabella bit speciali (M) (serie A)

I relé speciali (M) sono relé interni utilizzati per molti scopi diversi, ad es. segnalazione di errore, funzioni speciali ecc. La tabella seguente contiene una panoramica dei relé speciali della serie A MELSEC, insieme ad una breve descrizione delle loro funzioni.

Esistono in generale due tipi di relé speciali:

- Relé speciali pilotati direttamente dalla CPU, che possono solo essere azzerati dall'utente.
- Relé speciali che possono essere attivati o disattivati in determinate condizioni, a seconda della loro funzione.

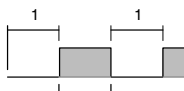
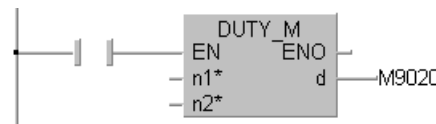
NOTA

L'uso dei relé speciali in un programma di sequenza deve essere controllato adeguatamente.

I relé speciali contrassegnati da ①, ② o ③ nella colonna "Indirizzo" non possono essere attivati o disattivati liberamente. Le spiegazioni relative sono riportate successivamente a questa tabella, a pagina 37.

La possibilità di utilizzo dei relé speciali in funzione di una determinata CPU è riportata nella tabella seguente:

CPU		Significato
		Per tutte le CPU senza restrizioni
○	(CPU AnA)	Non per le CPU indicate
●	CPU A2C	Solo per le CPU specificate

Numero	Significato	Stato	Descrizione	CPU
① M9000	Fusibile interrotto	OFF: Normale ON: Fusibile interrotto	Commutato su ON se esistono fusibili interrotti su una o più unità di uscita. Rimane ON anche dopo che il fusibile è stato ripristinato. Lo stato dei fusibili viene controllato anche sulle stazioni di I/O remoto	○ Utilizzabile con tutti i tipi di CPU. (Per la CPU A2C sono significative solo le informazioni per le stazioni di I/O remoto).
① M9002	Errore verifica modulo di I/O	OFF: Normale ON: Errore	ON se lo stato di I/O è diverso da quello rilevato all'accensione. Rimane ON anche dopo è stato ripristinato lo stato normale. La verifica dei moduli di I/O viene effettuata anche sui moduli delle stazioni di I/O remoto. (Il reset è abilitato solo se i registri speciali da D9116 a D9123 sono azzerati).	○ Utilizzabile con tutti i tipi di CPU. (Per la CPU A2C sono significative solo le informazioni per le stazioni di I/O remoto).
M9004	Errori modulo di comunicazione master MINI	OFF: Normale ON: Errore	Commuta su ON se viene rilevato un errore di comunicazione MINI (S3) su almeno uno dei moduli AJ71PT32 (S3) installati. Rimane ON anche dopo che il fusibile è stato ripristinato.	● Dedicato alle CPU AnA e AnU.
① M9005	Mancanza rete	OFF: Alimentazione presente ON: Alimentazione assente	Commuta su ON in caso di interruzione momentanea della rete di 20 ms o meno. Azzerato spostando l'interruttore POWER da OFF a ON.	○ Utilizzabile con tutti i tipi di CPU.
M9006	Tensione batteria insufficiente	OFF: Normale ON: Tensione batteria insufficiente	Commuta su ON se la tensione di batteria scende sotto al valore nominale. Commuta OFF quando la tensione di batteria torna normale.	○ Utilizzabile con tutti i tipi di CPU.
① M9007	Memoria tensione batteria insufficiente	OFF: Normale ON: Tensione batteria insufficiente	Commuta su ON se la tensione di batteria scende sotto al valore nominale. Rimane ON se la tensione di batteria torna normale.	○ Utilizzabile con tutti i tipi di CPU.
① M9008	Errore di autodiagnosi	OFF: Nessun errore ON: Errore	Si attiva se si verifica un errore dovuto ad autodiagnosi.	○ Utilizzabile con tutti i tipi di CPU.
M9009	Rilevamento spia di errore	OFF: Nessun rilevamento ON: Rilevamento	OD durante l'esecuzione delle istruzioni OUT F e SET F. OFF quando viene azzerato il dato D9124.	○ Utilizzabile con tutti i tipi di CPU.
M9010	Errore di esecuzione	OFF: Nessun errore ON: Errore	Commuta su ON in caso di errore durante l'esecuzione di una istruzione applicativa. Commuta OFF quando viene eliminato l'errore.	○ Non utilizzabile con A3H, A3M, AnA(-F) e AnU.
① M9011	Errore di esecuzione	OFF: Nessun errore ON: Errore	Commuta su ON in caso di errore durante l'esecuzione di una istruzione applicativa. Rimane ON anche dopo che il fusibile è stato ripristinato.	○ Utilizzabile con tutti i tipi di CPU.
M9012	Segnale di riporto	OFF: Riporto OFF ON: Riporto ON	Segnale di riporto utilizzato dalle istruzioni applicative.	○ Utilizzabile con tutti i tipi di CPU.
M9016	Flag cancellazione memoria dati	OFF: Nessuna elaborazione ON: Cancellazione	Cancella la memoria dati compresa la zona retentiva (tranne relé e registri speciali) nel modo RUN remoto da computer ecc., se M9016 è ON.	○ Utilizzabile con tutti i tipi di CPU.
M9017	Flag cancellazione memoria dati	OFF: Nessuna elaborazione ON: Cancellazione	Cancella la memoria dati non retentiva (tranne relé e registri speciali) nel modo RUN remoto da computer ecc., se M9017 è ON.	○ Utilizzabile con tutti i tipi di CPU.
M9020	Temporizzatore utente 0	 <p>1 Scan n2 2 Scan n1</p>	<p>Relé che commuta ripetutamente ON/OFF a intervalli predefiniti. All'accensione o dopo reset, il segnale parte dallo stato OFF. La durata degli stati ON/OFF è definita dall'istruzione DUTY.</p> 	<p>☆ Utilizzabile con tutti i tipi di CPU.</p>
M9021	Temporizzatore utente 1			
M9022	Temporizzatore utente 2			
M9023	Temporizzatore utente 3			
M9024	Temporizzatore utente 4			
② M9025	Richiesta impostazioni dati orologio	OFF: Nessuna elaborazione ON: Richiesta impostazione	Scriva i dati contenuti da D9025 a D9028 negli elementi dell'orologio al termine dell'esecuzione dell'istruzione END durante la scansione in cui M9025 è passato da OFF a ON.	● Non utilizzabile con An, A3H, A3M, A3V, A2C, A52G e A0J2H.
M9026	Dati orologio errati	OFF: Nessun errore ON: Errore	Commutato ON con errore sui dati orologio (D9025 - D9028).	● Non utilizzabile con An, A3H, A3M, A3V, A2C, A52G e A0J2H.

Panoramica dei relé speciali

Numero	Significato	Stato	Descrizione	CPU
M9027	Visualizzazione dati orologio	OFF: Nessuna elaborazione ON: Visualizzazione	Vengono letti i dati dell'orologio nei registri da D9025 a D9028, mentre mese, giorno, ora, minuto e secondo vengono visualizzati sul display a LED del frontalino della CPU.	● Utilizzabile con CPU A3N, A3N-F, A3A, A73 e A3N.
② M9028	Richiesta lettura dati orologio	OFF: Nessuna elaborazione ON: Richiesta di lettura	Legge i dati orologio nei registri D9025 - D9028 in BCD se M9028 è ON.	● Non utilizzabile con An, A3H, A3M, A3V, A2C e AQJ2H.
M9030	Impulso di clock da 0.1 s	OFF: 0,05 s ON: 0,05 s	Vengono generati impulsi di clock da 0,1 secondo, 0,2 secondi, 1 secondo, 2 secondi e 1 minuto. La commutazione non è sincrona con la scansione, ma può avvenire in qualsiasi punto, allo scadere del tempo corrispondente. Parte dallo stato OFF all'accensione o dopo un reset.	☆ Non utilizzabile con A3V
M9031	Impulso di clock da 0.2 s	OFF: 0.1 s ON: 0.1 s		
M9032	Impulso di clock da 1 s	OFF: 0.5 s ON: 0.5 s		
M9033	Impulso di clock da 2 s	OFF: 1 s ON: 1 s		
M9034	Impulso da 1 minuto	OFF: 30 s ON: 30 s		
M9036	Normalmente ON	Sempre ON	Usati come contatti virtuali (dummy) nelle istruzioni di inizializzazione e applicative del programma di sequenza. M9036 e M9037 commutano ON e OFF indipendentemente dalla posizione del commutatore sul frontalino della CPU. M9038 e M9039 vengono attivati e disattivati in funzione dello stato RUN/STOP della CPU. Sono nello stato OFF con il commutatore in posizione STOP. M9038 viene attivato (ON) per una sola scansione, mentre M9039 rimane OFF per una sola scansione, quando il commutatore viene portato su RUN.	☆ Utilizzabile con tutti i tipi di CPU.
M9037	Normalmente OFF	Sempre OFF		
M9038	ON per 1 sola scansione dopo l'ingresso in RUN	ON: Per una scansione dopo RUN OFF: dopo la prima scansione		
M9039	Flag di RUN (OFF solo per 1 scansione dopo RUN)	ON: dopo la prima scansione OFF: per la prima scansione dopo RUN		
M9040	Bobina abilitazione PAUSE	OFF: PAUSE disabilitato ON: PAUSE abilitato	Se il commutatore di RUN è in posizione PAUSE, oppure il contatto remoto di pausa è attivo e M9040 è ON, viene impostato il modo PAUSE e M9041 si attiva.	● Utilizzabile con tutti i tipi di CPU.
M9041	Contatto stato PAUSE	OFF: PAUSE non attivo ON: PAUSE attivo		
M9042	Contatto stato STOP	OFF: Durante STOP ON: Non durante STOP	Sono nello stato ON con il commutatore RUN in posizione STOP.	● Utilizzabile con tutti i tipi di CPU.
M9043	Completamento traccia di campionamento	OFF: Durante la traccia di campionamento ON: Traccia di campionamento completata	Commuta su ON al termine della traccia di campionamento eseguita per il numero di volte determinato dai parametri dell'istruzione STRA. Azzerato durante l'esecuzione dell'istruzione STRA.	○ Non utilizzabile con A1 e A1N
M9044	Traccia campionamento	da OFF a ON: STRA eseguita da ON a OFF: STRAR eseguita	La commutazione ON/OFF di M9044 provoca l'esecuzione delle istruzioni STRA/STRAR. (M9044 viene forzato ON/OFF da un dispositivo periferico). Commutazione da OFF a ON: istruzione STRA Commutazione da ON a OFF: istruzione STRAR Il valore contenuto in D9044 viene usato come condizione per la traccia di campionamento. ("0" corrisponde a campionamento ad ogni passo, con incrementi di 10 ms).	● Non utilizzabile con A1 e A1N
M9046	Traccia campionamento	OFF: Tranne durante la traccia ON: Durante la traccia	Commuta su ON durante la traccia di campionamento.	○ Non utilizzabile con A1 e A1N
M9047	Preparazione traccia campionamenti	OFF: Traccia campionamenti arrestata ON: Traccia di campionamento avviata	Commuta su ON per avviare la traccia di campionamento. Commuta su OFF per arrestare la traccia di campionamento.	○ Non utilizzabile con A1 e A1N
M9049	Selezione del numero di caratteri inviati	OFF: Emissione fino al codice NUL. ON: Emissione di 16 caratteri.	Se M9049 è OFF, vengono emessi tutti i caratteri fino al codice NUL (00 _H). Se M99049 è ON, vengono emessi 16 caratteri ASCII.	○ Utilizzabile con An, A3V, A2C e A52G
② M9050	Contatto scambio memoria risultati operazione (per istruzione CHG)	OFF: Nessuno scambio ON: Scambio	Commutato su ON per scambiare i dati della memoria risultati operazione con i dati dell'area di salvataggio (per i dettagli, fare riferimento alla sezione 7.6.8).	● Specifico per A3

Numero	Significato	Stato	Descrizione	CPU
M9051	Disabilitazione esecuzione istruzione CHG	OFF: Disabilitazione ON: Abilitazione	ON per disabilitare l'istruzione CHG. Commuta su ON alla richiesta di trasferimento e commutato automaticamente su OFF al completamento del trasferimento.	● Utilizzabile con CPU A3, A3N(-F), A3H, A3M, A3V, A3A(-F), A3U, A4U, A73 e A3N
M9052	Commutazione istruzione SEG	OFF: Display 7 segmenti ON: Rinfresco parziale I/O	ON per usare l'istruzione SEG come istruzione per rinfresco parziale degli I/O. OFF per usare l'istruzione SEG come istruzione per display a 7 segmenti (per i dettagli, vedere le sezioni 6.7.2 e 7.5.5).	○ Non utilizzabile con CPU An, A3V e A3N
M9053	Commutazione istruzioni EI/DI	OFF: Controllo interruzione sequenza ON: Controllo interruzione di comunicazione	Commutato su ON per eseguire le istruzioni di abilitazione e disabilitazione del rinfresco dei dati di comunicazione. (per i dettagli, vedere le sezioni 6.6.1 e 6.7.4).	● Non utilizzabile con An, A3H, A3M, AnA, AnA-F, AnU e A3V
M9054	Flag STEP RUN	OFF: Diverso da STEP RUN ON: Durante STEP RUN	Commutato ON con il commutatore RUN in posizione STEP RUN.	○ Non utilizzabile con A1S, A2C, A0J2H e A52G
M9055	Flag completamento memorizzazione stato	OFF: Non completata ON: Completata	Commuta su ON al completamento della memorizzazione di stato. Azzerato a reset.	○ Non utilizzabile con A1 e A1N.
M9056	Programma principale richiesta impostazione P, I	OFF: Nessuna richiesta ON: Richiesta impostazione P, I	Consente la richiesta di impostazione P e I al termine del trasferimento dell'altro programma (ad esempio un sottoprogramma mentre il programma principale è in esecuzione) in stato RUN. Commuta automaticamente su OFF al termine dell'impostazione P, I.	● Utilizzabile con CPU A3, A3N, A3N-F, A3H, A3M, A3V, A3A, A3A-F, A73, A3U, A4U e A3N
M9057	Sottoprogramma 1 richiesta impostazione P, I	OFF: Nessuna richiesta impostazione P, I ON: Richiesta impostazione P, I		
M9060	Sottoprogramma 2 richiesta impostazione P, I			
M9061	Sottoprogramma 3 richiesta impostazione P, I			
M9060	Errore terminale remoto	OFF: Normale ON: Errore	Commuta ON se uno dei terminali remoti corrisponde a una stazione in avaria. L'errore di comunicazione viene rilevato durante la normale comunicazione non viene ripristinata dopo il numero di tentativi impostati in D9174. Commuta su OFF se viene ripristinata la comunicazione con tutti i terminali remoti, se abilitato il ripristino automatico online. Rimane ON se il ripristino automatico online è disabilitato. Nessuna commutazione se il rilevamento errori di comunicazione è sospeso.	● Utilizzabile con A2C e A52G
M9061	Errore di comunicazione	OFF: Normale ON: Errore	Commuta ON in caso di errore di comunicazione con moduli terminali remoti o I/O remoto. Errore di comunicazione per una delle cause seguenti. Errore inizializzazione Rottura cavo Mancanza alimentazione su moduli terminali remoti o I/O remoto. Commuta su OFF se la comunicazione viene ripristinata, con ripristino automatico online abilitato. Rimane ON con rilevamento errori di comunicazione sospeso e ripristino automatico online disabilitato.	● Utilizzabile con A2C e A52G
M9065	Stato trasferimento a pacchetti	OFF: Durante l'esecuzione di altre istruzioni ON: Trasferimento a pacchetti	Il flag è impostato ON durante il trasferimento a pacchetti delle maschere video su AD57(S1)/AD58 e azzerato al termine del trasferimento.	● Utilizzabile con CPU AnA(-F) e AnU.
M9066	Commutazione modo di trasferimento	OFF: Trasferimento in blocco ON: Trasferimento a pacchetti	Commutato ON quando l'aggiornamento delle maschere video su AD57(S1)/AD58 viene eseguito con trasferimento a pacchetti.	● Utilizzabile con CPU AnA(-F) e AnU.
M9067	Rilevamento errori modulo di I/O	OFF: Normale ON: Errore	ON se uno dei moduli I/O è in avaria. L'errore di comunicazione viene rilevato durante la normale comunicazione non viene ripristinata dopo il numero di tentativi impostati in D9174. Commuta su OFF se viene ripristinata la comunicazione con tutti i moduli di I/O, se abilitato il ripristino automatico online. Rimane ON se il ripristino automatico online è disabilitato. Nessuna commutazione se il rilevamento errori di comunicazione è sospeso.	● Utilizzabile con A2C e A52G.

Panoramica dei relé speciali

Numero	Significato	Stato	Descrizione	CPU
M9068	Modo test	OFF: Ripristino automatico online abilitato Ripristino automatico online disabilitato Sospensione comunicazione con errore online Controllo linea ON:	ON se viene eseguito il controllo della linea con moduli di I/O e terminali remoti. OFF se la linea funziona in modo normale.	● Utilizzabile con A2C e A52G.
M9069	Uscite con errore di rete	OFF: Tutte le uscite disabilitate ON: Stato delle uscite conservato	Impostare per determinare il comportamento delle uscite in caso di errore di comunicazione. OFF: Tutte le uscite vengono disabilitate con errori di comunicazione. ON Viene conservato lo stato delle uscite precedente all'errore.	● Utilizzabile con A2C e A52G.
M9081	Richiesta di comunicazione con terminali remoti	OFF: Richiesta di comunicazione con terminali remoti abilitata ON: Richiesta di comunicazione con terminali disabilitata	Indicazione della abilitazione/disabilitazione della comunicazione con terminali remoti collegati con AJ71PT32-S3, A2C o A52G.	● Utilizzabile con AnA, AnA-F, A2C e A52G
M9082	Corrispondenza numero di stazioni	OFF: Corrispondenza numero di stazioni ON: Mancata corrispondenza numero di stazioni	Commuta su ON se il numero di terminali remoti e moduli di I/O remoto collegati con A2C o A52G non corrisponde al numero totale di stazioni impostato in configurazione. OFF se il numero di stazioni corrisponde al numero totale rilevato durante la transizione STOP →RUN.	● Specifico per A2C e A52G.
M9084	Controllo errori	OFF: Controllo abilitato ON: Controllo disabilitato	Specifica se devono essere controllati gli errori seguenti, dopo l'esecuzione dell'istruzione END (per ridurre il tempo di esecuzione di END): Fusibile interrotto Errore verifica modulo di I/O Errore batteria	○ Non utilizzabile su An, A2C e A3V.
M9086	Programma BASIC Flag di RUN	OFF: A3M-BASIC stop ON: A3M-BASIC RUN	Attivo se A3M-BASIC è in stato RUN, azzerato se in stato STOP.	● Specifico per A3M.
M9087	Programma BASIC Flag di PAUSE	OFF: A3M-BASIC RUN abilitato ON: A3M-BASIC RUN disabilitato	Abilita/disabilita l'esecuzione di A3M-BASIC quando la CPU A3M è in stato PAUSE. OFF: A3M-BASIC in esecuzione ON: A3M-BASIC non in esecuzione	● Specifico per A3M.
M9089	Uscita su morsetto ERR	OFF: nessun segnale su uscita ERR ON: segnale su uscita ERR	Il relé interno è attivo se il programma di sequenza emette un segnale sui morsetti ERR. Il relé può essere disattivato solo se i relé M9089 e M9090 vengono azzerati contemporaneamente.	● Specifico per CPU A2C
M9090	Uscita su morsetto ERR	OFF: nessun segnale su uscita ERR ON: segnale su uscita ERR	Il relé interno si attiva in caso di errore su MELSECNET/MINI o nel programma di sequenza (a fine elaborazione). Il relé viene azzerato quando l'errore sulla rete è stato eliminato o il programma ripristinato.	● Specifico per CPU A2C
① M9091	Flag dettaglio errore di esecuzione	OFF: Nessun errore ON: Errore	Attivo se una indicazione dettagliata di errore è stata memorizzata in D9091; rimane attivo anche dopo il ripristino dello stato normale.	● Utilizzabile con AnA(-F) e AnU.
	Flag errore di chiamata subroutine microcomputer	OFF: Nessun errore ON: Errore	Attivo se si riscontra un errore nell'esecuzione del programma microcomputer; rimane attivo anche dopo il ripristino del funzionamento normale.	○ Non utilizzabile su AnA(-F) e AnU.
②③ M9094	Flag sostituzione I/O	OFF: Sostituzione ON: Nessuna sostituzione	Dopo aver impostato l'indirizzo iniziale di I/O in D9094, l'attivazione di M9094 consente la sostituzione al volo del modulo. (È possibile la sostituzione di un solo modulo alla volta). Commutare a ON da programma o da terminale di programmazione in modo test, per sostituire il modulo con CPU in RUN. Commutare a ON da terminale di programmazione in modo test, per sostituire il modulo con CPU in STOP. Il modo RUN/STOP non deve essere modificato prima del completamento della sostituzione del modulo.	○ Non utilizzabile con CPU An, A3H, A3V, A0J2H, A2C, A52G e A3N.

NOTA

Tutti i relé speciali vengono azzerati dopo aver tolto l'alimentazione, dopo una cancellazione memoria o RESET.

Se il commutatore RUN viene portato su STOP, il contenuto dei relé viene mantenuto.

I relé speciali contrassegnati con ❶ rimangono attivi anche se viene ripristinato lo stato normale. Possono essere disattivati come segue:

- Inserire una riga nel programma di sequenza per azzerare i relé speciali con una istruzione RST condizionata opportunamente.
- Forzare un RESET con il terminale di programmazione.
- Azzerare la CPU portando l'interruttore della CPU su RESET.

I relé speciali contrassegnati ❷ possono essere modificati solo con istruzioni del programma.

I relé speciali contrassegnati ❸ vengono modificati tramite terminale di programmazione in modo test..

Panoramica dei relé speciali

A.4.3 Tabella bit di comunicazione (solo serie A)

I relé di comunicazione sono relé interni che vengono attivati o disattivati durante la comunicazione dati in una rete a seconda di diverse condizioni. Il loro stato varia a seguito di un errore nell'esecuzione del programma.

L'elaborazione dei relé di comunicazione è diversa a seconda che la CPU sia inserita in una stazione master o locale.

Relé di comunicazione nella stazione master

Numero	Significato	Stato	Descrizione
M9200	Ricezione istruzione LRDP	OFF: Non ricevuta ON: Ricevuta	Segnala lo stato di ricezione dell'istruzione LRDP (lettura operandi a word). Usato nel programma come interblocco per l'istruzione LRDP. Usare l'istruzione RST per azzerarlo.
M9201	Completamento istruzione LRDP	OFF: Non completa ON: Completata	Segnala lo stato di completamento esecuzione dell'istruzione LRDP (lettura operandi a word). Usato come condizione d'ingresso per azzerare M9200 e M9201 al termine dell'istruzione LRDP. Usare l'istruzione RST per azzerarlo.
M9202	Ricezione istruzione LWTP	OFF: Non ricevuta ON: Ricevuta	Segnala lo stato di ricezione dell'istruzione LWTP (scrittura operandi a word). Usato nel programma come interblocco per l'istruzione LWTP. Usare l'istruzione RST per azzerarlo.
M9203	Completamento istruzione LWTP	OFF: Non completa ON: Completa	Segnala lo stato di completamento esecuzione dell'istruzione LWTP (scrittura operandi a word). Usato come condizione d'ingresso per azzerare M9202 e M9203 al termine dell'istruzione LWTP. Usare l'istruzione RST per azzerarlo.
M9206	Errore nei parametri di comunicazione della stazione host	OFF: Normale ON: Errore	Segnala la validità o meno dei parametri di comunicazione della stazione host.
M9207	Parametri di comunicazione non corrispondenti fra stazioni master	OFF: Normale ON: Non corrispondenti	Segnala la corrispondenza o meno fra i parametri di comunicazione della stazione master di livello 2 con quelli della stazione master di livello 3, in un sistema a tre livelli. Valido solo per le stazioni master di un sistema a tre livelli).
M9208	Imposta i campi di trasmissione B e W delle stazioni master di livello inferiore.	OFF: Trasmissione a livello 2 e livello 3 ON: Trasmissione a livello 2 solamente	Il relé interno indica se i dati di comunicazione in B e W vengono trasferiti dalla stazione master del primo livello verso le stazioni dei livelli inferiori (sotto stazioni). I dati vengono trasferiti solo se M9208 non è attivo.
M9209	Comando controllo parametri di comunicazione per stazioni master di livello inferiore.	OFF: Controllo ON: Nessun controllo	Il relé speciale viene attivato se gli operandi di comunicazione (B e W) del primo livello non devono essere confrontati con gli operandi corrispondenti del livello inferiore. Se M9209 non è attivo, gli operandi di collegamento fra il livello superiore e i livelli inferiori vengono controllati continuamente.
M9210	Errore scheda di comunicazione	OFF: Normale ON: Errore	Segnala presenza o assenza di un errore hardware della scheda di comunicazione. Sorvegliato dalla CPU.
M9224	Stato comunicazione	OFF: Online ON: Offline, test da stazione a stazione o test self-loopback	Segnala se la stazione master è online o offline, oppure sta eseguendo il test stazione-stazione o loopback.
M9225*	Errore anello avanti	OFF: Normale ON: Errore	Segnala la condizione di errore della linea dell'anello avanti.
M9226*	Errore anello indietro	OFF: Normale ON: Errore	Segnala la condizione di errore della linea dell'anello indietro.
M9227*	Test stato anello	OFF: Non eseguito ON: Test anello avanti o indietro eseguito	Segnala se la stazione master sta eseguendo un test sull'anello avanti o indietro.
M9232	Stato operativo stazione locale	OFF: Modo RUN o STEP RUN ON: Modo STOP o PAUSE	Segnala se la stazione locale è in modo STOP o PAUSE.

Relé di comunicazione nella stazione master

Numero	Significato	Stato	Descrizione
M9233	Rilevamento errori stazione locale	OFF:Nessun errore ON: Errore rilevato	Segnala se la stazione locale ha rilevato un errore in un'altra stazione.
M9235	Rilevato errore di parametrizzazione della stazione locale o di I/O remoto	OFF:Nessun errore ON:Errore rilevato	Segnala se la stazione locale o di I/O remoto ha rilevato un errore di parametrizzazione nella stazione master.
M9236	Stato comunicazione iniziale della stazione locale o di I/O remoto	OFF:Nessuna comunicazione ON:Comunicazione	Segnala se la stazione locale o di I/O remoto sta comunicando i dati di inizializzazione (ad es. parametri) con la stazione master.
M9237	Errore stazione locale o di I/O remoto	OFF:Normale ON:Errore	Segnala la condizione di errore di una stazione locale o di I/O remoto.
M9238*	Errore anello avanti/indietro stazione locale o di I/O remoto	OFF:Normale ON:Errore	Segnala la condizione di errore delle linee degli anelli avanti e indietro di una stazione locale o di I/O remoto.

* I relé speciali contrassegnati non possono essere utilizzati con MELSECNET/B.

Panoramica dei relé speciali

Relé di comunicazione nella stazione locale

Numero	Significato	Stato	Descrizione
M9204	Istruzione LRDP completata	OFF: Non completa ON: Completa	ON indica che l'istruzione LRDP è stata completata nella stazione locale.
M9205	Istruzione LWTP completata	OFF: Non completa ON: Completata	ON indica che l'istruzione LWTP è stata completata nella stazione locale.
M9211	Errore scheda di comunicazione (stazione locale)	OFF: Normale ON: Errore	Segnala presenza o assenza di un errore hardware della scheda di comunicazione. Sorvegliato dalla CPU.
M9240*	Stato comunicazione	OFF: Online ON: Offline, test stazione-stazione, o test self-loopback	Segnala se la stazione locale è online o offline, oppure sta eseguendo il test stazione-stazione o loopback.
M9241*	Errore anello avanti	OFF: Normale ON: Errore	Segnala la condizione di errore della linea dell'anello avanti.
M9242*	Errore anello indietro	OFF: Normale ON: Errore	Segnala la condizione di errore della linea dell'anello indietro.
M9243	Esecuzione loopback	OFF: non eseguito ON: eseguito	Segnala se la stazione locale sta eseguendo o meno il loopback.
M9246	Ricezione dati	OFF: Ricevuta ON: Non ricevuti	Segnale se i dati sono stati ricevuti dalla stazione master.
M9247	Ricezione dati	OFF: Ricevuta ON: Non ricevuti	Segnala se una stazione di livello tre ha ricevuto dati dalla sua stazione master in un sistema a tre livelli.
M9250	Ricezione parametri	OFF: Ricevuta ON: Non ricevuti	Segnale se i parametri di comunicazione sono stati ricevuti dalla stazione master.
M9251	Collegamento interrotto	OFF: Normale ON: Interrotto	Segnala lo stato del collegamento dati sulla stazione locale.
M9252	Test stato anello	OFF: Non eseguito ON: Test anello avanti o indietro in esecuzione	Segnala se la stazione locale sta eseguendo un test sull'anello avanti o indietro.
M9253	Stato operativo stazione master	OFF: Modo RUN o STEP RUN ON: Modo STOP o PAUSE	Segnala se la stazione master è in modo STOP o PAUSE.
M9254	Stato operativo delle altre stazioni locali	OFF: Modo RUN o STEP RUN ON: Modo STOP o PAUSE	Segnala se una stazione locale diversa dalla stazione host è in modo STOP o PAUSE.
M9255	Stato di errore nelle altre stazioni locali	OFF: Normale ON: Errore	Segnala se una delle altre stazioni locali diverse dalla stazione host presenta un errore.

* I relé speciali contrassegnati non possono essere utilizzati con MELSECNET/B.

A.5 Tabella dei registri speciali

A.5.1 Tabella dei bit speciali (MELSEC serie Q e System Q)

I registri speciali sono registri interni con funzione fissa del controllore programmabile.

Quindi i relé speciali non possono essere utilizzati come gli altri relé interni in un programma di sequenza. È comunque possibile scrivere dati in questi registri per controllare la CPU Q/QnA. I dati vengono normalmente scritti in binario a meno che non sia necessario un formato diverso.

NOTA

I registri speciali da SD1200 a SD1255 vengono usati nelle CPU QnA. Questi registri sono assenti nelle CPU System Q.

I registri speciali a partire da SD1500 sono dedicati per la CPU Q4AR.

Nella tabella seguente vengono spiegati i significati delle colonne delle tabelle alle pagine successive:

Caratteristica	Significato
Numero	Indica l'indirizzo del registro speciale.
Nome	Indica il nome del registro speciale.
Significato	Contiene una breve descrizione della funzione del registro speciale.
Descrizione	Contiene una descrizione dettagliata della funzione del registro.
Impostato da (se impostato)	<p>Indica se il relé speciale viene impostato dal sistema o dall'utente.</p> <p><Impostato da></p> <p>S : Impostato dal sistema</p> <p>U : Impostato dall'utente (tramite programma di sequenza o terminale di programmazione in modo test)</p> <p>S/U : Impostato dal sistema o utente</p> <p>Indicato solo se lo stato è imposto dal sistema.</p> <p><se impostato></p> <p>Elaborazione END : Impostato durante l'elaborazione dell'istruzione END</p> <p>Inizializzazione : Impostato durante il processo di inizializzazione (accensione, STOP->RUN)</p> <p>Cambiamento di stato : Impostato dopo un cambiamento di stato</p> <p>Errore : Impostato dopo un errore</p> <p>Esecuzione istruzione : Impostato durante l'esecuzione di una istruzione</p> <p>Richiesta : Impostato per richiesta dell'utente (tramite SM, ecc.)</p>
Registro CPU A corrispondente D9 [] [] []	<p>Indica il registro speciale M9 [] [] [] corrispondente alla CPU serie A (modifica e annotazione se il contenuto viene modificato).</p> <p>Le voci contrassegnate con "Nuovo" sono state aggiunte per le CPU serie Q/System Q.</p>
Valido per:	<p>Indica la CPU corrispondente:</p> <p>●: Utilizzabile da tutte le CPU</p> <p>Q CPU: utilizzabile da CPU System Q</p> <p>CPU QnA: utilizzabile da CPU serie QnA e Q2AS</p> <p>Nome CPU: utilizzabile solo dalla CPU indicata (ad es. CPU Q4AR)</p> <p>Rem: utilizzabile da un modulo di I/O remoto MELSECNET/H</p>

Per informazioni dettagliate sugli argomenti seguenti, fare riferimento ai manuali:

- Reti → Melsecnet/10 Manuale di riferimento sistema di rete per QnA
- SFC → Manuale di programmazione CPU QnA (SFC)

Tabella dei registri speciali

Tabella dei registri speciali

(1) Informazioni diagnostiche

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valido per:		
SD0	Errori diagnostici	Codice di errore	I codici di errore relativi agli errori rilevati dalla diagnostica vengono memorizzati come dati BIN. Il contenuto corrisponde alle informazioni più aggiornate dello storico errori.	S (Errore)	D9008 modifica formato			
SD1	Tempo orologio per occorrenza errore diagnostica	Tempo orologio per occorrenza errore diagnostica	I dati anno (ultime due cifre) e mese in cui SD0 è stato aggiornato, memorizzati come codice BCD a 2 cifre. Esempio: Ottobre 1995 H9510 b15 b8 b7 b0 Year (0 to 99) Month (1 to 31)	S (Errore)	Nuovo			
SD2			Giorno e ora in cui SD0 è stato aggiornato sono memorizzati come codice BCD a 2 cifre. Esempio: 10 pomeridiane del giorno 25 H2510 b15 b8 b7 b0 <table border="1" style="width: 100%; border-collapse: collapse;"><tr><td style="width: 50%;">Day (1 to 31)</td><td style="width: 50%;">Hour (0 to 23)</td></tr></table>				Day (1 to 31)	Hour (0 to 23)
Day (1 to 31)			Hour (0 to 23)					
SD3	Minuto e secondo in cui SD0 è stato aggiornato sono memorizzati come codice BCD a 2 cifre. Esempio: 35 minuti e 48 secondi H3548 b15 b8 b7 b0 <table border="1" style="width: 100%; border-collapse: collapse;"><tr><td style="width: 50%;">Minute (1 to 60)</td><td style="width: 50%;">Second (1 to 60)</td></tr></table>	Minute (1 to 60)	Second (1 to 60)					
Minute (1 to 60)	Second (1 to 60)							
SD4	Categorie delle informazioni di errore	Codice categoria informazioni di errore	I codici di categoria aiutano a determinare il tipo di informazione memorizzata nelle aree di informazione generale (da SD5 a SD15) e le aree di informazione specifica (da SD16 a SD26) sono memorizzati in questo registro. b15 b8 b7 b0 <table border="1" style="width: 100%; border-collapse: collapse;"><tr><td style="width: 50%;">Individual error info.</td><td style="width: 50%;">Common error info.</td></tr></table> I codici di categoria delle informazioni generali sono i seguenti: 0: Nessun errore 1: Numero uniti/modulo 2: Nome file/nome drive 3: Tempo (valore impostato) 4: Posizione errore programma I codici di categoria delle informazioni specifiche sono i seguenti: 0: Nessun errore 1: (Apertura) 2: Nome file/nome drive 3: Tempo (valore effettivo) 4: Posizione errore programma 5: Numero parametro 6: Numero spia di errore 7: Numero malfunzionamento istruzione di controllo	Individual error info.	Common error info.	S (Errore)	Nuovo	
Individual error info.	Common error info.							

Tabella dei registri speciali (continua)

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valido per:																																																																																																																	
SD5	Informazioni di errore generali	Informazioni di errore generali	<p>Le informazioni generali corrispondenti ai codici di errore (SD0) sono contenute in questo registro. Vengono memorizzati i seguenti quattro tipi di informazione:</p> <p>(1) Numero unità/modulo</p> <table border="1" style="margin: 10px auto; border-collapse: collapse;"> <thead> <tr style="background-color: #d3d3d3;"> <th style="width: 50px;">Numero</th> <th style="width: 150px;">Significato</th> </tr> </thead> <tbody> <tr><td>SD5</td><td>Numero stazione / modulo</td></tr> <tr><td>SD6</td><td>Indirizzo di I/O</td></tr> <tr><td>SD7</td><td rowspan="10" style="text-align: center;">Non usato</td></tr> <tr><td>SD8</td></tr> <tr><td>SD9</td></tr> <tr><td>SD10</td></tr> <tr><td>SD11</td></tr> <tr><td>SD12</td></tr> <tr><td>SD13</td></tr> <tr><td>SD14</td></tr> <tr><td>SD15</td></tr> </tbody> </table> <p>(2) Nome file/nome drive Esempio: Nome file = ABCDEFGH.IJK</p> <table border="1" style="margin: 10px auto; border-collapse: collapse;"> <thead> <tr style="background-color: #d3d3d3;"> <th style="width: 50px;">Numero</th> <th style="width: 150px;">Significato</th> <th style="width: 50px;">b15</th> <th style="width: 50px;">b0</th> </tr> </thead> <tbody> <tr><td>SD5</td><td>Numero</td><td></td><td></td></tr> <tr><td>SD6</td><td>Drive</td><td>B</td><td>A</td></tr> <tr><td>SD7</td><td>Nome file</td><td>D</td><td>C</td></tr> <tr><td>SD8</td><td>Codice ASCII: 8 caratteri</td><td>F</td><td>E</td></tr> <tr><td>SD9</td><td></td><td>H</td><td>G</td></tr> <tr><td>SD10</td><td>2E_H(.)</td><td>I</td><td>.</td></tr> <tr><td>SD11</td><td>Espansione</td><td>K</td><td>J</td></tr> <tr><td>SD12</td><td>Codice ASCII: 3 caratteri</td><td></td><td></td></tr> <tr><td>SD13</td><td></td><td></td><td></td></tr> <tr><td>SD14</td><td>Non usato</td><td></td><td></td></tr> <tr><td>SD15</td><td></td><td></td><td></td></tr> </tbody> </table> <p>(3) Tempo (valore impostato)</p> <table border="1" style="margin: 10px auto; border-collapse: collapse;"> <thead> <tr style="background-color: #d3d3d3;"> <th style="width: 50px;">Numero</th> <th style="width: 150px;">Significato</th> </tr> </thead> <tbody> <tr><td>SD5</td><td>Tempo: unità da 1µs (da 0 a</td></tr> <tr><td>SD6</td><td>Tempo: unità da 1ms (da 0 a</td></tr> <tr><td>SD7</td><td rowspan="10" style="text-align: center;">Non usato</td></tr> <tr><td>SD8</td></tr> <tr><td>SD9</td></tr> <tr><td>SD10</td></tr> <tr><td>SD11</td></tr> <tr><td>SD12</td></tr> <tr><td>SD13</td></tr> <tr><td>SD14</td></tr> <tr><td>SD15</td></tr> </tbody> </table> <p>(4) Posizione errore programma</p> <table border="1" style="margin: 10px auto; border-collapse: collapse;"> <thead> <tr style="background-color: #d3d3d3;"> <th style="width: 50px;">Numero</th> <th style="width: 150px;">Significato</th> </tr> </thead> <tbody> <tr><td>SD5</td><td rowspan="4" style="text-align: center;">Nome file (Codice ASCII: 8 caratteri)</td></tr> <tr><td>SD6</td></tr> <tr><td>SD7</td></tr> <tr><td>SD8</td></tr> <tr><td>SD9</td><td>Espansione 2E_H(.)</td></tr> <tr><td>SD10</td><td>(Codice ASCII: 3 caratteri)</td></tr> <tr><td>SD11</td><td>Maschera*</td></tr> <tr><td>SD12</td><td>Numero blocco</td></tr> <tr><td>SD13</td><td>Numero passo / transizione</td></tr> <tr><td>SD14</td><td>Num. passo sequenza (L)</td></tr> <tr><td>SD15</td><td>Num. passo sequenza (H)</td></tr> </tbody> </table> <p>* Contenuto della</p> <table border="1" style="margin: 10px auto; border-collapse: collapse;"> <tr> <td style="text-align: center;">15 14</td> <td style="text-align: center;">-- --</td> <td style="text-align: center;">4 3</td> <td style="text-align: center;">2 1</td> <td style="text-align: center;">0</td> <td style="text-align: center;">← (Bit No.)</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> </tr> </table> <p style="font-size: small; margin-top: 5px;"> Non utilizzato ┌───┐ Designazione blocco SFC presente (1) / assente (0) └───┘ Designazione passo SFC presente (1) / assente (0) └───┘ Designazione transizione SFC presente (1) / assente (0) </p>	Numero	Significato	SD5	Numero stazione / modulo	SD6	Indirizzo di I/O	SD7	Non usato	SD8	SD9	SD10	SD11	SD12	SD13	SD14	SD15	Numero	Significato	b15	b0	SD5	Numero			SD6	Drive	B	A	SD7	Nome file	D	C	SD8	Codice ASCII: 8 caratteri	F	E	SD9		H	G	SD10	2E _H (.)	I	.	SD11	Espansione	K	J	SD12	Codice ASCII: 3 caratteri			SD13				SD14	Non usato			SD15				Numero	Significato	SD5	Tempo: unità da 1µs (da 0 a	SD6	Tempo: unità da 1ms (da 0 a	SD7	Non usato	SD8	SD9	SD10	SD11	SD12	SD13	SD14	SD15	Numero	Significato	SD5	Nome file (Codice ASCII: 8 caratteri)	SD6	SD7	SD8	SD9	Espansione 2E _H (.)	SD10	(Codice ASCII: 3 caratteri)	SD11	Maschera*	SD12	Numero blocco	SD13	Numero passo / transizione	SD14	Num. passo sequenza (L)	SD15	Num. passo sequenza (H)	15 14	-- --	4 3	2 1	0	← (Bit No.)	0	0	0	0	0	0	S (Errore)	Nuovo	●
Numero				Significato																																																																																																																			
SD5				Numero stazione / modulo																																																																																																																			
SD6				Indirizzo di I/O																																																																																																																			
SD7				Non usato																																																																																																																			
SD8																																																																																																																							
SD9																																																																																																																							
SD10																																																																																																																							
SD11																																																																																																																							
SD12																																																																																																																							
SD13																																																																																																																							
SD14																																																																																																																							
SD15																																																																																																																							
Numero					Significato	b15	b0																																																																																																																
SD5				Numero																																																																																																																			
SD6	Drive	B	A																																																																																																																				
SD7	Nome file	D	C																																																																																																																				
SD8	Codice ASCII: 8 caratteri	F	E																																																																																																																				
SD9		H	G																																																																																																																				
SD10	2E _H (.)	I	.																																																																																																																				
SD11	Espansione	K	J																																																																																																																				
SD12	Codice ASCII: 3 caratteri																																																																																																																						
SD13																																																																																																																							
SD14	Non usato																																																																																																																						
SD15																																																																																																																							
Numero	Significato																																																																																																																						
SD5	Tempo: unità da 1µs (da 0 a																																																																																																																						
SD6	Tempo: unità da 1ms (da 0 a																																																																																																																						
SD7	Non usato																																																																																																																						
SD8																																																																																																																							
SD9																																																																																																																							
SD10																																																																																																																							
SD11																																																																																																																							
SD12																																																																																																																							
SD13																																																																																																																							
SD14																																																																																																																							
SD15																																																																																																																							
Numero		Significato																																																																																																																					
SD5	Nome file (Codice ASCII: 8 caratteri)																																																																																																																						
SD6																																																																																																																							
SD7																																																																																																																							
SD8																																																																																																																							
SD9	Espansione 2E _H (.)																																																																																																																						
SD10	(Codice ASCII: 3 caratteri)																																																																																																																						
SD11	Maschera*																																																																																																																						
SD12	Numero blocco																																																																																																																						
SD13	Numero passo / transizione																																																																																																																						
SD14	Num. passo sequenza (L)																																																																																																																						
SD15	Num. passo sequenza (H)																																																																																																																						
15 14	-- --	4 3	2 1	0	← (Bit No.)																																																																																																																		
0	0	0	0	0	0																																																																																																																		

Tabella dei registri speciali

Tabella dei registri speciali (continua)

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valid o per																																																																				
Significato estensione:																																																																										
<table border="1"> <thead> <tr> <th>SD10 (SD9)</th> <th colspan="2">SD11 (SD10)</th> <th rowspan="2">Extension name</th> <th rowspan="2">File type</th> </tr> <tr> <th>Higher byte</th> <th>Lower byte</th> <th>Higher byte</th> </tr> </thead> <tbody> <tr> <td>51H</td> <td>50H</td> <td>41H</td> <td>QPA</td> <td>Parameters</td> </tr> <tr> <td>51H</td> <td>50H</td> <td>47H</td> <td>QPG</td> <td>Sequence program</td> </tr> <tr> <td>51H</td> <td>43H</td> <td>44H</td> <td>QCD</td> <td>Device comment</td> </tr> <tr> <td>51H</td> <td>44H</td> <td>49H</td> <td>QDI</td> <td>Device initial value</td> </tr> <tr> <td>51H</td> <td>44H</td> <td>52H</td> <td>QDR</td> <td>File register</td> </tr> <tr> <td>51H</td> <td>44H</td> <td>53H</td> <td>QDS</td> <td>Simulation data</td> </tr> <tr> <td>51H</td> <td>44H</td> <td>4CH</td> <td>QDL</td> <td>Local device</td> </tr> <tr> <td>51H</td> <td>54H</td> <td>53H</td> <td>QTS</td> <td>Sampling trace data (QnA-CPU only)</td> </tr> <tr> <td>51H</td> <td>54H</td> <td>4CH</td> <td>QTL</td> <td>Status latch data (QnA-CPU only)</td> </tr> <tr> <td>51H</td> <td>54H</td> <td>50H</td> <td>QTP</td> <td>Program trace data (QnA-CPU only)</td> </tr> <tr> <td>51H</td> <td>54H</td> <td>52H</td> <td>QTR</td> <td>SFC trace file</td> </tr> <tr> <td>51H</td> <td>46H</td> <td>44H</td> <td>QFD</td> <td>Trouble history data</td> </tr> </tbody> </table>							SD10 (SD9)	SD11 (SD10)		Extension name	File type	Higher byte	Lower byte	Higher byte	51H	50H	41H	QPA	Parameters	51H	50H	47H	QPG	Sequence program	51H	43H	44H	QCD	Device comment	51H	44H	49H	QDI	Device initial value	51H	44H	52H	QDR	File register	51H	44H	53H	QDS	Simulation data	51H	44H	4CH	QDL	Local device	51H	54H	53H	QTS	Sampling trace data (QnA-CPU only)	51H	54H	4CH	QTL	Status latch data (QnA-CPU only)	51H	54H	50H	QTP	Program trace data (QnA-CPU only)	51H	54H	52H	QTR	SFC trace file	51H	46H	44H	QFD	Trouble history data
SD10 (SD9)	SD11 (SD10)		Extension name	File type																																																																						
Higher byte	Lower byte	Higher byte																																																																								
51H	50H	41H	QPA	Parameters																																																																						
51H	50H	47H	QPG	Sequence program																																																																						
51H	43H	44H	QCD	Device comment																																																																						
51H	44H	49H	QDI	Device initial value																																																																						
51H	44H	52H	QDR	File register																																																																						
51H	44H	53H	QDS	Simulation data																																																																						
51H	44H	4CH	QDL	Local device																																																																						
51H	54H	53H	QTS	Sampling trace data (QnA-CPU only)																																																																						
51H	54H	4CH	QTL	Status latch data (QnA-CPU only)																																																																						
51H	54H	50H	QTP	Program trace data (QnA-CPU only)																																																																						
51H	54H	52H	QTR	SFC trace file																																																																						
51H	46H	44H	QFD	Trouble history data																																																																						

Tabella dei registri speciali (continua)

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valido per																																																						
SD16			<p>Le informazioni specifiche corrispondenti ai codici di errore (SD0) sono contenute in questo registro. Vengono memorizzati i seguenti sei tipi di informazione:</p> <p>(1) Nome file/nome drive</p> <div style="display: flex; justify-content: space-between;"> <table border="1" style="font-size: small;"> <thead> <tr><th>Numero</th><th>Significato</th></tr> </thead> <tbody> <tr><td>SD16</td><td>Drive</td></tr> <tr><td>SD17</td><td rowspan="2">Nome file Codice ASCII: 8 caratteri</td></tr> <tr><td>SD18</td></tr> <tr><td>SD19</td><td rowspan="2">Espansione 2E_H(.)</td></tr> <tr><td>SD20</td></tr> <tr><td>SD21</td><td rowspan="2">Codice ASCII: 3 caratteri</td></tr> <tr><td>SD22</td></tr> <tr><td>SD23</td><td rowspan="4">Non usato</td></tr> <tr><td>SD24</td></tr> <tr><td>SD25</td></tr> <tr><td>SD26</td></tr> </tbody> </table> <div style="font-size: x-small;"> <p>Esempio: Nome file = ABCDEFGH.IJK</p> <table border="1" style="font-size: x-small;"> <tr><td colspan="2" style="text-align: center;">b15</td><td colspan="2" style="text-align: center;">b0</td></tr> <tr><td style="text-align: center;">B</td><td style="text-align: center;">A</td><td style="text-align: center;">D</td><td style="text-align: center;">C</td></tr> <tr><td style="text-align: center;">F</td><td style="text-align: center;">E</td><td style="text-align: center;">H</td><td style="text-align: center;">G</td></tr> <tr><td style="text-align: center;">I</td><td style="text-align: center;">.</td><td style="text-align: center;">K</td><td style="text-align: center;">J</td></tr> </table> </div> </div>	Numero	Significato	SD16	Drive	SD17	Nome file Codice ASCII: 8 caratteri	SD18	SD19	Espansione 2E _H (.)	SD20	SD21	Codice ASCII: 3 caratteri	SD22	SD23	Non usato	SD24	SD25	SD26	b15		b0		B	A	D	C	F	E	H	G	I	.	K	J	S (Errore)	Nuovo	●																				
Numero				Significato																																																								
SD16				Drive																																																								
SD17				Nome file Codice ASCII: 8 caratteri																																																								
SD18																																																												
SD19				Espansione 2E _H (.)																																																								
SD20																																																												
SD21				Codice ASCII: 3 caratteri																																																								
SD22																																																												
SD23				Non usato																																																								
SD24																																																												
SD25																																																												
SD26																																																												
b15		b0																																																										
B	A	D	C																																																									
F	E	H	G																																																									
I	.	K	J																																																									
SD17																																																												
SD18																																																												
SD19																																																												
SD20																																																												
SD21																																																												
SD22																																																												
SD23																																																												
SD24																																																												
SD25																																																												
SD26			<p>(2) Tempo (valore effettivo misurato)</p> <table border="1" style="font-size: small;"> <thead> <tr><th>Numero</th><th>Significato</th></tr> </thead> <tbody> <tr><td>SD16</td><td>Tempo: uniti da 1fs (da 0 a 999 fs)</td></tr> <tr><td>SD17</td><td>Tempo: uniti da 1ms (da 0 a 999 ms)</td></tr> <tr><td>SD18</td><td rowspan="10">Non usato</td></tr> <tr><td>SD19</td></tr> <tr><td>SD20</td></tr> <tr><td>SD21</td></tr> <tr><td>SD22</td></tr> <tr><td>SD23</td></tr> <tr><td>SD24</td></tr> <tr><td>SD25</td></tr> <tr><td>SD26</td></tr> </tbody> </table> <p>(3) Posizione errore programma</p> <table border="1" style="font-size: small;"> <thead> <tr><th>Numero</th><th>Significato</th></tr> </thead> <tbody> <tr><td>SD16</td><td rowspan="2">Nome file (Codice ASCII: 8 caratteri)</td></tr> <tr><td>SD17</td></tr> <tr><td>SD18</td><td rowspan="2">Espansione 2E_H(.)</td></tr> <tr><td>SD19</td></tr> <tr><td>SD20</td><td rowspan="2">(Codice ASCII: 3 caratteri)</td></tr> <tr><td>SD21</td></tr> <tr><td>SD22</td><td>Maschera*</td></tr> <tr><td>SD23</td><td>Numero blocco</td></tr> <tr><td>SD24</td><td>Numero passo / transizione</td></tr> <tr><td>SD25</td><td>Num. passo sequenza (L)</td></tr> <tr><td>SD26</td><td>Num. passo sequenza (H)</td></tr> </tbody> </table> <p>* Contenuto della</p> <table style="font-size: x-small;"> <tr> <td style="border: 1px solid black; padding: 2px;">15</td> <td style="border: 1px solid black; padding: 2px;">14</td> <td style="border: 1px solid black; padding: 2px;">--</td> <td style="border: 1px solid black; padding: 2px;">--</td> <td style="border: 1px solid black; padding: 2px;">4</td> <td style="border: 1px solid black; padding: 2px;">3</td> <td style="border: 1px solid black; padding: 2px;">2</td> <td style="border: 1px solid black; padding: 2px;">1</td> <td style="border: 1px solid black; padding: 2px;">0</td> <td style="border: 1px solid black; padding: 2px;">← (Bit No.)</td> </tr> <tr> <td style="border: 1px solid black; padding: 2px;">0</td> <td style="border: 1px solid black; padding: 2px;">0</td> <td style="border: 1px solid black; padding: 2px;">--</td> <td style="border: 1px solid black; padding: 2px;">0</td> <td style="border: 1px solid black; padding: 2px;">0</td> <td style="border: 1px solid black; padding: 2px;">1</td> <td style="border: 1px solid black; padding: 2px;">1</td> <td style="border: 1px solid black; padding: 2px;">1</td> <td style="border: 1px solid black; padding: 2px;">1</td> <td style="border: 1px solid black; padding: 2px;">1</td> </tr> </table> <p style="font-size: x-small;"> Non utilizzato Designazione blocco SFC presente (1) / assente (0) Designazione passo SFC presente (1) / assente (0) Designazione transizione SFC presente (1) / assente (0) </p>	Numero	Significato	SD16	Tempo: uniti da 1fs (da 0 a 999 fs)	SD17	Tempo: uniti da 1ms (da 0 a 999 ms)	SD18	Non usato	SD19	SD20	SD21	SD22	SD23	SD24	SD25	SD26	Numero	Significato	SD16	Nome file (Codice ASCII: 8 caratteri)	SD17	SD18	Espansione 2E _H (.)	SD19	SD20	(Codice ASCII: 3 caratteri)	SD21	SD22	Maschera*	SD23	Numero blocco	SD24	Numero passo / transizione	SD25	Num. passo sequenza (L)	SD26	Num. passo sequenza (H)	15	14	--	--	4	3	2	1	0	← (Bit No.)	0	0	--	0	0	1	1	1	1	1
Numero	Significato																																																											
SD16	Tempo: uniti da 1fs (da 0 a 999 fs)																																																											
SD17	Tempo: uniti da 1ms (da 0 a 999 ms)																																																											
SD18	Non usato																																																											
SD19																																																												
SD20																																																												
SD21																																																												
SD22																																																												
SD23																																																												
SD24																																																												
SD25																																																												
SD26																																																												
Numero		Significato																																																										
SD16	Nome file (Codice ASCII: 8 caratteri)																																																											
SD17																																																												
SD18	Espansione 2E _H (.)																																																											
SD19																																																												
SD20	(Codice ASCII: 3 caratteri)																																																											
SD21																																																												
SD22	Maschera*																																																											
SD23	Numero blocco																																																											
SD24	Numero passo / transizione																																																											
SD25	Num. passo sequenza (L)																																																											
SD26	Num. passo sequenza (H)																																																											
15	14	--	--	4	3	2	1	0	← (Bit No.)																																																			
0	0	--	0	0	1	1	1	1	1																																																			

Tabella dei registri speciali

Tabella dei registri speciali (continua)

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valido per:																					
SD16	Informazioni di errore specifiche	Informazioni di errore specifiche	(4) Numero parametro (5) Numero spia di errore Istruzione CHK numero malfunzionamento	S (Errore)	Nuovo																						
SD17																											
SD18																											
SD19																											
SD20																											
SD21																											
SD22																											
SD23																											
SD24																											
SD25																											
SD26			(6) Intelligent function module parameter error (for System Q CPU only)																								
			<table border="1"> <thead> <tr> <th>Numero</th> <th>Significato</th> <th>Numero</th> <th>Significato</th> </tr> </thead> <tbody> <tr> <td>SD16</td> <td>Numero parametro</td> <td>SD16</td> <td>Num.</td> </tr> <tr> <td>SD17</td> <td rowspan="10">Non usato</td> <td>SD17</td> <td rowspan="10">Non usato</td> </tr> <tr><td>SD18</td></tr> <tr><td>SD19</td></tr> <tr><td>SD20</td></tr> <tr><td>SD21</td></tr> <tr><td>SD22</td></tr> <tr><td>SD23</td></tr> <tr><td>SD24</td></tr> <tr><td>SD25</td></tr> <tr><td>SD26</td></tr> </tbody> </table>	Numero	Significato	Numero	Significato	SD16	Numero parametro	SD16	Num.	SD17	Non usato	SD17	Non usato	SD18	SD19	SD20	SD21	SD22	SD23	SD24	SD25	SD26			
Numero	Significato	Numero	Significato																								
SD16	Numero parametro	SD16	Num.																								
SD17	Non usato	SD17	Non usato																								
SD18																											
SD19																											
SD20																											
SD21																											
SD22																											
SD23																											
SD24																											
SD25																											
SD26																											
			<table border="1"> <thead> <tr> <th>Number</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>SD16</td> <td>Parameter No.</td> </tr> <tr> <td>SD17</td> <td>Error code for intelligent function module</td> </tr> <tr> <td>SD18</td> <td rowspan="9">Vacant</td> </tr> <tr><td>SD19</td></tr> <tr><td>SD20</td></tr> <tr><td>SD21</td></tr> <tr><td>SD22</td></tr> <tr><td>SD23</td></tr> <tr><td>SD24</td></tr> <tr><td>SD25</td></tr> <tr><td>SD26</td></tr> </tbody> </table>	Number	Meaning	SD16	Parameter No.	SD17	Error code for intelligent function module	SD18	Vacant	SD19	SD20	SD21	SD22	SD23	SD24	SD25	SD26								
Number	Meaning																										
SD16	Parameter No.																										
SD17	Error code for intelligent function module																										
SD18	Vacant																										
SD19																											
SD20																											
SD21																											
SD22																											
SD23																											
SD24																											
SD25																											
SD26																											
SD50	Cancellazione errore	Numero errore che esegue cancellazione errore	Memorizza il numero dell'errore che esegue una cancellazione dell'errore	U	Nuovo																						
SD51	Memoria tensione batteria insufficiente	Maschera di bit che indica dove è avvenuta la caduta di tensione	Tutti i bit corrispondenti commutano ON quando la tensione di batteria si abbassa. Questi bit rimangono ON anche dopo che la tensione di batteria è tornata normale.	S (Errore)	Nuovo																						
			<p>Nelle CPU Q00J, Q00, e Q01, viene attivato solo il Bit 0.</p>																								
SD52	Tensione batteria insufficiente	Maschera di bit che indica dove è avvenuta la caduta di tensione	Stessa configurazione come per SD51 Commuta su OFF quando la tensione di batteria viene ripristinata.	S (Errore)	Nuovo																						
SD53	Mancanza rete	Numero di occorrenze AC DOWN (mancanza rete)	Viene sommato 1 al valore memorizzato, ogni volta che la tensione d'ingresso scende sotto al 80% del valore nominale durante il funzionamento della CPU; il valore risultante viene memorizzato in formato BIN.	S (Errore)	D9005	Rem																					

Tabella dei registri speciali (continua)

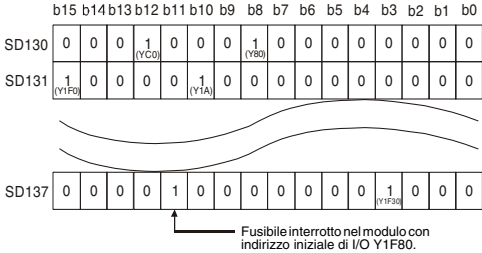
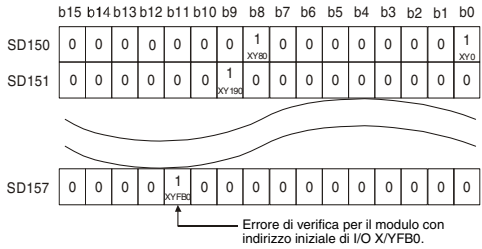
Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valido per:
SD54	Errori comunicazione MINI	Stato rilevamento errori	<p>(1) Il bit corrispondente alla stazione commuta ON quando uno dei moduli MINI (-S3) X(n+0)/X(n+20), X(n+6)/(n+26), X(n+7)/(n+27) o X(n+8)/X(n+28) diventa ON.</p> <p>(2) Commuta ON quando non è possibile la comunicazione fra i MINI (-S3) installati e la CPU.</p> <div style="text-align: center;"> </div>	S (Errore)	D9004 modifica formato	CPU QnA
SD60	Numero fusibile interrotto	Numero del modulo con fusibile interrotto	Il valore memorizzato è il più piccolo numero di stazione corrispondente al modulo con fusibile interrotto, diviso per 16.	S (Errore)	D9000	● Rem
SD61	Errore verifica modulo di I/O	Errore verifica modulo di I/O, numero modulo	Il più basso numero di modulo su cui è stata eseguita la verifica del numero modulo di I/O.	S (Errore)	D9002	
SD62	Numero spia di errore	Numero spia di errore	Viene memorizzato il primo numero di spia di errore rilevato.	S (esecuzione istruzione)	D9009	●
SD63	Numero spie di errore	Numero spie di errore	Memorizza il numero di spie di errore su cui viene eseguita la ricerca.	S (esecuzione istruzione)	D9124	

Tabella dei registri speciali (continua)

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [][][]	Valido per:	
SD90	Valore di impostazione timer di watchdog transizioni di passo (abilitato solo in caso di programma SFC)	Indirizzo F per errore valore impostazione timer e timer intervenuto	Corrisponde a SM90	<p>Numeri F attivati in caso di errore durante l'impostazione del valore del timer di watchdog delle transizioni o dell'intervento del timer.</p> <div style="text-align: center;"> </div> <p>Il timer viene avviato nella transizione ON dei flag da SM90 a SM99 durante il passo attivo, e se le condizioni per le transizioni dei passi non vengono soddisfatte entro il limite impostato, la spia di errore indicata (F) viene attivata.</p>	U	D9108	● (tranne CPU Q00J, Q00 e Q01)
SD91			Corrisponde a SM91			D9109	
SD92			Corrisponde a SM92			D9110	
SD93			Corrisponde a SM93			D9111	
SD94			Corrisponde a SM94			D9112	
SD95			Corrisponde a SM95			D9113	
SD96			Corrisponde a SM96			D9114	
SD97			Corrisponde a SM97			Nuovo	
SD98			Corrisponde a SM98			Nuovo	
SD99			Corrisponde a SM99			Nuovo	
SD100	Velocità di trasmissione	Memorizza la velocità di trasmissione specificata nella configurazione della porta seriale.	K96: 9600 bps, K192: 19.2 kbps, K384: 38.4 kbps, K576: 57.6 kbps, K1152: 115.2 kbps		Nuovo	Q00JCPU, Q00CPU, Q01CPU	
SD101	Configurazione comunicazione	Memorizza la configurazione della comunicazione seriale	Bit 4 = OFF: Senza somma di controllo Bit 4 = ON: Con somma di controllo Bit 5 = OFF: Correzione programma online disabilitata Bit 5 = ON: Correzione programma online abilitata Gli altri bit non sono significativi.	S (accensione o reset)	Nuovo		
SD102	Tempo attesa messaggio	Memorizza il tempo di attesa specificato nella configurazione della porta seriale.	0: Nessuna attesa da 1 a F _H ; Tempo di attesa (unità 10 ms) Default: 0		Nuovo		
SD105	CH1 impostazione velocità di trasmissione (RS232)	Memorizza la velocità di trasmissione attuale.	K3: 300 bps, K6: 600 bps, K24: 2400 bps, K48: 4800 bps, K96: 9600 bps, K192: 19.2 kbps, K384: 38.4 kbps, K576: 57.6 kbps, K1152: 115.2 kbps	S	Nuovo	Q CPU (tranne CPU Q00J, Q00 e Q01)	
SD110	Risultato invio dati	Memorizza il risultato dell'invio dati quando viene usata la comunicazione seriale.	Memorizza il codice di errore incontrato durante la trasmissione sulla linea seriale.	S (Errore)	Nuovo	Q00JCPU, Q00CPU, Q01CPU	
SD111	Risultato ricezione dati	Memorizza il risultato della ricezione dati quando viene usata la comunicazione seriale.	Memorizza il codice di errore incontrato durante la ricezione sulla linea seriale.	S (Errore)	Nuovo		
SD120	Numero di errore per mancanza alimentazione esterna	Numero del modulo che presenta l'errore di mancanza alimentazione.	Memorizza il primo indirizzo iniziale del modulo con alimentazione esterna OFF.	S (Errore)	Nuovo	Q CPU (tranne CPU Q00J, Q00 e Q01)	

Tabella dei registri speciali

(2) Informazioni di sistema

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [][][]	Valido per:						
SD130 SD131 SD132 SD133 SD134 SD135 SD136	Modulo con fusibile interrotto	La maschera di bit (16 bit) indica i moduli con fusibile interrotto. 0: Fusibile non interrotto 1: Fusibile interrotto rilevato	<ul style="list-style-type: none"> Il numero dei moduli di uscita con fusibili interrotti è inserito come una maschera di bit in gruppi da 16. Se il numero dei moduli è inserito da parametri, vengono memorizzati i numeri parametrizzati. Vengono rilevati anche i fusibili interrotti dei moduli di uscita delle stazioni remote. Un bit attivo non viene cancellato automaticamente quando viene sostituito il fusibile interrotto. Il flag è azzerato da una operazione di cancellazione errori. 	S (Errore)	Nuovo	Q00JCPU, Q00CPU, Q01CPU						
SD150 SD151 SD152 SD153 SD154 SD155 SD156							Modulo di I/O errore di verifica	La maschera di bit (16 bit) indica i moduli con errori di verifica. 0: Nessun errore errore di verifica 1: Errore verifica I/O presente	<ul style="list-style-type: none"> All'accensione, i numeri dei moduli di I/O le cui informazioni sono diverse da quelle dei moduli di I/O registrati vengono inseriti in questo registro (in gruppi da 16 punti). Sono anche rilevate le informazioni di I/O. 	S (Errore)	Nuovo	
SD157												

(2) Informazioni di sistema

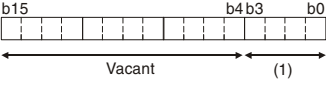
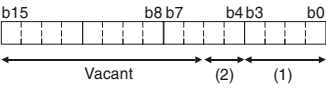
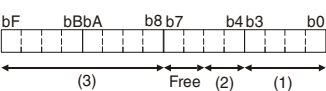

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [][][]	Valido per:
SD200	Stato commutatore	Stato del commutatore della CPU	Lo stato del modulo di I/O remoto viene memorizzato nel formato seguente:  <div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 5px auto;"> (1) Remote I/O module switch status Always 1: STOP </div>	S (Continuo)	Nuovo	I/O Remoto
			Lo stato del commutatore della CPU è memorizzato nel formato seguente:  <div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 5px auto;"> (1) CPU switch status (0): RUN (1): STOP (2) Memory card switch Always OFF </div>		Nuovo	Q00JCPU Q00ĈCPU, Q01CPU
			Lo stato del commutatore della CPU è memorizzato nel formato seguente:  <div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 5px auto;"> (1) CPU switch status (0): RUN (1): STOP (2): L.CLR (2) Memory card switch Always OFF (3) DIP-Switch b8 to bC correspond to SW1 through SW5 of system setting switch 1 . 0: OFF, 1: ON bD,bE and bF are vacant </div>	S (ad ogni elaborazione END)	Nuovo	Q CPU (tranne CPU Q00J, Q00 e Q01)
			Lo stato del commutatore della CPU è memorizzato nel formato seguente:  <div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 5px auto;"> (1) : CPU Status (0) : RUN (1) : STOP (2) : L.CLR (2) : Memory card switch B4 corresponds to card A, B5 corresponds to card B OFF for 0; ON for 1 (3) : DIP switch B8 to B15 correspond to SW1 to SW8 OFF for 0; ON for 1 </div>	S (ad ogni elaborazione END)	Nuovo	CPU QnA

Tabella dei registri speciali

(2) Informazioni di sistema

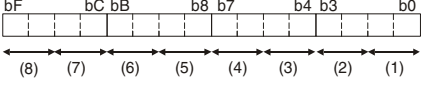
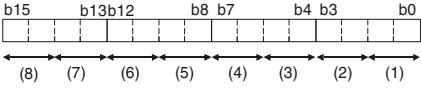
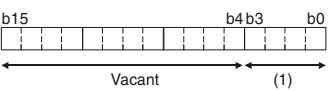
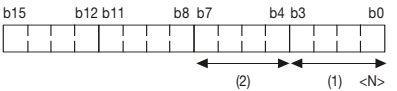
Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [][][]	Valido per:
SD201	Stato LED	Stato LED della CPU	<p>La maschera di bit seguente viene usata per definire lo stato dei LED della CPU:</p>  <p>(1) : RUN (5) : BOOT (2) : ERROR (6) : Non usato (3) : USER (7) : Non usato (4) : BAT.ALARM (8) : MODE Maschere bit per MODE 0: OFF 1: Verde 2: Arancio</p> <p>I bit da 3 a 8 non sono disponibili per Q00JCPU, Q00CPU or Q01CPU.</p>	S (variazione di stato)	Nuovo	CPU System Q
			<p>Le informazioni relative ai seguenti stati dei LED della CPU sono memorizzate nelle maschere seguenti: 0 = OFF; 1 = ON; 2 = lampeggiante</p>  <p>(1) : RUN (5) : BOOT (2) : ERROR (6) : Memory card A (3) : USER (7) : Memory card B (4) : BAT.ALARM (8) : Non usato</p>	S (variazione di stato)	Nuovo	CPU QnA
SD202	LED spenti	Maschera di bit dei LED spenti	<p>Maschere dei LED da spegnere (abilitati solo USER e BOOT) Spento se 1, non spento se 0</p>	U	Nuovo	CPU QnA
SD203	Stato operativo della CPU	Stato operativo della CPU	<p>Lo stato operativo del modulo di I/O remoto viene memorizzato nel formato seguente:</p>  <p>(1) Remote I/O module operating status Always 2: STOP</p>	S (Continuo)	Nuovo	I/O Remoto
			<p>Lo stato operativo della CPU è memorizzato come indicato nella figura seguente:</p>  <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>(1) : Stato operativo CPU0 : RUN 1 : STEP-RUN 2 : STOP 3 : PAUSE</p> <p>(2) : Causa STOP/PAUSE 0 : Commutatore a chiave 1 : Contatto remoto 2 : Periferica, collegamento con computer o operazione da qualche altra sorgente remota 3 : Istruzione di programma</p> </div>	S (ad ogni elaborazione END)	D9015 (modifica formato)	●

Tabella dei registri speciali (continua)

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valido per:															
SD206	Test operandi tipo esecuzione	Indica il tipo di test operandi	<p>Se il test operandi viene eseguito da un dispositivo di programmazione, il contenuto di questo registro riflette lo stato del test:</p> <p>0 = Test non ancora eseguito 1 = Test operandi di ingresso (X) 2 = Test operandi di uscita (Y) 3 = Test operandi di ingresso e uscita (X/Y)</p>	S (Richiesta)	Nuovo	I/O Remoto															
SD207	Priorità visualizzazione LED di errore	Priorità da 1 a 4	<p>Quando viene generato un errore il display a LED visualizza (lampeggiante) il numero di errore in base alle priorità impostate. Le priorità vengono impostate come segue:</p> <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="padding: 2px;">SD207</td> <td style="padding: 2px;">Priorità 4</td> <td style="padding: 2px;">Priorità 3</td> <td style="padding: 2px;">Priorità 2</td> <td style="padding: 2px;">Priorità 1</td> </tr> <tr> <td style="padding: 2px;">SD208</td> <td style="padding: 2px;">Priorità 8</td> <td style="padding: 2px;">Priorità 7</td> <td style="padding: 2px;">Priorità 6</td> <td style="padding: 2px;">Priorità 5</td> </tr> <tr> <td style="padding: 2px;">SD209</td> <td colspan="2" style="padding: 2px;"></td> <td style="padding: 2px;">Priorità 10</td> <td style="padding: 2px;">Priorità 9</td> </tr> </table> <p style="text-align: center;">(4321_H) (8765_H) (00A9_H)</p> <p>Nessuna visualizzazione se è stato impostato "0". Tuttavia in questo caso le informazioni relative agli errori con arresto del funzionamento della CPU (compresa impostazione dei parametri) vengono visualizzate sul display a LED senza condizioni.</p>	SD207	Priorità 4	Priorità 3	Priorità 2	Priorità 1	SD208	Priorità 8	Priorità 7	Priorità 6	Priorità 5	SD209			Priorità 10	Priorità 9	U	D9038	U (tranne CPU Q00J, Q00 e Q01)
SD207		Priorità 4		Priorità 3	Priorità 2	Priorità 1															
SD208		Priorità 8		Priorità 7	Priorità 6	Priorità 5															
SD209			Priorità 10	Priorità 9																	
SD208	Priorità da 5 a 8	D9039 (modifica formato)																			
SD209	Priorità da 9 a 10	Nuovo																			
SD210	Dati orologio	Dati orologio (anno, mese)	<p>L'anno (ultime due cifre) e il mese vengono memorizzati in SD210 in codice BCD come segue:</p> <table style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="padding: 2px;">b15</td><td style="padding: 2px;">b12b11</td><td style="padding: 2px;">b8 b7</td><td style="padding: 2px;">b4 b3</td><td style="padding: 2px;">b0</td> </tr> <tr> <td style="border: 1px solid black; width: 15px; height: 15px;"></td><td style="border: 1px solid black; width: 30px; height: 15px;"></td><td style="border: 1px solid black; width: 30px; height: 15px;"></td><td style="border: 1px solid black; width: 30px; height: 15px;"></td><td style="border: 1px solid black; width: 15px; height: 15px;"></td> </tr> <tr> <td colspan="2" style="text-align: center;">Anno</td><td colspan="3" style="text-align: center;">Mese</td> </tr> </table> <p style="margin-left: 100px;">Esempio: Luglio 1993 = H9307</p>	b15	b12b11	b8 b7	b4 b3	b0						Anno		Mese			D9025		
b15	b12b11	b8 b7	b4 b3	b0																	
Anno		Mese																			
SD211	Dati orologio	Dati orologio (giorno, ora)	<p>Giorno e ora vengono memorizzati in SD211 in codice BCD come segue:</p> <table style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="padding: 2px;">b15</td><td style="padding: 2px;">b12b11</td><td style="padding: 2px;">b8 b7</td><td style="padding: 2px;">b4 b3</td><td style="padding: 2px;">b0</td> </tr> <tr> <td style="border: 1px solid black; width: 15px; height: 15px;"></td><td style="border: 1px solid black; width: 30px; height: 15px;"></td><td style="border: 1px solid black; width: 30px; height: 15px;"></td><td style="border: 1px solid black; width: 30px; height: 15px;"></td><td style="border: 1px solid black; width: 15px; height: 15px;"></td> </tr> <tr> <td colspan="2" style="text-align: center;">Giorno</td><td colspan="3" style="text-align: center;">Ora</td> </tr> </table> <p style="margin-left: 100px;">Esempio: 31, 10 del mattino = H3110</p>	b15	b12b11	b8 b7	b4 b3	b0						Giorno		Ora			S/U (Richiesta)	D9026	● Rem
b15	b12b11	b8 b7	b4 b3	b0																	
Giorno		Ora																			
SD212	Dati orologio	Dati orologio (minuto, secondo)	<p>Minuti e secondi relativi all'ora vengono memorizzati in SD212 in codice BCD come segue:</p> <table style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="padding: 2px;">b15</td><td style="padding: 2px;">b12b11</td><td style="padding: 2px;">b8 b7</td><td style="padding: 2px;">b4 b3</td><td style="padding: 2px;">b0</td> </tr> <tr> <td style="border: 1px solid black; width: 15px; height: 15px;"></td><td style="border: 1px solid black; width: 30px; height: 15px;"></td><td style="border: 1px solid black; width: 30px; height: 15px;"></td><td style="border: 1px solid black; width: 30px; height: 15px;"></td><td style="border: 1px solid black; width: 15px; height: 15px;"></td> </tr> <tr> <td colspan="2" style="text-align: center;">Minuto</td><td colspan="3" style="text-align: center;">Secondo</td> </tr> </table> <p style="margin-left: 100px;">Esempio: 35 min., 48 sec. = H3548</p>	b15	b12b11	b8 b7	b4 b3	b0						Minuto		Secondo			D9027		
b15	b12b11	b8 b7	b4 b3	b0																	
Minuto		Secondo																			

Tabella dei registri speciali

Tabella dei registri speciali (continua)

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valido per:																								
SD213	Dati orologio	Dati orologio (giorno della settimana)	<p>Il giorno della settimana viene memorizzato in SD213 in codice BCD come segue:</p> <p>b15-----b12 b11-----b8 b7-----b4 b3-----b0</p> <p>Cifre pesanti dell'anno (da 0 a 99)</p> <table border="1"> <tr><th>Giorno della</th></tr> <tr><td>0</td><td>Domenic</td></tr> <tr><td>1</td><td>Lunedì</td></tr> <tr><td>2</td><td>Martedì</td></tr> <tr><td>3</td><td>Mercoledì</td></tr> <tr><td>4</td><td>Giovedì</td></tr> <tr><td>5</td><td>Venerdì</td></tr> <tr><td>6</td><td>Sabato</td></tr> </table>	Giorno della	0	Domenic	1	Lunedì	2	Martedì	3	Mercoledì	4	Giovedì	5	Venerdì	6	Sabato	S/U (Richiesta)	D9028	Q CPU Rem									
			Giorno della																											
0	Domenic																													
1	Lunedì																													
2	Martedì																													
3	Mercoledì																													
4	Giovedì																													
5	Venerdì																													
6	Sabato																													
<p>Il giorno della settimana viene memorizzato in SD213 in codice BCD come segue:</p> <p>b15-----b12 b11-----b8 b7-----b4 b3-----b0</p> <p>Sempre 0</p> <table border="1"> <tr><th>Giorno della</th></tr> <tr><td>0</td><td>Domenic</td></tr> <tr><td>1</td><td>Lunedì</td></tr> <tr><td>2</td><td>Martedì</td></tr> <tr><td>3</td><td>Mercoledì</td></tr> <tr><td>4</td><td>Giovedì</td></tr> <tr><td>5</td><td>Venerdì</td></tr> <tr><td>6</td><td>Sabato</td></tr> </table>	Giorno della	0	Domenic	1	Lunedì	2	Martedì	3	Mercoledì	4	Giovedì	5	Venerdì	6	Sabato	S/U (Richiesta)	CPU QnA													
Giorno della																														
0	Domenic																													
1	Lunedì																													
2	Martedì																													
3	Mercoledì																													
4	Giovedì																													
5	Venerdì																													
6	Sabato																													
SD220	Dati per display a LED	Dati per display	<p>I dati ASCII per il display a LED (16 caratteri) sono memorizzati qui:</p> <p>da b15 a b8 da b7 a b0</p> <table border="1"> <tr> <td>SD220</td> <td>Quindicesimo carattere da destra</td> <td>Sedicesimo carattere da destra</td> </tr> <tr> <td>SD221</td> <td>Tredicesimo carattere da destra</td> <td>Quattordicesimo carattere da destra</td> </tr> <tr> <td>SD222</td> <td>Undicesimo carattere da destra</td> <td>Dodicesimo carattere da destra</td> </tr> <tr> <td>SD223</td> <td>Nono carattere da destra</td> <td>Decimo carattere da destra</td> </tr> <tr> <td>SD224</td> <td>Settimo carattere da destra</td> <td>Ottavo carattere da destra</td> </tr> <tr> <td>SD225</td> <td>Quinto carattere da destra</td> <td>Sesto carattere da destra</td> </tr> <tr> <td>SD226</td> <td>Terzo carattere da destra</td> <td>Quarto carattere da destra</td> </tr> <tr> <td>SD227</td> <td>Primo carattere da destra</td> <td>Secondo carattere da destra</td> </tr> </table>	SD220	Quindicesimo carattere da destra	Sedicesimo carattere da destra	SD221	Tredicesimo carattere da destra	Quattordicesimo carattere da destra	SD222	Undicesimo carattere da destra	Dodicesimo carattere da destra	SD223	Nono carattere da destra	Decimo carattere da destra	SD224	Settimo carattere da destra	Ottavo carattere da destra	SD225	Quinto carattere da destra	Sesto carattere da destra	SD226	Terzo carattere da destra	Quarto carattere da destra	SD227	Primo carattere da destra	Secondo carattere da destra	S (variazione di stato)	Nuovo	●
SD220			Quindicesimo carattere da destra	Sedicesimo carattere da destra																										
SD221			Tredicesimo carattere da destra	Quattordicesimo carattere da destra																										
SD222			Undicesimo carattere da destra	Dodicesimo carattere da destra																										
SD223			Nono carattere da destra	Decimo carattere da destra																										
SD224			Settimo carattere da destra	Ottavo carattere da destra																										
SD225			Quinto carattere da destra	Sesto carattere da destra																										
SD226			Terzo carattere da destra	Quarto carattere da destra																										
SD227	Primo carattere da destra	Secondo carattere da destra																												
SD221																														
SD222																														
SD223																														
SD224																														
SD225																														
SD226																														
SD227																														
SD240	Modo operativo del telaio base	0: Modo automatico 1: Modo dettagliato	Memorizza il modo operativo del telaio	S (Inizializzazione)	Nuovo	Q CPU Rem																								
SD241	Numero di telai di espansione	0: Solo telaio di base da 1 a 7; Numero di telai di espansione	Memorizza il numero di telai di espansione installati	S (Inizializzazione)	Nuovo																									

Tabella dei registri speciali (continua)

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valido per:															
SD242	Selezione telai serie A/Q	0: QA[] [] B installato (modo A) 1: Q[] [] B installato (modo Q)	<p style="text-align: center; font-size: small;">When no expansion base is installed, the value for b1 to b4 is fixed to "0".</p>	S (Inizializzazione)	Nuovo	Q00JCPU Q00CPU, Q01CPU															
			<p style="text-align: center; font-size: small;">When no expansion base is installed, the value for b1 to b7 is fixed to "0".</p>			CPU System Q (tranne Q00JCPU Q00CPU, Q01CPU)															
SD243	Numero slot del telaio	Numero slot del telaio Le zone dal quinto al settimo telaio di espansione sono fissate a "0" per Q00JCPU, Q00CPU o Q01CPU.	<table border="1" style="font-size: x-small; border-collapse: collapse;"> <tr> <td style="text-align: center;">bF</td> <td style="text-align: center;">bC bB</td> <td style="text-align: center;">b8 b7</td> <td style="text-align: center;">b4 b3</td> <td style="text-align: center;">b0</td> </tr> <tr> <td style="text-align: center;">SM243</td> <td style="text-align: center;">3rd ext.</td> <td style="text-align: center;">2nd ext.</td> <td style="text-align: center;">1th ext.</td> <td style="text-align: center;">Basic</td> </tr> <tr> <td style="text-align: center;">SM244</td> <td style="text-align: center;">7th ext.</td> <td style="text-align: center;">6th ext.</td> <td style="text-align: center;">5th ext.</td> <td style="text-align: center;">4th ext.</td> </tr> </table>	bF	bC bB	b8 b7	b4 b3	b0	SM243	3rd ext.	2nd ext.	1th ext.	Basic	SM244	7th ext.	6th ext.	5th ext.	4th ext.	S (Inizializzazione)	Nuovo	CPU System Q
bF			bC bB	b8 b7	b4 b3	b0															
SM243	3rd ext.	2nd ext.	1th ext.	Basic																	
SM244	7th ext.	6th ext.	5th ext.	4th ext.																	
SD244	The number of slots being installed is stored in the respective areas for the basic base and the extension bases (ext.).																				
SD250	Caricamento I/O massimo	Caricato massimo numero di I/O	Quando SM250 passa da OFF a ON, viene sommato 1 alle 2 cifre più pesanti dell'indirizzo di I/O dell'ultimo modulo caricato e memorizzato il risultato come valore binario.	S (Richiesta END)	Nuovo	●															
SD251	Indirizzo iniziale I/O da sostituire	Indirizzo iniziale I/O per il modulo da sostituire	Memorizza le due cifre più pesanti del primo indirizzo di un modulo di I/O che viene rimosso/sostituito in stato online.	U	D9094	Q2A(S1) Q3A Q4A Q4AR															
SD253	Velocità trasmissione RS422	Velocità trasmissione RS422	Memorizza la velocità di trasmissione RS422: 0: 9600 bps, 1: 19,2 bps, 2: 38,4 bps	S (se modificato)	Nuovo	CPU QnA															

Tabella dei registri speciali

Tabella dei registri speciali (continua)

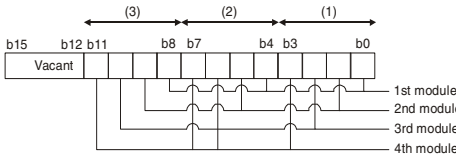
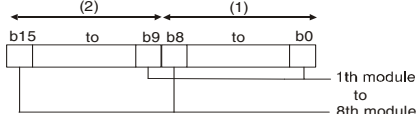
Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valido per:	
SD254	Informazioni MELSECNET/10	Numero di moduli installati	Indica il numero di moduli installati su NET/10	S (Inizializzazione)	Nuovo	●	
SD255		Informazioni dal primo modulo	Indirizzo I/O				Indirizzo I/O NET/10 del primo modulo installato
SD256			Numero rete				Indirizzo di rete NET/10 del primo modulo installato
SD257			Numero gruppo				Numero di gruppo NET/10 del primo modulo installato
SD258			Numero stazione				Numero di stazione NET/10 del primo modulo installato
SD259			Informazioni standby				In caso di stazioni in standby, viene memorizzato il numero di modulo della stazione in standby (da 1 a 4).
SD260 – SD264		Informazioni dal secondo modulo	Configurazione identica a quella del primo modulo.				
SD265 – SD269		Informazioni dal terzo modulo	Configurazione identica a quella del primo modulo.				
SD270 – SD274		Informazioni dal quarto modulo	Configurazione identica a quella del primo modulo.				
SD280	Errore CC-Link	Error detection status	 <p>(1) Se Xn0 del CC-Link installato commuta ON, il bit corrispondente alla stazione commuta ON.</p> <p>(2) Se Xn1 o XnF del CC-Link installato commuta su OFF, il bit corrispondente alla stazione commuta su ON.</p> <p>(3) Commuta ON se la CPU non riesce a comunicare con il CC-Link installato.</p>	S (errore)	Nuovo	Q CPU	
			 <p>(1) Se Xn0 del CC-Link installato commuta ON, il bit corrispondente alla stazione commuta ON.</p> <p>(2) Se Xn1 o XnF del CC-Link installato commuta su OFF, il bit corrispondente alla stazione commuta su ON.</p>				S (errore)

Tabella dei registri speciali (continua)

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valido per:
SD290	Allocazione operandi (identica alla parametrizzazione)	Numero di locazioni allocate per X	● Contiene il numero di locazioni attualmente impostate per X	S (Inizializzazione)	Nuovo	● Rem
SD291		Numero di locazioni allocate per Y	● Contiene il numero di locazioni attualmente impostate per Y			
SD292		Numero di locazioni allocate per M	● Contiene il numero di locazioni attualmente impostate per M			
SD293		Numero di locazioni allocate per L	● Contiene il numero di locazioni attualmente impostate per L			
SD294		Numero di locazioni allocate per B	● Contiene il numero di locazioni attualmente impostate per B			
SD295		Numero di locazioni allocate per F	● Contiene il numero di locazioni attualmente impostate per F			
SD296		Numero di locazioni allocate per SB	● Contiene il numero di locazioni attualmente impostate per SB			
SD297		Numero di locazioni allocate per V	● Contiene il numero di locazioni attualmente impostate per V			
SD298		Numero di locazioni allocate per S	● Contiene il numero di locazioni attualmente impostate per S			
SD299		Numero di locazioni allocate per T	● Contiene il numero di locazioni attualmente impostate per T			
SD300		Numero di locazioni allocate per ST	● Contiene il numero di locazioni attualmente impostate per ST			
SD301		Numero di locazioni allocate per C	● Contiene il numero di locazioni attualmente impostate per C			
SD302		Numero di locazioni allocate per D	● Contiene il numero di locazioni attualmente impostate per D			
SD303		Allocazione operandi (identica alla parametrizzazione)	Numero di locazioni allocate per W			
SD304	Numero di locazioni allocate per SW		● Contiene il numero di locazioni attualmente impostate per SW			
SD315	Tempo riservato per processi di comunicazione	Tempo riservato per processi di comunicazione	<p>Riserva il tempo specificato per i processi di comunicazione con GX Developer o altre unit�.</p> <p>A un tempo specificato pi� grande corrisponde un tempo di risposta per la comunicazione con altri dispositivi (GX Developer, unit� di comunicazione seriale) pi� breve.</p> <p>Campo impostazione: da 1 a 100 ms.</p> <p>Se il valore specificato � fuori campo, viene considerato come non impostato.</p> <p>Il tempo di scansione si allunga del tempo specificato.</p>	Elaborazione END	Nuovo	CPU System Q

Tabella dei registri speciali

Tabella dei registri speciali (continua)

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] [] []	Valido per:	
SD340	Informazioni Ethernet	Numero di moduli installato	● Indica il numero di moduli installati sulla rete Ethernet.	S (Inizializzazione)	Nuovo	CPU System Q Rem	
SD341		Informazioni dal primo modulo	Indirizzo di I/O				● Indirizzo di I/O Ethernet del primo modulo installato.
SD342			Numero di rete				● Numero di rete Ethernet del primo modulo installato.
SD343			Numero gruppo				● Numero gruppo Ethernet del primo modulo installato.
SD344			Numero stazione				● Numero stazione Ethernet del primo modulo installato.
SD345 e SD346			Non usato				● Non usato (con una CPU System Q, l'indirizzo IP Ethernet del primo modulo è contenuto nel buffer di memoria.
SD347			Non usato			● Non usato (Con CPU System Q, il codice di errore Ethernet del primo modulo viene letto con l'istruzione ERRORRD.	
da SD348 a SD354			Informazioni dal secondo modulo			● Configurazione identica a quella del primo modulo.	
da SD355 a SD361		Informazioni dal terzo modulo	● Configurazione identica a quella del primo modulo.				
da SD362 a SD368		Informazioni dal quarto modulo	● Configurazione identica a quella del primo modulo.				
SD340		Informazioni Ethernet	Numero di moduli installato			● Indica il numero di moduli installati sulla rete Ethernet.	S (Inizializzazione)
SD341	Informazioni dal primo modulo		Indirizzo di I/O	● Indirizzo di I/O Ethernet del primo modulo installato.			
SD342			Numero di rete	● Numero di rete Ethernet del primo modulo installato.			
SD343			Numero gruppo	● Numero gruppo Ethernet del primo modulo installato.			
SD344			Numero stazione	● Numero stazione Ethernet del primo modulo installato.			
SD345 e SD346			Indirizzo IP	● Indirizzo IP Ethernet del primo modulo installato.			
SD347			Codice di errore	● Codice di errore Ethernet del primo modulo installato.			
da SD348 a SD354			Informazioni dal secondo modulo	● Configurazione identica a quella del primo modulo.			
da SD355 a SD361	Informazioni dal terzo modulo		● Configurazione identica a quella del primo modulo.				
da SD362 a SD368	Informazioni dal quarto modulo		● Configurazione identica a quella del primo modulo.				

Tabella dei registri speciali (continua)

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [][][]	Valido per:
SD380	Stato ricezione Ethernet	Stato ricezione istruzione del primo modulo	<p>ON: Received (Channel is used) OFF: Not received (Channel is not used)</p>	S (Inizializzazione)	Nuovo	CPU QnA
SD381		Stato ricezione istruzione del secondo modulo	● Configurazione identica a quella del primo modulo.			
SD382		Stato ricezione istruzione del terzo modulo	● Configurazione identica a quella del primo modulo.			
SD383		Stato ricezione istruzione del quarto modulo	● Configurazione identica a quella del primo modulo.			
SD392	Versione software	Versione software sistema interno	<ul style="list-style-type: none"> ● La versione software è memorizzata nel byte pesante di SD392 in codice ASCII. Ad esempio, la versione "A" viene memorizzata come 41_H. Il dato nel byte leggero non è significativo. Nota: La versione interna del software di sistema può essere diversa da quella indicata dal simbolo di versione stampato sull'involucro. 	S (Inizializzazione)	D9060	
SD395	Numero CPU	1: CPU Numero 1 2: CPU Numero 2 3: CPU Numero 3 4: CPU Numero 4	Memorizza il numero della CPU quando questa funziona in un sistema multi CPU.	S (Inizializzazione)	Nuovo	Q02CPU Q02HCPU Q06HCPU Q12HCPU Q25HCPU con versione funzionale B o successiva

Tabella dei registri speciali

(3) Orologio di sistema/contatori

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] [] []	Valido per:
SD412	Contatore a 1 secondo	Numero di conti in unitf da 1 secondo	Dal momento in cui la CPU entra nello stato RUN, il valore 1 viene sommato ad ogni secondo. Il conteggio si ripete da 0 a 32767 e da -32768 a 0	S (variazione di stato)	D9022	●
SD414	n = passi da 1 secondo	Impulsi da 2n secondi	Memorizza il valore n di un impulso a cadenza 2n secondi (default 30). Il campo di impostazione ċ da 1 a 32767.	U	Nuovo	
SD415	n = passi da 1 ms	Impulsi da 2n ms	Memorizza il valore n di un impulso a cadenza 2n ms (default 30). Il campo di impostazione ċ da 1 a 32767.	U	Nuovo	CPU System Q tranne Q00JCPU Q00ĈCPU, Q01CPU)
SD420	Contatore scansioni	Conta il numero di scansioni	Incrementato di 1 all'esecuzione di ogni scansione, dopo che la CPU entra nello stato RUN. Il conteggio si ripete da 0 a 32767 e da -32768 a 0	S (ad ogni elaborazione END)	Nuovo	●
SD430	Contatore cicli lenti	Conta il numero di scansioni	Incrementato di 1 all'esecuzione di ogni scansione, dopo che la CPU entra nello stato RUN. Il conteggio si ripete da 0 a 32767 e da -32768 a 0 Usato solo per programmi a ciclo lento	S (ad ogni elaborazione END)	Nuovo	● (tranne Q00JCPU Q00ĈCPU, Q01CPU)

(4) Informazioni scansione

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valido per:
SD500	Esecuzione programma numero	Tipo di esecuzione del programma eseguito	Numero di programma del programma attualmente in esecuzione, memorizzato come valore binario.	S (variazione di stato)	Nuovo	● (tranne Q00JCPU)
SD510	Numero programma a ciclo lento	Nome file del programma a ciclo lento	Numero di programma del programma a ciclo lento in esecuzione, memorizzato come valore binario. Abilitato solo se SM510 è ON.	S (ad ogni elaborazione END)	Nuovo	● Q00CPU, Q01CPU)
SD520	Tempo scansione attuale	Tempo scansione attuale (unitr da 1 ms)	Memorizza il tempo di scansione attuale (in unitr da 1 ms) Campo da 0 a 65535	S (ad ogni elaborazione END)	D9017 (modifica formato)	●
SD521		Tempo scansione attuale (in unitr da 1 μs)	Memorizza il tempo di scansione attuale (in unitr da 1 μs) Campo da 00000 a 900 (Esempio) Una scansione di durata 23,6 ms viene memorizzata come segue: D520 = 23 D521 = 600		Nuovo	
SD522	Tempo prima scansione	Tempo prima scansione (unitr da 1 ms)	Memorizza il tempo della prima scansione (in unitr da 1 ms) Campo da 0 a 65535	S (alla prima elaborazione END)	Nuovo	● (tranne Q00JCPU Q00CPU, Q01CPU)
SD523		Tempo prima scansione (in unitr da 100 μs)	Memorizza il tempo della prima scansione (in unitr da 1 μs) Campo da 000 a 900			
SD524	Minimo tempo di scansione	Minimo tempo di scansione (unitr da 1 ms)	Memorizza il più basso tempo di scansione (in unitr da 1 ms) Campo da 0 a 65535	S (ad ogni elaborazione END)	D9018 (modifica formato)	●
SD525		Minimo tempo di scansione (in unitr da 100 μs)	Memorizza il più basso tempo di scansione (in unitr da 100 μs) Campo da 000 a 900		Nuovo	
SD526	Massimo tempo di scansione	Massimo tempo di scansione (unitr da 1 ms)	Memorizza il più alto tempo di scansione, tranne la prima scansione (in unitr da 1 ms) Campo da 0 a 65535	S (ad ogni elaborazione END)	D9019 (modifica formato)	●
SD527		Massimo tempo di scansione (in unitr da 100 μs)	Memorizza il più alto tempo di scansione, tranne la prima scansione (in unitr da 100 μs) Campo da 000 a 900		Nuovo	
SD528	Tempo di scansione attuale per programmi a ciclo lento	Tempo scansione attuale (unitr da 1 ms)	Memorizza il tempo di scansione attuale per i programmi a ciclo lento (in unitr da 1 ms).	S (ad ogni elaborazione END)	Nuovo	●
SD529		Tempo scansione attuale (in unitr da 100 μs)	Memorizza il tempo di scansione attuale per i programmi a ciclo lento (in unitr da 100 μs). Campo da 000 a 900			
SD532	Minimo tempo di scansione per i programmi a ciclo lento	Minimo tempo di scansione (unitr da 1 ms)	Memorizza il tempo di scansione più basso per i programmi a ciclo lento (in unitr da 1 ms). Campo da 0 a 65535	S (ad ogni elaborazione END)	Nuovo	● (tranne Q00JCPU Q00CPU, Q01CPU)
SD533		Minimo tempo di scansione (in unitr da 100 μs)	Memorizza il tempo di scansione più basso per i programmi a ciclo lento (in unitr da 100 μs). Campo da 000 a 900			
SD534	Massimo tempo di scansione per i programmi a ciclo lento	Massimo tempo di scansione (in unitr da 1 ms)	Memorizza il più alto tempo di scansione per i programmi a ciclo lento, non considerando la prima scansione (in unitr da 1 ms). Campo da 0 a 65535	S (ad ogni elaborazione END)	Nuovo	●
SD535		Massimo tempo di scansione (in unitr da 100 μs)	Memorizza il più alto tempo di scansione per i programmi a ciclo lento, non considerando la prima scansione (in unitr da 100 μs). Campo da 000 a 900			

Tabella dei registri speciali

Tabella dei registri speciali (continua)

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valido per:
SD540	Tempo elaborazione END	Tempo di elaborazione END (in unitr da 1 ms)	Memorizza il tempo dal completamento di una scansione di programma all'inizio della successiva (unitr da 1 ms). Campo da 0 a 65535	S (ad ogni elaborazione END)	Nuovo	●
SD541		Tempo elaborazione END (in unitr da 100 μs)	Memorizza il tempo dal completamento di una scansione di programma all'inizio della successiva (unitr da 100 μs). Campo da 000 a 900			
SD542	Tempo di attesa per scansione costante	Tempo di attesa per scansione costante (in unitr da 1 ms)	Memorizza il tempo di attesa se è stata impostata la scansione a cadenza costante (unitr da 1 ms). Campo da 0 a 65535	S (alla prima elaborazione END)	Nuovo	●
SD543		Tempo di attesa per scansione costante (in unitr da 100 μs)	Memorizza il tempo di attesa se è stata impostata la scansione a cadenza costante (unitr da 100 μs). Campo da 000 a 900			
SD544	Tempo di esecuzione cumulativo per i programmi a ciclo lento	Tempo di esecuzione cumulativo per i programmi a ciclo lento (unitr da 1 ms)	Memorizza il tempo di esecuzione cumulativo per i programmi a ciclo lento (unitr da 1 ms). Campo da 0 a 65535 Azzerato dopo una scansione a ciclo lento	S (ad ogni elaborazione END)	Nuovo	● (tranne Q00JCPU, Q00CPU, Q01CPU)
SD545		Tempo di esecuzione cumulativo per i programmi a ciclo lento (in unitr da 100 μs)	Memorizza il tempo di esecuzione cumulativo per i programmi a ciclo lento (in unitr da 100 μs) Campo da 000 a 900 Azzerato dopo una scansione a ciclo lento			
SD546	Tempo di esecuzione per programmi a ciclo lento	Tempo di esecuzione per programmi a ciclo lento (unitr da 1 ms)	Memorizza il tempo di esecuzione di un programma a ciclo lento durante 1 scansione (unitr da 1 ms). Campo da 0 a 65535 Memorizza tutte le scansioni	S (ad ogni elaborazione END)	Nuovo	●
SD547		Tempo di esecuzione per programmi a ciclo lento (in unitr da 100 μs)	Memorizza il tempo di esecuzione di un programma a ciclo lento durante 1 scansione (unitr da 100 μs). Campo da 000 a 900 Memorizza tutte le scansioni			
SD548	Tempo di esecuzione programma "scan execution"	Tempo di esecuzione programma "scan execution" (unitr da 1 ms)	Memorizza il tempo di esecuzione per un programma "scan execution" in una scansione (unitr da 1 ms). Campo da 0 a 65535 Memorizza tutte le scansioni	S (ad ogni elaborazione END)	Nuovo	●
SD549		Tempo di esecuzione programma "scan execution" (in unitr da 100 μs)	Memorizza il tempo di esecuzione per un programma "scan execution" in una scansione (unitr da 100 μs). Campo da 000 a 900 Memorizza tutte le scansioni			
SD550	Misura intervalli di manutenzione modulo	Num. unitr/modulo	Imposta l'indirizzo di I/O del modulo per cui si misura l'intervallo di manutenzione.	U	Nuovo	●
SD551	Intervallo di servizio	Intervallo di servizio modulo (unitr da 1 ms)	Se SM551 è ON, memorizza l'intervallo di servizio per il modulo indicato da SD550 (unitr da 1 ms). Campo da 0 a 65535	S (Richiesta)	Nuovo	● (tranne Q00JCPU, Q00CPU, Q01CPU)
SD552		Intervallo di servizio modulo (in unitr da 100 μs)	Se SM551 è ON, memorizza l'intervallo di servizio per il modulo indicato da SD550 (unitr da 1 μs). Campo da 0 a 999			

(5) Memory card

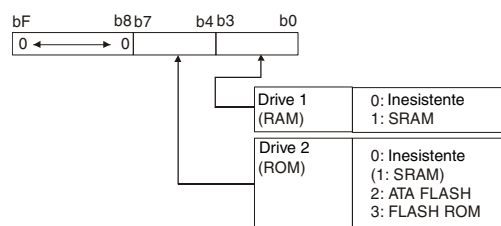
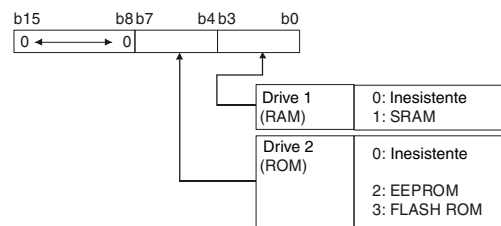
Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valido per:														
SD600	Modelli memory card A	Memory card A Modelli	Indica il modello di memory card A installato. 	S (inizializzazione e rimozione scheda)	Nuovo	CPU System Q (tranne Q00JCPU, Q00CPU, Q01CPU)														
			Indica il modello di memory card A installato. 	S (inizializzazione e rimozione scheda)	Nuovo	CPU QnA														
SD602	Capacità drive 1 (RAM)	Capacità drive 1	La capacità del drive 1 è espressa in unità da 1 kByte	S (inizializzazione e rimozione scheda)	Nuovo	● (tranne Q00JCPU, Q00CPU, Q01CPU)														
SD603	Capacità drive 2 (ROM)	Capacità drive 2	La capacità del drive 2 è espressa in unità da 1 kByte	S (inizializzazione e rimozione scheda)	Nuovo	● (tranne Q00JCPU, Q00CPU, Q01CPU)														
SD604	Condizioni di utilizzo memory card A	Condizioni di utilizzo memory card A	Le condizioni di utilizzo della memory card A sono memorizzate come maschere di bit (condizione usata = ON). Il significato di queste maschere è mostrato di seguito: <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">b0 : BOOT operation (QBT)</td> <td style="padding: 2px;">b8 : CPU fault history (QFD)</td> </tr> <tr> <td style="padding: 2px;">b1 : Parameters (QPT)</td> <td style="padding: 2px;">b9 : SFC trace (QTS)</td> </tr> <tr> <td style="padding: 2px;">b2 : Device comments (QCD)</td> <td style="padding: 2px;">bA : Local device (QDL)</td> </tr> <tr> <td style="padding: 2px;">b3 : Device initial value (QDI)</td> <td style="padding: 2px;">bC :</td> </tr> <tr> <td style="padding: 2px;">b4 : File Register (QDR)</td> <td style="padding: 2px;">bD :</td> </tr> <tr> <td style="padding: 2px;">b5 : Trace (QTS)</td> <td style="padding: 2px;">bE :</td> </tr> <tr> <td style="padding: 2px;">b6 :</td> <td style="padding: 2px;">bF :</td> </tr> </table>	b0 : BOOT operation (QBT)	b8 : CPU fault history (QFD)	b1 : Parameters (QPT)	b9 : SFC trace (QTS)	b2 : Device comments (QCD)	bA : Local device (QDL)	b3 : Device initial value (QDI)	bC :	b4 : File Register (QDR)	bD :	b5 : Trace (QTS)	bE :	b6 :	bF :	S (variazione di stato)	Nuovo	CPU System Q (tranne Q00JCPU, Q00CPU, Q01CPU)
			b0 : BOOT operation (QBT)	b8 : CPU fault history (QFD)																
b1 : Parameters (QPT)	b9 : SFC trace (QTS)																			
b2 : Device comments (QCD)	bA : Local device (QDL)																			
b3 : Device initial value (QDI)	bC :																			
b4 : File Register (QDR)	bD :																			
b5 : Trace (QTS)	bE :																			
b6 :	bF :																			
Le condizioni di utilizzo della memory card A sono memorizzate come maschere di bit (condizione usata = ON). Il significato di queste maschere è mostrato di seguito: <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">b0 : BOOT operation (QBT)</td> <td style="padding: 2px;">b8 : Simulation data (QDS)</td> </tr> <tr> <td style="padding: 2px;">b1 : Parameters (QPT)</td> <td style="padding: 2px;">b9 : CPU fault history (QFD)</td> </tr> <tr> <td style="padding: 2px;">b2 : Device comments (QCD)</td> <td style="padding: 2px;">bA : SFC trace (QTS)</td> </tr> <tr> <td style="padding: 2px;">b3 : Device initial value (QDI)</td> <td style="padding: 2px;">bB : Local device (QDL)</td> </tr> <tr> <td style="padding: 2px;">b4 : File Register (QDR)</td> <td style="padding: 2px;">bC :</td> </tr> <tr> <td style="padding: 2px;">b5 : Sampling trace (QTS)</td> <td style="padding: 2px;">bD :</td> </tr> <tr> <td style="padding: 2px;">b6 : Status latch (QTL)</td> <td style="padding: 2px;">bE :</td> </tr> <tr> <td style="padding: 2px;">b7 : Program trace (QTP)</td> <td style="padding: 2px;">bF :</td> </tr> </table>	b0 : BOOT operation (QBT)	b8 : Simulation data (QDS)	b1 : Parameters (QPT)	b9 : CPU fault history (QFD)	b2 : Device comments (QCD)	bA : SFC trace (QTS)	b3 : Device initial value (QDI)	bB : Local device (QDL)	b4 : File Register (QDR)	bC :	b5 : Sampling trace (QTS)	bD :	b6 : Status latch (QTL)	bE :	b7 : Program trace (QTP)	bF :	S (variazione di stato)	Nuovo	CPU QnA	
b0 : BOOT operation (QBT)	b8 : Simulation data (QDS)																			
b1 : Parameters (QPT)	b9 : CPU fault history (QFD)																			
b2 : Device comments (QCD)	bA : SFC trace (QTS)																			
b3 : Device initial value (QDI)	bB : Local device (QDL)																			
b4 : File Register (QDR)	bC :																			
b5 : Sampling trace (QTS)	bD :																			
b6 : Status latch (QTL)	bE :																			
b7 : Program trace (QTP)	bF :																			

Tabella dei registri speciali

(5) Memory card

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valido per:																
SD620	Modelli memory card B	Memory card B Modelli	<p>Indica il modello di memory card B installato.</p> <p>Il valore del drive 4 è fisso a "3" perché incorpora una FLASH ROM.</p> <table border="1"> <tr> <td>Drive 3 (RAM)</td> <td>0: Inesistente 1: SRAM</td> </tr> <tr> <td>Drive 4 (ROM)</td> <td>0: Inesistente 1: SRAM 2: ATA FLASH 3: FLASH ROM</td> </tr> </table>	Drive 3 (RAM)	0: Inesistente 1: SRAM	Drive 4 (ROM)	0: Inesistente 1: SRAM 2: ATA FLASH 3: FLASH ROM	S (Inizializzazione)	Nuovo	CPU System Q												
			Drive 3 (RAM)	0: Inesistente 1: SRAM																		
Drive 4 (ROM)	0: Inesistente 1: SRAM 2: ATA FLASH 3: FLASH ROM																					
<p>Indica il modello di memory card B installato.</p> <table border="1"> <tr> <td>Drive 1 (RAM)</td> <td>0: Inesistente 1: SRAM</td> </tr> <tr> <td>Drive 2 (ROM)</td> <td>0: Inesistente 2: EEPROM 3: FLASH ROM</td> </tr> </table>	Drive 1 (RAM)	0: Inesistente 1: SRAM	Drive 2 (ROM)	0: Inesistente 2: EEPROM 3: FLASH ROM	S (Inizializzazione)	Nuovo	CPU QnA															
Drive 1 (RAM)	0: Inesistente 1: SRAM																					
Drive 2 (ROM)	0: Inesistente 2: EEPROM 3: FLASH ROM																					
SD622	Capacità drive 3 (RAM)	Capacità drive 3	<p>La capacità del drive 3 è espressa in unità da 1 kByte. In una CPU serie Q questo valore è fisso a "61" perché la RAM interna è da 61 kByte.</p>	S (Inizializzazione)	Nuovo	CPU System Q																
			<p>La capacità del drive 3 è espressa in unità da 1 kByte</p>	S (Inizializzazione)	Nuovo	Q2(S1) Q3A Q4A Q4AR CPU																
SD623	Capacità drive 4 (ROM)	Capacità drive 4	La capacità del drive 4 è espressa in unità da 1 kByte	S (Inizializzazione)	Nuovo	Q2(S1) Q3A Q4A Q4AR System Q CPU																
SD624	Condizioni di uso drive 3	Condizioni di uso drive 3	<p>La condizione di uso del drive 3 è indicata dal bit 4: b4 = OFF: Drive 3 non usato b4 = ON: Drive 3 usato per memorizzare i file registri</p>	S (variazione di stato)	Nuovo	Q00JCPU, Q00CPU, Q01CPU																
	Condizioni di uso drive 3 e drive 4	Condizioni di uso drive 3 e drive 4	<p>Le condizioni di utilizzo della memory card B sono memorizzate come maschere di bit. (usata se ON) Il significato di queste maschere è mostrato di seguito:</p> <table border="1"> <tr> <td>b0 : BOOT operation (QBT)</td> <td>b8 : CPU fault history (QFD)</td> </tr> <tr> <td>b1 : Parameters (QPA)</td> <td>b9 : SFC trace (QTS)</td> </tr> <tr> <td>b2 : Device comments (QCD)</td> <td>bA : Local device (QDL)</td> </tr> <tr> <td>b3 : Device initial value (QDI)</td> <td>bB : Local device (QDL)</td> </tr> <tr> <td>b4 : File R (QDR)</td> <td>bC :</td> </tr> <tr> <td>b5 : Trace (QTS)</td> <td>bD :</td> </tr> <tr> <td>b6 :</td> <td>bE :</td> </tr> <tr> <td>b7 :</td> <td>bF :</td> </tr> </table>	b0 : BOOT operation (QBT)	b8 : CPU fault history (QFD)	b1 : Parameters (QPA)	b9 : SFC trace (QTS)	b2 : Device comments (QCD)	bA : Local device (QDL)	b3 : Device initial value (QDI)	bB : Local device (QDL)	b4 : File R (QDR)	bC :	b5 : Trace (QTS)	bD :	b6 :	bE :	b7 :	bF :	S (variazione di stato)	Nuovo	CPU System Q (tranne Q00JCPU Q00CPU Q01CPU)
	b0 : BOOT operation (QBT)	b8 : CPU fault history (QFD)																				
b1 : Parameters (QPA)	b9 : SFC trace (QTS)																					
b2 : Device comments (QCD)	bA : Local device (QDL)																					
b3 : Device initial value (QDI)	bB : Local device (QDL)																					
b4 : File R (QDR)	bC :																					
b5 : Trace (QTS)	bD :																					
b6 :	bE :																					
b7 :	bF :																					
Memory card B Condizioni di uso	Memory card B Condizioni di uso	<p>Le condizioni di utilizzo della memory card B sono memorizzate come maschere di bit. (usata se ON) Il significato di queste maschere è mostrato di seguito:</p> <table border="1"> <tr> <td>b0 : BOOT operation (QBT)</td> <td>b8 : Simulation data (QDS)</td> </tr> <tr> <td>b1 : Parameters (QPT)</td> <td>b9 : CPU fault history (QFD)</td> </tr> <tr> <td>b2 : Device comments (QCD)</td> <td>bA : SFC trace (QTS)</td> </tr> <tr> <td>b3 : Device initial value (QDI)</td> <td>bB : Local device (QDL)</td> </tr> <tr> <td>b4 : File Register (QDR)</td> <td>bC :</td> </tr> <tr> <td>b5 : Sampling trace (QTS)</td> <td>bD :</td> </tr> <tr> <td>b6 : Status latch (QTL)</td> <td>bE :</td> </tr> <tr> <td>b7 : Program trace (QTP)</td> <td>bF :</td> </tr> </table>	b0 : BOOT operation (QBT)	b8 : Simulation data (QDS)	b1 : Parameters (QPT)	b9 : CPU fault history (QFD)	b2 : Device comments (QCD)	bA : SFC trace (QTS)	b3 : Device initial value (QDI)	bB : Local device (QDL)	b4 : File Register (QDR)	bC :	b5 : Sampling trace (QTS)	bD :	b6 : Status latch (QTL)	bE :	b7 : Program trace (QTP)	bF :	S (variazione di stato)	Nuovo	Q2(S1) Q3A Q4A Q4AR CPU	
b0 : BOOT operation (QBT)	b8 : Simulation data (QDS)																					
b1 : Parameters (QPT)	b9 : CPU fault history (QFD)																					
b2 : Device comments (QCD)	bA : SFC trace (QTS)																					
b3 : Device initial value (QDI)	bB : Local device (QDL)																					
b4 : File Register (QDR)	bC :																					
b5 : Sampling trace (QTS)	bD :																					
b6 : Status latch (QTL)	bE :																					
b7 : Program trace (QTP)	bF :																					

Tabella dei registri speciali (continua)

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valido per:
SD640	Drive per file registri	Numero drive	Memorizza il numero di drive usato dal file registri	S (variazione di stato)	Nuovo	
SD641	File registri Nome file	File registri Nome file	Memorizza il nome del file registri (con estensione) definito in configurazione o tramite l'uso di una istruzione QDRSET, in codice ASCII.	S (variazione di stato)	Nuovo	●
SD642						
SD643			b15 b8 b7 b0			
SD644			SD641 Secondo carattere Primo carattere			
SD645			SD642 Quarto carattere Terzo carattere			
SD646			SD643 Sesto carattere Quinto carattere			
			SD644 Ottavo carattere Settimo carattere			
	SD645 Primo carattere estensione 2EH (.)					
	SD646 Terzo carattere estensione Secondo carattere estensione					
SD647	Capacità file registri	Capacità file registri	Memorizza la capacità di memorizzazione del file registri attualmente selezionato in unità da 1 K word.	S (variazione di stato)	Nuovo	
SD648	Numero di blocco file registri	Numero di blocco file registri	Memorizza il numero di blocco attualmente selezionato del file registri.	S (variazione di stato)	D9035	
SD650	Drive commenti	Drive commenti	Memorizza il numero del drive per i commenti selezionato con la parametrizzazione o con l'istruzione QCDSET.	S (variazione di stato)	Nuovo	
SD651	Nome file commenti	Nome file commenti	Memorizza il nome del file commenti (con estensione) selezionato da parametrizzazione o tramite l'istruzione QCDSET, in codice ASCII.	S (variazione di stato)	Nuovo	● (tranne Q00JCPU, Q00CPU, Q01CPU)
SD652						
SD653			b15 b8 b7 b0			
SD654			SD651 Secondo carattere Primo carattere			
SD655			SD652 Quarto carattere Terzo carattere			
SD656			SD653 Sesto carattere Quinto carattere			
			SD654 Ottavo carattere Settimo carattere			
	SD655 Primo carattere estensione 2EH (.)					
	SD656 Terzo carattere estensione Secondo carattere estensione					
SD660	File per processo di boot	Numero drive per file di boot	Memorizza il numero di drive contenente il file di boot (*.QBT).	S (Inizializzazione)	Nuovo	
SD661		Nome del file di boot	Memorizza il nome del file di boot (*.QBT).	S (Inizializzazione)	Nuovo	
SD662			b15 b8 b7 b0			
SD663			SD661 Secondo carattere Primo carattere			
SD664			SD662 Quarto carattere Terzo carattere			
SD665			SD663 Sesto carattere Quinto carattere			
SD666			SD664 Ottavo carattere Settimo carattere			
			SD665 Primo carattere estensione 2EH (.)			
	SD666 Terzo carattere estensione Secondo carattere estensione					

Tabella dei registri speciali

(6) Registri relativi alle istruzioni

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valido per:												
SD705	Maschera	Maschera	Durante le operazioni sui blocchi, l'attivazione di SM705 consente di usare la maschera contenuta in SD705 (o in SD705 e SD706 se vengono usate doppie parole) per operare su tutti i dati del blocco con valori mascherati.	U	Nuovo	● (tranne Q00JCPU, Q00CPU, Q01CPU)												
SD706																		
SD714	Numero di comunicazioni non usate nelle aree di registrazione richiesta	Primo carattere da 0 a 32	Memorizza il numero di blocchi non usati nell'area di richiesta comunicazione per i moduli terminali remoti collegati a AJ71PT32-S3.	S (durante l'esecuzione)	M9081	CPU QnA												
SD715	IMASK istruzione maschera	Maschera	I valori mascherati dall'uso dell'istruzione IMASK sono memorizzati nel modo seguente: <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">b15</td> <td style="text-align: center;">.....</td> <td style="text-align: center;">b0</td> </tr> <tr> <td>SD715</td> <td> 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0</td> <td></td> </tr> <tr> <td>SD716</td> <td> 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16</td> <td></td> </tr> <tr> <td>SD717</td> <td> 47 46 45 44 43 42 41 40 39 38 37 36 35 34 33 32</td> <td></td> </tr> </table>	b15	b0	SD715	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0		SD716	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16		SD717	47 46 45 44 43 42 41 40 39 38 37 36 35 34 33 32		S (durante l'esecuzione)	Nuovo	●
b15				b0													
SD715				15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0														
SD716	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16																	
SD717	47 46 45 44 43 42 41 40 39 38 37 36 35 34 33 32																	
SD716																		
SD717																		
SD718	Accumulatore	Accumulatore	Usato come sostituzione degli accumulatori usati nei programmi della serie A.	S/U	Nuovo													
SD719																		
SD720	Numero programma destinazione per istruzione PLOAD	Numero programma destinazione per istruzione PLOAD	Memorizza il numero del programma da caricare dall'istruzione PLOAD. Il valore può variare nel campo da 1 a 124.	U	Nuovo	CPU System Q												
SD730	Numero di aree di registrazione mancanti in una richiesta di comunicazione e CC-Link	da 0 a 32	Memorizza il numeri di registrazioni mancanti per una richiesta di comunicazione con la stazione intelligente collegata con A(1S)J61QBT61.	S (durante l'esecuzione)	Nuovo	CPU QnA												
SD736	Ingresso PKEY	Ingresso PKEY	Registro SD che memorizza temporaneamente l'ingresso da tastiera tramite l'istruzione PKEY.	S (durante l'esecuzione)	Nuovo	● (tranne Q00JCPU, Q00CPU, Q01CPU)												

(6) Registri relativi alle istruzioni

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valido per:																																																																																																
SD738 SD739 SD740 SD741 SD742 SD743 SD744 SD745 SD746 SD747 SD748 SD749 SD750 SD751 SD752 SD753 SD754 SD755 SD756 SD757 SD758 SD759 SD760 SD761 SD762 SD763 SD764 SD765 SD766 SD767 SD768 SD769	Memoria messaggio	Memoria messaggio	<p>Memorizza il messaggio emesso con l'istruzione MSG.</p> <div style="text-align: center;"> $b_{15} \longleftrightarrow b_8 \quad b_7 \longleftrightarrow b_0$ </div> <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <tr><td>SD738</td><td>Secondo carattere</td><td>Primo carattere</td></tr> <tr><td>SD739</td><td>Quarto carattere</td><td>Terzo carattere</td></tr> <tr><td>SD740</td><td>Sesto carattere</td><td>Quinto carattere</td></tr> <tr><td>SD741</td><td>Ottavo carattere</td><td>Settimo carattere</td></tr> <tr><td>SD742</td><td>Decimo carattere</td><td>Nono carattere</td></tr> <tr><td>SD743</td><td>Dodicesimo carattere</td><td>Undicesimo carattere</td></tr> <tr><td>SD744</td><td>Quattordicesimo carattere</td><td>Tredicesimo carattere</td></tr> <tr><td>SD745</td><td>Sedicesimo carattere</td><td>Quindicesimo carattere</td></tr> <tr><td>SD746</td><td>Diciottesimo carattere</td><td>Diciassettesimo carattere</td></tr> <tr><td>SD747</td><td>Ventesimo carattere</td><td>Diciannovesimo carattere</td></tr> <tr><td>SD748</td><td>Ventiduesimo carattere</td><td>Ventunesimo carattere</td></tr> <tr><td>SD749</td><td>Ventiquattresimo carattere</td><td>Ventitreesimo carattere</td></tr> <tr><td>SD750</td><td>Ventiseiesimo carattere</td><td>Venticinquesimo carattere</td></tr> <tr><td>SD751</td><td>Ventottesimo carattere</td><td>Ventisettesimo carattere</td></tr> <tr><td>SD752</td><td>Trentesimo carattere</td><td>Ventinovesimo carattere</td></tr> <tr><td>SD753</td><td>Trentaduesimo carattere</td><td>Trentunesimo carattere</td></tr> <tr><td>SD754</td><td>Trentaquattresimo carattere</td><td>Trentatreesimo carattere</td></tr> <tr><td>SD755</td><td>Trentaseiesimo carattere</td><td>Trentacinquesimo carattere</td></tr> <tr><td>SD756</td><td>Trentottesimo carattere</td><td>Trentasettesimo carattere</td></tr> <tr><td>SD757</td><td>Quarantesimo carattere</td><td>Trentanovesimo carattere</td></tr> <tr><td>SD758</td><td>Quarantaduesimo carattere</td><td>Quarantunesimo carattere</td></tr> <tr><td>SD759</td><td>Quarantaquattresimo carattere</td><td>Quarantatreesimo carattere</td></tr> <tr><td>SD760</td><td>Quarantaseiesimo carattere</td><td>Quarantacinquesimo carattere</td></tr> <tr><td>SD761</td><td>Quarantottesimo carattere</td><td>Quarantasettesimo</td></tr> <tr><td>SD762</td><td>Cinquantesimo carattere</td><td>Quarantanovesimo</td></tr> <tr><td>SD763</td><td>Cinquantaduesimo carattere</td><td>Cinquantunesimo carattere</td></tr> <tr><td>SD764</td><td>Cinquantaquattresimo carattere</td><td>Cinquantatreesimo carattere</td></tr> <tr><td>SD765</td><td>Cinquantaseiesimo carattere</td><td>Cinquantacinquesimo carattere</td></tr> <tr><td>SD766</td><td>Cinquantottesimo carattere</td><td>Cinquantasettesimo carattere</td></tr> <tr><td>SD767</td><td>Sessantesimo carattere</td><td>Cinquantanovesimo carattere</td></tr> <tr><td>SD768</td><td>Sessantaduesimo carattere</td><td>Sessantunesimo carattere</td></tr> <tr><td>SD769</td><td>Sessantaquattresimo carattere</td><td>Sessantatreesimo carattere</td></tr> </table>	SD738	Secondo carattere	Primo carattere	SD739	Quarto carattere	Terzo carattere	SD740	Sesto carattere	Quinto carattere	SD741	Ottavo carattere	Settimo carattere	SD742	Decimo carattere	Nono carattere	SD743	Dodicesimo carattere	Undicesimo carattere	SD744	Quattordicesimo carattere	Tredicesimo carattere	SD745	Sedicesimo carattere	Quindicesimo carattere	SD746	Diciottesimo carattere	Diciassettesimo carattere	SD747	Ventesimo carattere	Diciannovesimo carattere	SD748	Ventiduesimo carattere	Ventunesimo carattere	SD749	Ventiquattresimo carattere	Ventitreesimo carattere	SD750	Ventiseiesimo carattere	Venticinquesimo carattere	SD751	Ventottesimo carattere	Ventisettesimo carattere	SD752	Trentesimo carattere	Ventinovesimo carattere	SD753	Trentaduesimo carattere	Trentunesimo carattere	SD754	Trentaquattresimo carattere	Trentatreesimo carattere	SD755	Trentaseiesimo carattere	Trentacinquesimo carattere	SD756	Trentottesimo carattere	Trentasettesimo carattere	SD757	Quarantesimo carattere	Trentanovesimo carattere	SD758	Quarantaduesimo carattere	Quarantunesimo carattere	SD759	Quarantaquattresimo carattere	Quarantatreesimo carattere	SD760	Quarantaseiesimo carattere	Quarantacinquesimo carattere	SD761	Quarantottesimo carattere	Quarantasettesimo	SD762	Cinquantesimo carattere	Quarantanovesimo	SD763	Cinquantaduesimo carattere	Cinquantunesimo carattere	SD764	Cinquantaquattresimo carattere	Cinquantatreesimo carattere	SD765	Cinquantaseiesimo carattere	Cinquantacinquesimo carattere	SD766	Cinquantottesimo carattere	Cinquantasettesimo carattere	SD767	Sessantesimo carattere	Cinquantanovesimo carattere	SD768	Sessantaduesimo carattere	Sessantunesimo carattere	SD769	Sessantaquattresimo carattere	Sessantatreesimo carattere	S (durante l'esecuzione)	Nuovo	● (tranne Q00JCPU, Q00CPU, Q01CPU)
SD738			Secondo carattere	Primo carattere																																																																																																		
SD739			Quarto carattere	Terzo carattere																																																																																																		
SD740			Sesto carattere	Quinto carattere																																																																																																		
SD741			Ottavo carattere	Settimo carattere																																																																																																		
SD742			Decimo carattere	Nono carattere																																																																																																		
SD743			Dodicesimo carattere	Undicesimo carattere																																																																																																		
SD744			Quattordicesimo carattere	Tredicesimo carattere																																																																																																		
SD745			Sedicesimo carattere	Quindicesimo carattere																																																																																																		
SD746			Diciottesimo carattere	Diciassettesimo carattere																																																																																																		
SD747			Ventesimo carattere	Diciannovesimo carattere																																																																																																		
SD748			Ventiduesimo carattere	Ventunesimo carattere																																																																																																		
SD749			Ventiquattresimo carattere	Ventitreesimo carattere																																																																																																		
SD750			Ventiseiesimo carattere	Venticinquesimo carattere																																																																																																		
SD751			Ventottesimo carattere	Ventisettesimo carattere																																																																																																		
SD752			Trentesimo carattere	Ventinovesimo carattere																																																																																																		
SD753			Trentaduesimo carattere	Trentunesimo carattere																																																																																																		
SD754			Trentaquattresimo carattere	Trentatreesimo carattere																																																																																																		
SD755			Trentaseiesimo carattere	Trentacinquesimo carattere																																																																																																		
SD756			Trentottesimo carattere	Trentasettesimo carattere																																																																																																		
SD757			Quarantesimo carattere	Trentanovesimo carattere																																																																																																		
SD758			Quarantaduesimo carattere	Quarantunesimo carattere																																																																																																		
SD759			Quarantaquattresimo carattere	Quarantatreesimo carattere																																																																																																		
SD760			Quarantaseiesimo carattere	Quarantacinquesimo carattere																																																																																																		
SD761			Quarantottesimo carattere	Quarantasettesimo																																																																																																		
SD762			Cinquantesimo carattere	Quarantanovesimo																																																																																																		
SD763			Cinquantaduesimo carattere	Cinquantunesimo carattere																																																																																																		
SD764			Cinquantaquattresimo carattere	Cinquantatreesimo carattere																																																																																																		
SD765			Cinquantaseiesimo carattere	Cinquantacinquesimo carattere																																																																																																		
SD766			Cinquantottesimo carattere	Cinquantasettesimo carattere																																																																																																		
SD767			Sessantesimo carattere	Cinquantanovesimo carattere																																																																																																		
SD768			Sessantaduesimo carattere	Sessantunesimo carattere																																																																																																		
SD769			Sessantaquattresimo carattere	Sessantatreesimo carattere																																																																																																		
SD774 e SD775			Impostazione limite PID	0: Limite impostato 1: Limite non impostato	<p>● Specifica il limite per ciascun loop PID come segue:</p> <div style="text-align: center;"> $b_{15} \longleftrightarrow b_1 \quad b_0$ </div> <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <tr> <td style="width: 25%;">SD774</td> <td style="width: 25%;">Loop 16</td> <td style="width: 25%;">Loop 2</td> <td style="width: 25%;">Loop 1</td> </tr> <tr> <td>SD775</td> <td>Loop 32</td> <td>Loop 18</td> <td>Loop 17</td> </tr> </table>	SD774	Loop 16	Loop 2	Loop 1	SD775	Loop 32	Loop 18	Loop 17	U	Nuovo	CPU System Q (tranne Q00JCPU, Q00CPU, Q01CPU)																																																																																						
SD774	Loop 16	Loop 2	Loop 1																																																																																																			
SD775	Loop 32	Loop 18	Loop 17																																																																																																			
SD780	Numero restante di esecuzione simultanea istruzioni dedicate CC-Link	da 0 a 32	<p>● Memorizza il numero restante di istruzioni dedicate CC-Link in esecuzione simultanea</p>	U	Nuovo	CPU QnA																																																																																																
da SD781 a SD793	Maschera istruzione IMASK	Maschera	<p>● Memorizza la maschera usata dall'istruzione IMASK come segue:</p> <div style="text-align: center;"> $b_{15} \quad b_{11} \quad b_0$ </div> <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 10px;"> <tr> <td style="width: 25%;">SD781</td> <td style="width: 25%;">I63</td> <td style="width: 25%;">I59</td> <td style="width: 25%;">I48</td> </tr> <tr> <td>SD782</td> <td>I79</td> <td>I65</td> <td>I64</td> </tr> <tr> <td>SD793</td> <td>I255</td> <td>I241</td> <td>I240</td> </tr> </table>	SD781	I63	I59	I48	SD782	I79	I65	I64	SD793	I255	I241	I240	S (durante l'esecuzione)	Nuovo	CPU System Q																																																																																				
SD781	I63	I59	I48																																																																																																			
SD782	I79	I65	I64																																																																																																			
SD793	I255	I241	I240																																																																																																			
<p>Solo i registri speciali da SD781 (I48) a SD785 (I127) sono disponibili per le CPU Q00J, Q00 o Q01.</p>																																																																																																						

Tabella dei registri speciali

(7) Debug

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valido per:																
SD806	Nome file memoria stati	Nome file memoria stati	Memorizza il nome del file in codice ASCII (senza estensione) nel momento in cui è stata eseguita la memorizzazione. b15 b8 b7 b0 SD806 Secondo carattere Primo carattere SD807 Quarto carattere Terzo carattere SD808 Sesto carattere Quinto carattere SD809 Ottavo carattere Settimo carattere SD810 Primo carattere estensione 2EH (.) SD811 Terzo carattere estensione Secondo carattere estensione	S (durante l'esecuzione)	Nuovo	CPU QnA																
SD807																						
SD808																						
SD809																						
SD810																						
SD811																						
SD812	Passo memorizzazione e stato	Passo memorizzazione stato	Memorizza il numero di passo nel momento in cui è avvenuta la memorizzazione. SD812 Pattern* SD813 Block No. SD814 Step No. / transition No. SD815 Sequence step No. (L) SD816 Sequence step No. (H)	S (durante l'esecuzione)	D9055 modifica formato	CPU QnA																
SD813																						
SD814																						
SD815																						
SD816																						
SD816			* Contents of pattern data <table border="1"> <tr> <td>15</td><td>14</td><td>to</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td> </tr> <tr> <td>0</td><td>0</td><td>to</td><td>0</td><td>0</td><td>*</td><td>*</td><td>*</td> </tr> </table> ← Bit No. SFC block designation present (1)/absent (0) SFC block designation present (1)/absent (0) SFC transition designation present (1)/absent (0)	15	14	to	4	3	2	1	0	0	0	to	0	0	*	*	*			
15	14	to	4	3	2	1	0															
0	0	to	0	0	*	*	*															

Tabella dei registri speciali

(8) Area retentiva

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valido per:																																																
SD900	Numero drive durante mancanza rete	Numero del drive in uso nel momento della mancanza rete	Memorizza il numero di drive del file in uso al momento della mancanza rete.	S (variazione di stato)	Nuovo																																																	
SD901	Nome file attivo durante mancanza rete	Nome del file in uso durante mancanza rete	Memorizza in codice ASCII il nome (con estensione) del file a cui si faceva accesso durante la mancanza rete.	S (variazione di stato)	Nuovo																																																	
SD902																																																						
SD903																																																						
SD904																																																						
SD905																																																						
SD906																																																						
			<table style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: right;">b15</td> <td style="text-align: center;">b8 b7</td> <td style="text-align: left;">b0</td> </tr> <tr> <td>SD901</td> <td style="border: 1px solid black; text-align: center;">Secondo carattere</td> <td style="border: 1px solid black; text-align: center;">Primo carattere</td> </tr> <tr> <td>SD902</td> <td style="border: 1px solid black; text-align: center;">Quarto carattere</td> <td style="border: 1px solid black; text-align: center;">Terzo carattere</td> </tr> <tr> <td>SD903</td> <td style="border: 1px solid black; text-align: center;">Sesto carattere</td> <td style="border: 1px solid black; text-align: center;">Quinto carattere</td> </tr> <tr> <td>SD904</td> <td style="border: 1px solid black; text-align: center;">Ottavo carattere</td> <td style="border: 1px solid black; text-align: center;">Settimo carattere</td> </tr> <tr> <td>SD905</td> <td style="border: 1px solid black; text-align: center;">Primo carattere estensione</td> <td style="border: 1px solid black; text-align: center;">2EH (.)</td> </tr> <tr> <td>SD906</td> <td style="border: 1px solid black; text-align: center;">Terzo carattere estensione</td> <td style="border: 1px solid black; text-align: center;">Secondo carattere estensione</td> </tr> </table>	b15	b8 b7	b0	SD901	Secondo carattere	Primo carattere	SD902	Quarto carattere	Terzo carattere	SD903	Sesto carattere	Quinto carattere	SD904	Ottavo carattere	Settimo carattere	SD905	Primo carattere estensione	2EH (.)	SD906	Terzo carattere estensione	Secondo carattere estensione																														
b15	b8 b7	b0																																																				
SD901	Secondo carattere	Primo carattere																																																				
SD902	Quarto carattere	Terzo carattere																																																				
SD903	Sesto carattere	Quinto carattere																																																				
SD904	Ottavo carattere	Settimo carattere																																																				
SD905	Primo carattere estensione	2EH (.)																																																				
SD906	Terzo carattere estensione	Secondo carattere estensione																																																				
SD910	Ingresso RKEY	Ingresso RKEY	Memorizzato nella sequenza in cui i codice dei tasti sono stati inseriti.	S (durante l'esecuzione)	Nuovo	CPU QnA																																																
SD911																																																						
SD912																																																						
SD913																																																						
SD914																																																						
SD915																																																						
SD916																																																						
SD917																																																						
SD918																																																						
SD919																																																						
SD920																																																						
SD921																																																						
SD922																																																						
SD923																																																						
SD924																																																						
SD925																																																						
			<table style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: right;">b15</td> <td style="text-align: center;">b8 b7</td> <td style="text-align: left;">b0</td> </tr> <tr> <td>SD910</td> <td style="border: 1px solid black; text-align: center;">Secondo carattere</td> <td style="border: 1px solid black; text-align: center;">Primo carattere</td> </tr> <tr> <td>SD911</td> <td style="border: 1px solid black; text-align: center;">Quarto carattere</td> <td style="border: 1px solid black; text-align: center;">Terzo carattere</td> </tr> <tr> <td>SD912</td> <td style="border: 1px solid black; text-align: center;">Sesto carattere</td> <td style="border: 1px solid black; text-align: center;">Quinto carattere</td> </tr> <tr> <td>SD913</td> <td style="border: 1px solid black; text-align: center;">Ottavo carattere</td> <td style="border: 1px solid black; text-align: center;">Settimo carattere</td> </tr> <tr> <td>SD914</td> <td style="border: 1px solid black; text-align: center;">Decimo carattere</td> <td style="border: 1px solid black; text-align: center;">Nono carattere</td> </tr> <tr> <td>SD915</td> <td style="border: 1px solid black; text-align: center;">Dodicesimo carattere</td> <td style="border: 1px solid black; text-align: center;">Undicesimo carattere</td> </tr> <tr> <td>SD916</td> <td style="border: 1px solid black; text-align: center;">Quattordicesimo carattere</td> <td style="border: 1px solid black; text-align: center;">Tredicesimo carattere</td> </tr> <tr> <td>SD917</td> <td style="border: 1px solid black; text-align: center;">Sedicesimo carattere</td> <td style="border: 1px solid black; text-align: center;">Quindicesimo carattere</td> </tr> <tr> <td>SD918</td> <td style="border: 1px solid black; text-align: center;">Diciottesimo carattere</td> <td style="border: 1px solid black; text-align: center;">Diciassettesimo carattere</td> </tr> <tr> <td>SD919</td> <td style="border: 1px solid black; text-align: center;">Ventesimo carattere</td> <td style="border: 1px solid black; text-align: center;">Diciannovesimo carattere</td> </tr> <tr> <td>SD920</td> <td style="border: 1px solid black; text-align: center;">Ventiduesimo carattere</td> <td style="border: 1px solid black; text-align: center;">Ventunesimo carattere</td> </tr> <tr> <td>SD921</td> <td style="border: 1px solid black; text-align: center;">Ventiquattresimo carattere</td> <td style="border: 1px solid black; text-align: center;">Ventitreesimo carattere</td> </tr> <tr> <td>SD922</td> <td style="border: 1px solid black; text-align: center;">Ventesiesimo carattere</td> <td style="border: 1px solid black; text-align: center;">Venticinquesimo carattere</td> </tr> <tr> <td>SD923</td> <td style="border: 1px solid black; text-align: center;">Ventottesimo carattere</td> <td style="border: 1px solid black; text-align: center;">Ventisettesimo carattere</td> </tr> <tr> <td>SD924</td> <td style="border: 1px solid black; text-align: center;">Trentesimo carattere</td> <td style="border: 1px solid black; text-align: center;">Ventinovesimo carattere</td> </tr> <tr> <td>SD925</td> <td style="border: 1px solid black; text-align: center;">Trentaduesimo carattere</td> <td style="border: 1px solid black; text-align: center;">Trentunesimo carattere</td> </tr> </table>	b15	b8 b7	b0	SD910	Secondo carattere	Primo carattere	SD911	Quarto carattere	Terzo carattere	SD912	Sesto carattere	Quinto carattere	SD913	Ottavo carattere	Settimo carattere	SD914	Decimo carattere	Nono carattere	SD915	Dodicesimo carattere	Undicesimo carattere	SD916	Quattordicesimo carattere	Tredicesimo carattere	SD917	Sedicesimo carattere	Quindicesimo carattere	SD918	Diciottesimo carattere	Diciassettesimo carattere	SD919	Ventesimo carattere	Diciannovesimo carattere	SD920	Ventiduesimo carattere	Ventunesimo carattere	SD921	Ventiquattresimo carattere	Ventitreesimo carattere	SD922	Ventesiesimo carattere	Venticinquesimo carattere	SD923	Ventottesimo carattere	Ventisettesimo carattere	SD924	Trentesimo carattere	Ventinovesimo carattere	SD925	Trentaduesimo carattere	Trentunesimo carattere
b15	b8 b7	b0																																																				
SD910	Secondo carattere	Primo carattere																																																				
SD911	Quarto carattere	Terzo carattere																																																				
SD912	Sesto carattere	Quinto carattere																																																				
SD913	Ottavo carattere	Settimo carattere																																																				
SD914	Decimo carattere	Nono carattere																																																				
SD915	Dodicesimo carattere	Undicesimo carattere																																																				
SD916	Quattordicesimo carattere	Tredicesimo carattere																																																				
SD917	Sedicesimo carattere	Quindicesimo carattere																																																				
SD918	Diciottesimo carattere	Diciassettesimo carattere																																																				
SD919	Ventesimo carattere	Diciannovesimo carattere																																																				
SD920	Ventiduesimo carattere	Ventunesimo carattere																																																				
SD921	Ventiquattresimo carattere	Ventitreesimo carattere																																																				
SD922	Ventesiesimo carattere	Venticinquesimo carattere																																																				
SD923	Ventottesimo carattere	Ventisettesimo carattere																																																				
SD924	Trentesimo carattere	Ventinovesimo carattere																																																				
SD925	Trentaduesimo carattere	Trentunesimo carattere																																																				

Tabella dei registri speciali

(9) Moduli con fusibile interrotto

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valido per:
SD1300	Modulo con fusibile interrotto	Maschere di bit in gruppi da 16 punti, che indicano i moduli con fusibili interrotti 0 : Fusibile non interrotto 1 : Fusibile interrotto presente	<p>Il numero dei moduli di uscita con fusibili interrotti è inserito come una maschera di bit in gruppi da 16.</p> <p>(Se i numeri di modulo sono impostati con parametri, vengono memorizzati i numeri definiti dalla parametrizzazione).</p> <p>Vengono rilevati anche gli stati di fusibile interrotto nei moduli di uscita delle stazioni remote.</p>	S (Errore)	D9100	● (tranne Q00JCPU, Q00CPU, Q01CPU)
SD1301					D9101	
SD1302					D9102	
SD1303					D9103	
SD1304					D9104	
SD1305					D9105	
SD1306					D9106	
SD1307					D9107	
SD1308					Nuovo	
SD1309 ? SD1330					Nuovo	
SD1331					Nuovo	
da SD1350 a SD1381	Alimentazione esterna scollegata	Maschere di bit in gruppi di 16 punti, che indicano i moduli su cui è stata scollegata l'alimentazione esterna 0 : Alimentazione esterna scollegata 1 : Alimentazione esterna non scollegata	<p>● I numeri dei moduli di uscita su cui è stata scollegata l'alimentazione esterna sono memorizzati come maschere di bit (in gruppi a 16 punti). (Se i numeri di modulo sono impostati con parametri, vengono memorizzati i numeri definiti dalla parametrizzazione).</p>	S (Errore)	Nuovo	CPU System Q (tranne Q00JCPU, Q00CPU, Q01CPU)

(10) Verifica moduli di I/O

Numero	Nome	Significato	Descrizione	Impostato da (se impostato)	ACPU EEPROM D9 [] [] []	Valido per:																																																																																				
SD1400	Modulo di I/O errore di verifica	Maschere di bit, in gruppi da 16 punti, che indicano i moduli con errori di verifica. 0 : Nessun errore di verifica I/O 1 : Errore verifica I/O presente	All'accensione, i numero dei moduli di I/O le cui informazioni differiscono da quelle registrate vengono registrati in questo registro (in gruppi da 16 punti). (Se i numeri di modulo sono impostati con parametri, vengono memorizzati i numeri definiti dalla parametrizzazione). Rileva anche le informazioni dei moduli di I/O. . <div style="text-align: center;"> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td></td> <td style="font-size: small;">b15</td><td style="font-size: small;">b14</td><td style="font-size: small;">b13</td><td style="font-size: small;">b12</td><td style="font-size: small;">b11</td><td style="font-size: small;">b10</td><td style="font-size: small;">b9</td><td style="font-size: small;">b8</td><td style="font-size: small;">b7</td><td style="font-size: small;">b6</td><td style="font-size: small;">b5</td><td style="font-size: small;">b4</td><td style="font-size: small;">b3</td><td style="font-size: small;">b2</td><td style="font-size: small;">b1</td><td style="font-size: small;">b0</td> </tr> <tr> <td style="font-size: small;">SD1400</td> <td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td> </tr> <tr> <td style="font-size: small;">SD1401</td> <td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td> </tr> <tr> <td style="font-size: small;">↓</td> <td style="text-align: center;">⋮</td><td style="text-align: center;">⋮</td><td style="text-align: center;">⋮</td><td style="text-align: center;">⋮</td><td style="text-align: center;">⋮</td><td style="text-align: center;">⋮</td><td style="text-align: center;">⋮</td><td style="text-align: center;">⋮</td><td style="text-align: center;">⋮</td><td style="text-align: center;">⋮</td><td style="text-align: center;">⋮</td><td style="text-align: center;">⋮</td><td style="text-align: center;">⋮</td><td style="text-align: center;">⋮</td><td style="text-align: center;">⋮</td> </tr> <tr> <td style="font-size: small;">SD1431</td> <td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td> </tr> </table> <p style="text-align: center; font-size: small;">↑ Indication of I/O module with verification error</p> </div>		b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	SD1400	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	SD1401	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	↓	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	SD1431	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	S (Errore)	D9116	● (tranne Q00JCPU, Q00CPU, Q01CPU)
				b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																							
SD1400				0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0																																																																							
SD1401				1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0																																																																							
↓				⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮																																																																								
SD1431				0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0																																																																							
SD1401				D9117																																																																																						
SD1402				D9118																																																																																						
SD1403				D9119																																																																																						
SD1404				D9120																																																																																						
SD1405				D9121																																																																																						
SD1406	D9122																																																																																									
SD1407	D9123																																																																																									
SD1408	Nuovo																																																																																									
da SD1409 a SD1430	Nuovo																																																																																									
SD1431	Nuovo																																																																																									

Tabella dei registri speciali

(11) Corrispondenza conversione da serie A MELSEC a serie Q/System Q MELSEC

Nelle conversioni da serie A a serie Q o System Q, i registri speciali da D9000 a D9255 (serie A) corrispondono ai registri speciali diagnostici da SD1000 a SD1255 della serie Q e System Q.

Questi registri speciali diagnostici sono tutti impostati dal sistema e non possono essere modificati dal programma utente, Gli utenti che necessitano di modificare questi registri devono modificare i loro programmi in modo da utilizzare solo registri speciali diagnostici System Q/QnA. Una eccezione è rappresentata dai registri speciali da D9200 a D9255. I dati di questi registri possono essere modificati dall'utente. Per questo, l'utente può modificare i dati nei registri speciali diagnostici da SD1200 a SD1255 dopo la conversione.

Fare riferimento ai manuali delle CPU e delle reti MELSECNET e MELSECNET/B per informazioni dettagliate sui registri speciali della serie A.

NOTA

Nel caso di indirizzi di operandi per cui viene indicata una equivalenza con i registri speciali diagnostici System Q/QnA, modificare l'indirizzo per farlo corrispondere a quello valido per le CPU System Q/QnA. Se non è indicato nessun registro speciale diagnostico System Q/QnA, può essere utilizzato il registro speciale dopo la conversione.

Tabella registri speciali e registri diagnostici

Registro speciale CPU serie A	Registro speciale dopo la conversione	Registro speciale diagnostico equivalente System Q/QnA	Nome	Significato	Valido per:
D9000	SD1000	—	Modulo con fusibile	Numero del modulo con fusibile interrotto	System Q CPU QnA
D9001	SD1001	—	Modulo con fusibile	Numero del modulo con fusibile interrotto	
D9002	SD1002	—	Errore verifica modulo di I/O	Numero modulo con errore di verifica I/O	
D9004	SD1004	—	Errori comunicazione MINI	Memorizza lo stato delle impostazioni di configurazione (moduli da 1 a 8)	CPU QnA
D9005	SD1005	—	Contatore AC DOWN	Numero di occorrenze AC DOWN (mancanza rete)	System Q CPU QnA
D9008	SD1008	SD0	Errore di autodiagnosi	Numero errore di autodiagnosi	
D9009	SD1009	SD62	Rilevamento spia di errore	Numero F su cui si è verificata l'anomalia esterna	
D9010	SD1010	Nessuna funzione per CPU System Q/QnA	Passo in errore	Numero di passo in cui si è verificato l'errore di esecuzione.	
D9011	SD1011		Passo in errore	Numero di passo in cui si è verificato l'errore di esecuzione.	
D9014	SD1014		Modo di controllo I/O	Numero modo di controllo I/O	
D9015	SD1015	SD203	Stato operativo della CPU	Stato operativo della CPU	
D9016	1016	Nessuna funzione per CPU System Q/QnA	Numero programma	Memorizza il numero del programma di sequenza in esecuzione come valore binario	
D9017	SD1017	SD520	Tempo di scansione	Minimo tempo di scansione (in unità da 10 ms)	
D9018	SD1018	SD524	Tempo di scansione	Tempo di scansione (unità da 10 ms)	
D9019	SD1019	SD526	Tempo di scansione	Massimo tempo di scansione (unità da 10 ms)	
D9020	SD1020	Nessuna funzione per CPU System Q/QnA	Scansione costante	Tempo di scansione costante (impostato da utente in unità da 10 ms)	

Tabella dei registri speciali

Tabella registri speciali e registri diagnostici

Registro speciale CPU serie A	Registro speciale dopo la conversione	Registro speciale diagnostico equivalente System Q/QnA	Nome	Significato	Valido per:
D9021	SD1021	—	Tempo di scansione	Tempo di scansione (unitr da 1 ms)	System Q CPU QnA
D9022	SD1022	SD412	Tempo	Tempo	
D9025	SD1025	SD210	Dati orologio	Dati orologio (anno, mese)	
D9026	SD1026	SD211	Dati orologio	Dati orologio (giorno, ora)	
D9027	SD1027	SD212	Dati orologio	Dati orologio (minuto, secondo)	
D9028	SD1028	SD213	Dati orologio	Dati orologio (giorno della settimana)	
D9035	SD1035	SD648	File registri di espansione	Usa numero blocco	
D9036	SD1036	Nessuna funzione per CPU System Q/QnA	File registri di espansione per indicazione indirizzo operando	Indirizzo operando quando vengono indirizzati direttamente singoli operandi in un file registri di espansione.	
D9037	SD1037				
D9038	SD1038	SD207	Prioritr visualizzazione LED di errore	Prioritr da 1 a 4	
D9039	SD1039	SD208		Prioritr da 5 a 7	
D9044	SD1044	Nessuna funzione per CPU System Q/QnA	Per traccia di campionamento	Passo o tempo durante la traccia	
D9049	SD1049		Area di lavoro per SFC	Numero blocco del file registri di espansione	
D9050	SD1050		Numero errore del programma SFC	Codice di errore generato dal programma SFC	
D9051	SD1051		Blocco in errore	Numero di blocco in cui si r verificato l'errore	
D9052	SD1052		Passo in errore	Numero di passo in cui si r verificato l'errore	
D9053	SD1053		Transizione in errore	Numero della condizione di transizione in cui si r verificato l'errore	
D9054	SD1054		Errore su passo SFC	Numero del passo SFC in cui si r verificato l'errore	
D9055	SD1055		SD812	Memoria di stato	
D9060	SD1060	SD392	Versione software	Versione software del software interno	
D9072	SD1072	Nessuna funzione per CPU System Q/QnA	Controllo comunicazioni PC	Controllo linea di comunicazione con PC	System Q CPU QnA
D9081	SD1081	SD714	Numero di blocchi vuoti nell'area di registrazione della richiesta di comunicazione	Numero di blocchi vuoti nell'area di registrazione della richiesta di comunicazione	
D9085	SD1085	Nessuna funzione per CPU System Q/QnA	Registrazione valore per intervallo di test	Valore di default 10 s	
D9090	SD1090		AnN: Area indirizzamento sottoprogrammi per programma microcomputer AnA:Nessuna funzione QnA:Numero di moduli funzione speciali	AnN: Area indirizzamento sottoprogrammi per programma microcomputer AnA:Nessuna funzione QnA:Numero di moduli funzione speciali	
D9091	SD1091		Codice di errore dettagliato	Codice di errore dettagliato autodiagnosi	
D9094	SD9094		SD251	Indirizzo iniziale I/O del modulo da sostituire	

Tabella dei registri speciali

Tabella registri speciali e registri diagnostici

Registro speciale CPU serie A	Registro speciale dopo la conversione	Registro speciale diagnostico equivalente System Q/QnA	Nome	Significato	Valido per:
D9100	SD1100	—	Fusibile del modulo interrotto	Maschere di bit in gruppi da 16 punti, che indicano i moduli con fusibili interrotti	System Q CPU QnA
D9101	SD1101				
D9102	SD1102				
D9103	SD1103				
D9104	SD1104				
D9105	SD1105				
D9106	SD1106				
D9107	SD1107				
D9108	SD1108	—	Impostazione timer sorveglianza transizione di passo	Imposta nel byte leggero il valore del timer di sorveglianza delle transizioni di passo. Il valore impostato può variare nel campo da 1 a 255 secondi. Definisce nel byte pesante il numero F da attivare allo scadere del tempo di sorveglianza. Il timer di sorveglianza viene avviato quando uno dei relé da SM1108 a SM1114 si attiva. Se la condizione di transizione per un passo corrispondente al timer non si verifica entro il tempo impostato, la spia di errore (F) viene attivata.	
D9109	SD1109				
D9110	SD1110				
D9111	SD1111				
D9112	SD1112				
D9113	SD1113				
D9114	SD1114				
D9116	SD1116	—	Errore verifica modulo di I/O	Maschere di bit, in gruppi da 16 punti, che indicano i moduli con errori di verifica.	
D9117	SD1117				
D9118	SD1118				
D9119	SD1119				
D9120	SD1120				
D9121	SD1121				
D9122	SD1122				
D9123	SD1123				
D9124	SD9124	SD63	Numero spie di errore	Numero spie di errore attive	
D9125	SD9125	SD64	Numero spie di errore rilevate	Numero spie di errore rilevate	
D9126	SD9126	SD65			
D9127	SD9127	SD66			
D9128	SD9128	SD67			
D9129	SD9129	SD68			
D9130	SD9130	SD69			
D9131	SD9131	SD70			
D9132	SD9132	SD71			

Tabella dei registri speciali

Tabella registri speciali e registri diagnostici

Registro speciale CPU serie A	Registro speciale dopo la conversione	Registro speciale diagnostico equivalente System Q/QnA	Nome	Significato	Valido per:
D9200	SD1200	—	Risultato esecuzione LRDP	0: Completamento normale 2: Errore impostazione istruzione LRDP 3: Errore sulla stazione interessata 4: Esecuzione LRDP su stazione interessata disabilitata	CPU QnA
D9201	SD1201	—	Risultato esecuzione LWTP	0: Completamento normale 2: Errore impostazione istruzione LRDP 3: Errore sulla stazione interessata 4: Esecuzione LWTP su stazione interessata disabilitata	
D9202	SD1202	—	Tipo collegamento stazione locale	Memorizza lo stato per gli indirizzi da 1 a 16	
D9203	SD1203	—		Memorizza lo stato per gli indirizzi da 17 a 32	
D9241	SD1241	—		Memorizza lo stato per gli indirizzi da 33 a 48	
D9242	SD1242	—		Memorizza lo stato per gli indirizzi da 49 a 64	
D9204	SD1204	—	Stato comunicazione	0: Loop avanti, durante la comunicazione 1: Loop indietro, durante la comunicazione 2: Loopback implementato nei loop avanti/indietro 3: Loopback implementato nel loop avanti 4: Loopback implementato nel loop indietro 5: Collegamento disabilitato	
D9205	SD1205	—	Stazioni che implementano il loopback	Stazione che implementa il loopback avanti	
D9206	SD1206	—	Stazioni che implementano il loopback	Stazione che implementa il loopback avanti	
D9207	SD1207	—	Tempo di scansione del collegamento	Valore massimo	
D9208	SD1208	—		Valore minimo	
D9209	SD1209	—		Valore attuale	
D9210	SD1210	—	Numero tentativi	Memorizzato come valore cumulativo	
D9211	SD1211	—	Numero commutazioni di loop	Memorizzato come valore cumulativo	
D9212	SD1212	—	Stato operativo stazione locale	Memorizza lo stato per le stazioni da 1 a 16	
D9213	SD1213	—		Memorizza lo stato per le stazioni da 17 a 32	
D9214	SD1214	—		Memorizza lo stato per le stazioni da 33 a 48	
D9215	SD1215	—		Memorizza lo stato per le stazioni da 49 a 64	

Tabella dei registri speciali

Tabella registri speciali e registri diagnostici

Registro speciale CPU serie A	Registro speciale dopo la conversione	Registro speciale diagnostico equivalente System Q/QnA	Nome	Significato	Valido per:
D9216	SD1216	—	Stato rilevamento errori stazione locale	Memorizza lo stato per le stazioni da 1 a 16	CPU QnA
D9217	SD1217	—		Memorizza lo stato per le stazioni da 17 a 32	
D9218	SD1218	—		Memorizza lo stato per le stazioni da 33 a 48	
D9219	SD1219	—		Memorizza lo stato per le stazioni da 49 a 64	
D9220	SD1220	—	Parametri stazione locale non conforme ; remota Stazione di I/O Errore allocazione I/O	Memorizza lo stato per le stazioni da 1 a 16	
D9221	SD1221	—		Memorizza lo stato per le stazioni da 17 a 32	
D9222	SD1222	—		Memorizza lo stato per le stazioni da 33 a 48	
D9223	SD1223	—		Memorizza lo stato per le stazioni da 49 a 64	
D9224	SD1224	—	Comunicazioni iniziali stazione locale e stazione di I/O remoto in corso	Memorizza lo stato per le stazioni da 1 a 16	
D9225	SD1225	—		Memorizza lo stato per le stazioni da 17 a 32	
D9226	SD1226	—		Memorizza lo stato per le stazioni da 33 a 4	
D9227	SD1227	—		Memorizza lo stato per le stazioni da 49 a 64	
D9228	SD1228	—	Errore su stazione locale o stazione di I/O remoto	Memorizza lo stato per le stazioni da 1 a 16	
D9229	SD1229	—		Memorizza lo stato per le stazioni da 17 a 32	
D9230	SD1230	—		Memorizza lo stato per le stazioni da 33 a 48	
D9231	SD1231	—		Memorizza lo stato per le stazioni da 49 a 64	

Tabella dei registri speciali

Tabella registri speciali e registri diagnostici

Registro speciale CPU serie A	Registro speciale dopo la conversione	Registro speciale diagnostico equivalente System Q/QnA	Nome	Significato	Valido per:
D9232	SD1232	—	Errore di loop su stazione locale o stazione di I/O remoto	Memorizza lo stato per le stazioni da 1 a 8	CPU QnA
D9233	SD1233	—		Memorizza lo stato per le stazioni da 9 a 16	
D9234	SD1234	—		Memorizza lo stato per le stazioni da 17 a 24	
D9235	SD1235	—		Memorizza lo stato per le stazioni da 25 a 32	
D9236	SD1236	—		Memorizza lo stato per le stazioni da 33 a 40	
D9237	SD1237	—		Memorizza lo stato per le stazioni da 41 a 48	
D9238	SD1238	—		Memorizza lo stato per le stazioni da 49 a 56	
D9239	SD1239	—		Memorizza lo stato per le stazioni da 57 a 64	
D9240	SD1240	—	Numero errori di comunicazione rilevati	Memorizza la sommatoria di tutti gli errori di ricezione	
D9243	SD1243	—	Informazione numero di stazione per stazione host	Memorizza il numero di stazione (da 0 a 64)	
D9244	SD1244	—	Numero stazioni slave della rete	Memorizza il numero di stazioni Slave	
D9245	SD1245	—	Numero errori di comunicazione rilevati	Memorizza la sommatoria di tutti gli errori di ricezione	
D9248	SD1248	—	Stato operativo stazione locale	Memorizza lo stato per le stazioni da 1 a 16	
D9249	SD1249	—		Memorizza lo stato per le stazioni da 17 a 32	
D9250	SD1250	—		Memorizza lo stato per le stazioni da 33 a 48	
D9251	SD1251	—		Memorizza lo stato per le stazioni da 49 a 64	
D9252	SD1252	—	Stato errori stazione locale	Memorizza lo stato per le stazioni da 1 a 16	
D9253	SD1253	—		Memorizza lo stato per le stazioni da 17 a 32	
D9254	SD1254	—		Memorizza lo stato per le stazioni da 33 a 48	
D9255	SD1255	—		Memorizza lo stato per le stazioni da 49 a 64	

Tabella dei registri speciali

A.5.2 Tabella registri speciali (D) (solo serie A)

I registri speciali (D) sono registri dati previsti per funzioni specifiche nella CPU. Per questa ragione il programma non deve scrivere nei registri speciali (tranne per i registri contrassegnati con **❶**).

Esistono in generale due tipi di registri speciali:

- Registri speciali scritti direttamente dalla CPU, che possono solo essere letti (e azzerati) dall'utente.
- I registri speciali possono essere sovrascritti solo in determinate circostanze.

L'uso dei registri speciali in un programma di sequenza deve essere controllato adeguatamente.

La tabella seguente contiene una panoramica dei registri speciali per tutte le serie MELSEC, insieme ad una breve descrizione della loro funzione.

Tabella registri speciali (serie A)

Numero	Significato	Descrizione	Dettagli	CPU
D9000	Modulo con fusibile	Numero del modulo con fusibile interrotto	Quando vengono rilevati moduli con fusibile interrotto, l'indirizzo della prima unità viene memorizzato in esadecimale (esempio: se si interrompono i fusibili dei moduli di uscita da Y50 a Y6F, viene memorizzato il numero esadecimale "50"). La memorizzazione dell'indirizzo del modulo e la visualizzazione su un dispositivo di programmazione è in codice esadecimale. Il registro viene cancellato quando il contenuto di tutti i registri da D9100 a D9107 è azzerato. Il controllo dei fusibili interrotti viene eseguito anche sui moduli di uscita delle stazioni di I/O remoto.	
D9002	Errore verifica modulo di I/O	Indirizzo del modulo di I/O con errori di verifica	Se vengono rilevati moduli di I/O i cui dati sono differenti da quelli rilevati all'accensione, l'indirizzo iniziale di I/O del primo modulo rilevato viene memorizzato in codice esadecimale. (Il metodo di memorizzazione è lo stesso di D9000). La memorizzazione dell'indirizzo del modulo e la visualizzazione su un dispositivo di programmazione è in codice esadecimale. Il registro viene cancellato quando il contenuto di tutti i registri da D9116 a D9123 è azzerato. Il controllo sui dati dei moduli di I/O viene eseguito anche sui moduli delle stazioni di I/O remoto.	
❶ D9004	Errori modulo di comunicazione master MINI	Stato rilevamento errori	Viene memorizzato lo stato degli errori della rete MINI(S3) rilevato sul modulo AJ71PT32(S3) caricato. Fare riferimento al manuale MELSECNET/MINI-S3 per ulteriori informazioni.	Solo per CPU AnA, AnAS e AnU
❶ D9005	Contatore AC DOWN	Conteggio AC DOWN	Viene sommato 1 ogni volta che la tensione d'ingresso diventa inferiore al 80% del valore nominale durante il funzionamento della CPU, e il valore viene memorizzato in formato BIN.	
❶ D9008	Errore di autodiagnosi	Numero errore di autodiagnosi	Se viene rilevato un errore come risultato di una autodiagnosi, il numero di errore viene memorizzato in codice BIN.	Solo per CPU AnS e A2C
D9009	Rilevamento spia di errore	Numero F corrispondente all'errore esterno	Se uno dei flag da F0 a F255 viene attivato con una istruzione OUT F o SET F, il primo numero F fra quelli attivi viene memorizzato come codice BIN. D9009 può essere azzerato da una istruzione RST F o LEDR. Se è stato rilevato un altro numero F, l'azzeramento di D9009 provoca la comparsa del numero successivo in D9009.	Non per CPU A3N CPU A3M CPU A3A CPU A3H
			Se uno dei flag da F0 a F255 viene attivato con una istruzione OUT F o SET F, il primo numero F fra quelli attivi viene memorizzato come codice BIN. D9009 può essere azzerato eseguendo una istruzione RST F o LEDR, oppure spostando l'interruttore INDICATOR RESET sul frontalino della CPU in posizione ON. Se è stato rilevato un altro numero F, l'azzeramento di D9009 provoca la comparsa del numero successivo in D9009.	Solo per CPU CPU A3N CPU A3M CPU A3A CPU A3H
D9010	Passo in errore	Numero di passo in cui si è verificato l'errore di esecuzione.	Se viene rilevato un errore di esecuzione durante una istruzione applicativa, il numero di passo nel quale si è verificato l'errore viene memorizzato in codice BIN. Successivamente, il contenuto di D9010 cambia ogni volta che si verifica un errore di esecuzione.	Non per CPU A3H, CPU A3M
❶ D9011	Passo in errore	Numero di passo in cui si è verificato l'errore di esecuzione.	Se viene rilevato un errore di esecuzione durante una istruzione applicativa, il numero di passo nel quale si è verificato l'errore viene memorizzato in codice BIN. Dato che la scrittura in D9011 viene eseguita quando M9011 passa da OFF a ON, il contenuto di D9010 non può essere modificato a meno che M9011 non venga azzerato da programma.	

Tabella registri speciali (serie A)

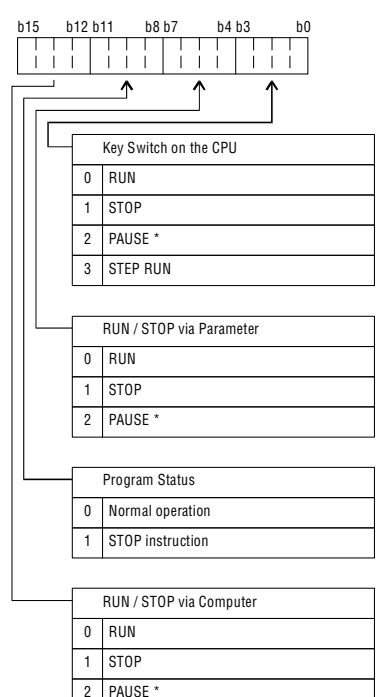
Numero	Significato	Descrizione	Dettagli	CPU
D9014	Modo di controllo I/O	Numero modo di controllo I/O	Il modo di controllo I/O impostato viene indicato con uno dei numeri seguenti: 0: Ingressi e uscite in modo diretto 1: Ingressi in modo rinfresco, uscite in modo diretto 3: Ingressi e uscite in modo rinfresco	Non per CPU A3H, CPU A3M, CPU An
D9015	Stati operativi CPU	Stato operativo della CPU	Gli stati operativi della CPU mostrati di seguito sono memorizzati in D9015:  <p style="text-align: center;">* Se la CPU è in modo RUN e M9040 è OFF, la CPU rimane in RUN se richiesto il modo PAUSE.</p>	
D9016	Impostazioni ROM/RAM	0: ROM 1: RAM 2: EEPROM	Indica la configurazione del chip di selezione memoria. Un valore da 0 a 2 viene memorizzato come codice BIN.	Solo per CPU CPU A1, CPU A1N
	Numero programma	0: Programma principale (ROM) 1: Programma principale (RAM) 2: Sottoprogramma (RAM)	Indica quale programma di sequenza è attivo. Un valore da 0 a 2 viene memorizzato come codice BIN. "2" non viene usato con le CPU A1S, A0J2H, A2C, A2, A2N, A2N-F, A2A, A2A-F e A2U.	Non per CPU A1, CPU A1N
D9017	Tempo di scansione	Minimo tempo di scansione (per 10 ms)	Se il tempo di scansione è inferiore al contenuto di D9017, il valore viene nuovamente memorizzato con ogni istruzione END. Il valore del tempo minimo di scansione viene memorizzato come codice BIN in D9017.	
D9018	Tempo di scansione	Tempo di scansione (per 10 ms)	Il tempo di scansione viene scritto come codice BIN ad ogni istruzione END e continuamente sovrascritto.	
D9019	Tempo di scansione	Massimo tempo di scansione (per 10 ms)	Se il tempo di scansione è superiore al contenuto di D9019, il valore viene nuovamente memorizzato con ogni istruzione END. Il valore del tempo massimo di scansione viene memorizzato come codice BIN in D9019.	
D9020	Scansione costante	Tempo di scansione costante (impostato dall'utente in incrementi di 10 ms)	Specifica l'intervallo fra due scansioni del programma, in multipli di 10 ms. 0: Nessuna impostazione da 1 a 200: Attivazione. Il programma viene eseguito ad intervalli di (valore impostato) x 10 ms	Non per CPU serie A
D9021	Tempo di scansione	Tempo di scansione (unità da 1 ms)	Il tempo di scansione viene scritto e aggiornato ad ogni esecuzione dell'istruzione END.	Solo per CPU AnA, AnAS e AnU

Tabella dei registri speciali

Tabella registri speciali (serie A)

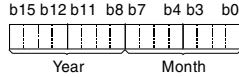
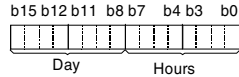
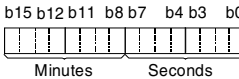
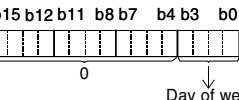
Numero	Significato	Descrizione	Dettagli	CPU
D9022	Contatore a 1 secondo	Incrementa di 1 ad ogni secondo	Dal momento che la CPU passa nello stato RUN, il contatore si incrementa ad ogni secondo. Inizia a contare da 0 a 32767, poi indietro a -32768 e successivamente ancora avanti fino a 0. Questa sequenza di conteggio viene sempre ripetuta.	Solo per CPU AnA, AnAS e AnU
② D9025	Dati orologio	(anno, mese)	Memorizza anno (2 cifre leggere) e mese in BCD: Esempio: 1992, luglio = H9207 	Solo per CPU CPU AnA CPU AnU CPU AnN CPU A1S
② D9026	Dati orologio	(giorno, ora)	Memorizza giorno e ora in BCD: Esempio: 31, 10 in punto = H3110 	Solo per CPU AnA, AnAS CPU AnU CPU AnN CPU AnS
② D9027	Dati orologio	(minuto, secondo)	Memorizza minuto e secondo in BCD: Esempio: 35 minuti, 48 secondi = H3548 	Solo per CPU AnA, AnAS CPU AnU CPU AnN CPU AnS
② D9028	Dati orologio	(Giorno della settimana)	Memorizza il giorno della settimana in BCD: (0=domenica, 1=lunedì, 2=martedì ecc.): 	Solo per CPU AnA, AnAS CPU AnU CPU AnN CPU AnS

Tabella registri speciali (serie A)

Numero	Significato	Descrizione	Dettagli	CPU																																
D9021	Terminale remoto Impostazione parametri	da 1 a 61	Imposta il numero di stazione (da 1 a 61) dei moduli terminale remoti collegati con A2C e A52G. Le impostazioni non sono obbligatoriamente ordinate secondo il numero di stazione. Configurazione dati: <div style="display: flex; justify-content: space-between;"> <div style="text-align: center;">D9021</div> <table border="1" style="border-collapse: collapse;"> <tr><td style="text-align: center;">Area terminale remoto modulo 1</td></tr> </table> </div> <div style="display: flex; justify-content: space-between;"> <div style="text-align: center;">D9022</div> <table border="1" style="border-collapse: collapse;"> <tr><td style="text-align: center;">Area terminale remoto modulo 2</td></tr> </table> </div> <div style="text-align: center;">:</div> <div style="text-align: center;">:</div> <div style="text-align: center;">:</div> <div style="text-align: center;">:</div> <div style="display: flex; justify-content: space-between;"> <div style="text-align: center;">D9032</div> <table border="1" style="border-collapse: collapse;"> <tr><td style="text-align: center;">Area terminale remoto modulo 12</td></tr> </table> </div> <div style="display: flex; justify-content: space-between;"> <div style="text-align: center;">D9033</div> <table border="1" style="border-collapse: collapse;"> <tr><td style="text-align: center;">Area terminale remoto modulo 13</td></tr> </table> </div> <div style="display: flex; justify-content: space-between;"> <div style="text-align: center;">D9034</div> <table border="1" style="border-collapse: collapse;"> <tr><td style="text-align: center;">Area terminale remoto modulo 14</td></tr> </table> </div>	Area terminale remoto modulo 1	Area terminale remoto modulo 2	Area terminale remoto modulo 12	Area terminale remoto modulo 13	Area terminale remoto modulo 14	Solo per CPU A2C																											
Area terminale remoto modulo 1																																				
Area terminale remoto modulo 2																																				
Area terminale remoto modulo 12																																				
Area terminale remoto modulo 13																																				
Area terminale remoto modulo 14																																				
D9022																																				
D9023																																				
D9024																																				
D9025																																				
D9026																																				
D9027																																				
D9028																																				
D9029																																				
D9030																																				
D9031																																				
D9032																																				
D9033																																				
D9034																																				
D9035	Attributi del modulo terminale remoto	0: Protocollo standard MINI 1: Nessun protocollo	Imposta gli attributi di ciascun modulo terminale remoto collegato con A2C e A52G con 1 o 0 su ogni bit. 0: Conforme al protocollo standard MINI o unitr terminale remoto. 1: AJ35PTF-R2 senza nessun protocollo Configurazione dati: <div style="text-align: center;"> <table style="border-collapse: collapse; margin: 0 auto;"> <tr> <td style="padding: 0 2px;">b15</td><td style="padding: 0 2px;">b14</td><td style="padding: 0 2px;">b13</td><td style="padding: 0 2px;">b12</td><td style="padding: 0 2px;">b11</td><td style="padding: 0 2px;">b10</td><td style="padding: 0 2px;">b9</td><td style="padding: 0 2px;">b8</td><td style="padding: 0 2px;">b7</td><td style="padding: 0 2px;">b6</td><td style="padding: 0 2px;">b5</td><td style="padding: 0 2px;">b4</td><td style="padding: 0 2px;">b3</td><td style="padding: 0 2px;">b2</td><td style="padding: 0 2px;">b1</td><td style="padding: 0 2px;">b0</td> </tr> <tr> <td style="border: 1px solid black; text-align: center;">0</td><td style="border: 1px solid black; text-align: center;">0</td><td style="border: 1px solid black; text-align: center;">0</td><td style="border: 1px solid black; text-align: center;">1</td><td style="border: 1px solid black; text-align: center;">0</td><td style="border: 1px solid black; text-align: center;">0</td><td style="border: 1px solid black; text-align: center;">0</td><td style="border: 1px solid black; text-align: center;">0</td><td style="border: 1px solid black; text-align: center;">0</td><td style="border: 1px solid black; text-align: center;">0</td><td style="border: 1px solid black; text-align: center;">0</td><td style="border: 1px solid black; text-align: center;">0</td><td style="border: 1px solid black; text-align: center;">0</td><td style="border: 1px solid black; text-align: center;">0</td><td style="border: 1px solid black; text-align: center;">0</td><td style="border: 1px solid black; text-align: center;">0</td> </tr> </table> <div style="margin-top: 5px; text-align: center;"> ↑ Indication of module with blown fuse </div> </div> Il bit b0 specifica gli attributi del modulo funzione speciale 1, bit b1 per il modulo 2, b2 per il modulo 3 ecc.	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																					
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0																					
D9036	Numero totale di stazioni	da 1 a 64	Imposta il numero totale di stazioni (da 1 a 64) dei moduli di I/O e moduli terminale remoto collegati con A2C e A52G.	Solo per CPU A2C																																
D9036	Indirizzo operandi file registri esteso	I numeri di operando usati per accedere direttamente a ciascun operando del file registri esteso	Indica il numero di operando per l'accesso diretto in lettura e scrittura al file registri esteso, come doppia word in D9036 e D9037, come dato BIN. Per indicare gli indirizzi di operando, usare numeri consecutivi iniziando con R0 del blocco 1.	Solo per CPU AnA, AnAS CPU AnU CPU AnU																																
D9037																																				
D9038	Priorità segnalazione LED	Priorità da 1 a 4 Priorità da 5 a 7	Imposta la priorità dei LED ERROR che si accendono (o lampeggiano) per segnalare gli errori con i numeri di codice errore. La configurazione delle aree di impostazione delle priorità è mostrata di seguito.	Solo per CPU AnA, AnAS CPU AnU CPU A2C, CPU AnS																																
D9039			<div style="text-align: center;"> <table border="1" style="border-collapse: collapse; margin: 0 auto;"> <tr> <td style="padding: 2px;">Priority 4</td> <td style="padding: 2px;">Priority 3</td> <td style="padding: 2px;">Priority 2</td> <td style="padding: 2px;">Priority 1</td> </tr> <tr> <td style="padding: 2px;"></td> <td style="padding: 2px;"></td> <td style="padding: 2px;">Priority 0</td> <td style="padding: 2px;"></td> </tr> </table> </div> Per i dettagli, fare riferimento al manuale utente della CPU appropriata e al manuale di programmazione (fondamenti) delle ACPU.		Priority 4	Priority 3	Priority 2	Priority 1			Priority 0																									
Priority 4	Priority 3	Priority 2	Priority 1																																	
		Priority 0																																		
D9055	Numero passo memorizzazione stato	Passo memorizzazione stato	Memorizza il numero di passo quando viene eseguita la memorizzazione stato.	Solo per CPU AnA, AnAS e AnU																																

Tabella dei registri speciali

Tabella registri speciali (serie A)



Numero	Significato	Descrizione	Dettagli	CPU																																																																																					
da D9056 a D9059	Segnalazione di stazione in errore	0: Normale 1: Errore	<p>Se si verifica un errore durante la comunicazione con un modulo remoto, il bit corrispondente a questo modulo viene impostato a "1" nel registro. Il bit è impostato a "1" se la comunicazione fallisce dopo il numero di tentativi specificati in D9174. Il bit rimane attivo anche se l'errore viene corretto e la stazione è stata ripristinata.</p> <p>Configurazione dati:</p> <table border="1"> <thead> <tr> <th></th> <th>b15</th> <th>b14</th> <th>b13</th> <th>b12</th> <th>b11</th> <th>b10</th> <th>b9</th> <th>b8</th> <th>b7</th> <th>b6</th> <th>b5</th> <th>b4</th> <th>b3</th> <th>b2</th> <th>b1</th> <th>b0</th> </tr> </thead> <tbody> <tr> <td>D9056</td> <td>16</td> <td>15</td> <td>14</td> <td>13</td> <td>12</td> <td>11</td> <td>10</td> <td>9</td> <td>8</td> <td>7</td> <td>6</td> <td>5</td> <td>4</td> <td>3</td> <td>2</td> <td>1</td> </tr> <tr> <td>D9057</td> <td>32</td> <td>31</td> <td>30</td> <td>29</td> <td>28</td> <td>27</td> <td>26</td> <td>25</td> <td>24</td> <td>23</td> <td>22</td> <td>21</td> <td>20</td> <td>19</td> <td>18</td> <td>17</td> </tr> <tr> <td>D9058</td> <td>48</td> <td>47</td> <td>46</td> <td>45</td> <td>44</td> <td>43</td> <td>42</td> <td>41</td> <td>40</td> <td>39</td> <td>38</td> <td>37</td> <td>36</td> <td>35</td> <td>34</td> <td>33</td> </tr> <tr> <td>D9059</td> <td>64</td> <td>63</td> <td>62</td> <td>61</td> <td>60</td> <td>59</td> <td>58</td> <td>57</td> <td>56</td> <td>55</td> <td>54</td> <td>53</td> <td>52</td> <td>51</td> <td>50</td> <td>49</td> </tr> </tbody> </table> <p>Numeri di stazione</p> <p>Il bit corrispondente è impostato a "1" anche se si interrompe il fusibile di un modulo di I/O remoto.</p>		b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	D9056	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	D9057	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	D9058	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	D9059	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	Solo per CPU A2C
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																									
D9056	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1																																																																									
D9057	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17																																																																									
D9058	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33																																																																									
D9059	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49																																																																									
D9061	Codice di errore di comunicazione	0: Normale 1: Errore inizializzazione 2: Errore di linea	<p>Memorizza il codice di errore quando M99061 viene attivato (comunicazione con moduli di I/O remoto o terminali remoti fallita).</p> <p>1: Il numero totale di stazioni per moduli I/O remoti o terminali remoti o il numero di tentativi è anomalo. Il programma di inizializzazione contiene un errore.</p> <p>2: Interruzione del cavo o mancanza alimentazione sui moduli I/O remoti o terminali remoti.</p>	Solo per CPU A2C																																																																																					
D9072	Controllo comunicazioni PC	Dati controllati da AJ71C24(S3/S6/S8)	Nella modalità di test loopback dei singoli moduli AJ71C24(S3/S6/S8), AJ71C24(S3/S6/S8) esegue lettura/scrittura dati e controllo della comunicazione.	Solo per CPU AnA, AnAS e AnU																																																																																					
D9073	Dati orologio	Anno e mese	Le funzioni corrispondono a quelle del registro speciale D9035.	Solo per CPU A2C C24(-PRF)																																																																																					
D9074	Dati orologio	Giorno e ora	Le funzioni corrispondono a quelle del registro speciale D9036.																																																																																						
D9075	Dati orologio	Minuto e secondo	Le funzioni corrispondono a quelle del registro speciale D9037.																																																																																						
D9076	Dati orologio	Giorno della settimana	Le funzioni corrispondono a quelle del registro speciale D9038.																																																																																						
D9081	Numero delle richieste di comunicazione per moduli terminale remoti	da 0 a 32	Memorizza il numero delle richieste di comunicazione eseguite verso i moduli terminale remoti collegati con AJ71PT32(S3), A2C e A52G. Sottrae 1 al completamento della comunicazione con un modulo terminale remoto.	Solo per CPU AnA, AnAS CPU AnU A2C																																																																																					
D9082	Ultimo numero stazione collegata	Ultimo numero stazione collegata	Memorizza il numero dell'ultima stazione relativa a moduli di I/O remoti o terminali remoti, collegati con A2C e A52G.	Solo per CPU A2C																																																																																					
D9090	Indirizzo iniziale operandi area dati d'ingresso sottoprogramma microcomputer	Dipende dal pacchetto di programma microcomputer usato.	Per i dettagli, consultare il manuale dei singoli pacchetti per programmi microcomputer.	Non per AnA, AnAS e AnU																																																																																					
D9091	Errore istruzione	Numero dettagli errore istruzione	Memorizza il codice dettagliato della causa di un errore istruzione.	Solo per CPU AnA, AnAS e AnU																																																																																					
D9091	Codice errore di chiamata subroutine microcomputer	Dipende dal pacchetto di programma microcomputer usato.	Per i dettagli, consultare il manuale dei singoli pacchetti per programmi microcomputer.	Non per AnA, AnAS e AnU																																																																																					
 D9094	Indirizzo iniziale modulo I/O in sostituzione	Indirizzo iniziale modulo I/O in sostituzione	<p>Memorizza in codice BIN le 2 cifre più pesanti dell'indirizzo iniziale di I/O del modulo da sostituire in modalità online.</p> <p>Esempio: Modulo ingressi X2F0 = H2F</p>	Solo per CPU CPU AnN AnA, AnAS e AnU																																																																																					
 D9100	Fusibile interrotto sul modulo	Maschera di bit dei moduli con fusibile interrotto	<p>Memorizza il numero del modulo di uscita con fusibili interrotti come maschera di bit.</p> <table border="1"> <thead> <tr> <th></th> <th>b15</th> <th>b14</th> <th>b13</th> <th>b12</th> <th>b11</th> <th>b10</th> <th>b9</th> <th>b8</th> <th>b7</th> <th>b6</th> <th>b5</th> <th>b4</th> <th>b3</th> <th>b2</th> <th>b1</th> <th>b0</th> </tr> </thead> <tbody> <tr> <td>D9100</td> <td>0</td> <td>0</td> <td>0</td> <td>1</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> </tbody> </table> <p style="text-align: center;">↑ Indication of module with blown fuse</p>		b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	D9100	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	Solo per CPU CPU A1S																																																			
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																									
D9100	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0																																																																									

Tabella registri speciali (serie A)

Numero	Significato	Descrizione	Dettagli	CPU																																																																				
1 D9100 1 D9101 1 D9102 1 D9103 1 D9104 1 D9105 1 D9106 1 D9107	Fusibile del modulo interrotto	Maschera di bit in gruppi da 16 punti di moduli con fusibili interrotti	Numero dei moduli di uscita (in gruppi da 16 punti) che presentano fusibili interrotti, rappresentati con maschera di bit. La maschera di bit corrisponde ai dati impostati con la parametrizzazione. <table style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td></td> <td style="text-align: center;">b15</td><td style="text-align: center;">b14</td><td style="text-align: center;">b13</td><td style="text-align: center;">b12</td><td style="text-align: center;">b11</td><td style="text-align: center;">b10</td><td style="text-align: center;">b9</td><td style="text-align: center;">b8</td><td style="text-align: center;">b7</td><td style="text-align: center;">b6</td><td style="text-align: center;">b5</td><td style="text-align: center;">b4</td><td style="text-align: center;">b3</td><td style="text-align: center;">b2</td><td style="text-align: center;">b1</td><td style="text-align: center;">b0</td> </tr> <tr> <td style="text-align: left;">D9100</td> <td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td> </tr> <tr> <td style="text-align: left;">D9101</td> <td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td> </tr> <tr> <td style="text-align: left;">D9107</td> <td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td> </tr> </table> Il controllo dei fusibili interrotti viene eseguito anche sui moduli di uscita delle stazioni di I/O remoto. Se viene ripristinato lo stato normale, i bit non vengono cancellati. La cancellazione è a carico del programma utente).		b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	D9100	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	D9101	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	D9107	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	Non per CPU A1S, solo con CPU AnS da D9100 a D9103
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																								
D9100	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0																																																								
D9101	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0																																																								
D9107	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0																																																								
1 D9116	Moduli di I/O con errore di verifica	Maschera dei numeri di modulo	Se lo stato attuale di un modulo di I/O è diverso dallo stato rilevato all'accensione, il bit corrispondente al modulo di I/O nella maschera contenuta in D9116 viene attivato (assegnazione definita dai parametri). Dopo il ripristino dello stato normale, il bit deve essere azzerato dal programma utente.	Solo per CPU CPU A1S																																																																				
D9116 D9117 D9118 D9119 D9120 D9121 D9122 D9123	Errore verifica modulo di I/O	Maschera di bit in gruppi da 16 punti delle unità con errori di verifica	Se vengono rilevati moduli di I/O i cui dati risultano diversi da quelli rilevati all'accensione, i bit corrispondenti all'indirizzo di I/O (in gruppi da 16 punti) vengono attivati nella maschera. La maschera di bit corrisponde ai dati impostati con la parametrizzazione. <table style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td></td> <td style="text-align: center;">b15</td><td style="text-align: center;">b14</td><td style="text-align: center;">b13</td><td style="text-align: center;">b12</td><td style="text-align: center;">b11</td><td style="text-align: center;">b10</td><td style="text-align: center;">b9</td><td style="text-align: center;">b8</td><td style="text-align: center;">b7</td><td style="text-align: center;">b6</td><td style="text-align: center;">b5</td><td style="text-align: center;">b4</td><td style="text-align: center;">b3</td><td style="text-align: center;">b2</td><td style="text-align: center;">b1</td><td style="text-align: center;">b0</td> </tr> <tr> <td style="text-align: left;">D9116</td> <td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td> </tr> <tr> <td style="text-align: left;">D9117</td> <td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td> </tr> <tr> <td style="text-align: left;">D9123</td> <td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">1</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td><td style="border: 1px solid black; padding: 2px;">0</td> </tr> </table> <div style="margin-left: 100px;"> ↑ Indication of I/O module with verification error </div> La verifica dei moduli di I/O viene effettuata anche sui moduli delle stazioni di I/O remoto. Se viene ripristinato lo stato normale, i bit non vengono cancellati. La cancellazione è a carico del programma utente).		b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	D9116	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	D9117	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	D9123	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	Non per CPU A1S, solo con CPU AnS da D9116 a D9119
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																								
D9116	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0																																																								
D9117	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0																																																								
D9123	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0																																																								
D9124	Numero spie di errore attive	Numero spie di errore attive	Se una delle spie da F0 a F255 viene attivata tramite una istruzione OUT F o SET F, il contenuto di D9124 viene incrementato di 1. Se viene eseguita una istruzione RST F o LEDR, il contenuto di D9124 viene decrementato di 1. Se la CPU possiede l'interruttore INDICATOR RESET, la stessa funzione si ottiene azionando l'interruttore. Il numero di flag attivati da OUT F o SET F viene memorizzato in D9124 come codice BIN. Il valore massimo di D9124 è 8.																																																																					

Tabella registri speciali (serie A)

Numero	Significato	Descrizione	Dettagli	CPU																																																																	
D9141 ? D9172	Numero di tentativi	Numero tentativi	<p>Memorizza il numero di tentativi eseguiti con i moduli di I/O remoto o terminali remoti che provocano gli errori di comunicazione. (Il processo di retry viene eseguito per il numero di volte impostato in D9174).</p> <p>Il dato diventa 0 quando la comunicazione viene ripristinata.</p> <p>L'impostazione del numero di stazione dei moduli di I/O remoto e terminali remoti è mostrata di seguito.</p> <table style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="padding: 2px;"></td> <td style="text-align: center; padding: 2px;">b15</td> <td style="text-align: center; padding: 2px;">-</td> <td style="text-align: center; padding: 2px;">b8</td> <td style="text-align: center; padding: 2px;">b7</td> <td style="text-align: center; padding: 2px;">-</td> <td style="text-align: center; padding: 2px;">b0</td> </tr> <tr> <td style="padding: 2px;">D9141</td> <td colspan="2" style="text-align: center; border: 1px solid black;">Station 2</td> <td colspan="4" style="text-align: center; border: 1px solid black;">Station 1</td> </tr> <tr> <td style="padding: 2px;">D9142</td> <td colspan="2" style="text-align: center; border: 1px solid black;">Station 4</td> <td colspan="4" style="text-align: center; border: 1px solid black;">Station 3</td> </tr> <tr> <td style="padding: 2px;">D9143</td> <td colspan="2" style="text-align: center; border: 1px solid black;">Station 6</td> <td colspan="4" style="text-align: center; border: 1px solid black;">Station 5</td> </tr> <tr> <td style="padding: 2px;">↓</td> <td colspan="2"></td> <td colspan="4"></td> </tr> <tr> <td style="padding: 2px;">D9171</td> <td colspan="2" style="text-align: center; border: 1px solid black;">Station 62</td> <td colspan="4" style="text-align: center; border: 1px solid black;">Station 61</td> </tr> <tr> <td style="padding: 2px;">D9172</td> <td colspan="2" style="text-align: center; border: 1px solid black;">Station 64</td> <td colspan="4" style="text-align: center; border: 1px solid black;">Station 63</td> </tr> </table> <p style="margin-left: 40px;">Il contatore di tentativi usa 8 bit per ciascuna stazione.</p> <table style="margin-left: auto; margin-right: auto; border-collapse: collapse;"> <tr> <td style="text-align: center; padding: 2px;">b(n+7)</td> <td style="text-align: center; padding: 2px;">b(n+6)</td> <td style="text-align: center; padding: 2px;">b(n+5)</td> <td style="text-align: center; padding: 2px;">b(n+4)</td> <td style="text-align: center; padding: 2px;">b(n+3)</td> <td style="text-align: center; padding: 2px;">b(n+2)</td> <td style="text-align: center; padding: 2px;">b(n+1)</td> <td style="text-align: center; padding: 2px;">b(n+0)</td> </tr> <tr> <td style="text-align: center; border: 1px solid black;">0/1</td> <td style="text-align: center; border: 1px solid black;"></td> <td style="text-align: center; border: 1px solid black;"></td> <td style="text-align: center; border: 1px solid black;"></td> <td style="text-align: center; border: 1px solid black;"></td> <td style="text-align: center; border: 1px solid black;"></td> <td style="text-align: center; border: 1px solid black;"></td> <td style="text-align: center; border: 1px solid black;"></td> </tr> </table> <p style="text-align: center; margin-left: 40px;">Number of retries</p> <p style="margin-left: 40px;">0: Normale 1: Stazione in errore</p> <p>"n" è determinato dal numero di stazione del modulo di I/O o del terminale remoto.</p> <p>Stazioni con numero dispari: da b0 a b7 (n = 0) Stazioni con numero pari: da b8 a b15 (n = 8)</p>		b15	-	b8	b7	-	b0	D9141	Station 2		Station 1				D9142	Station 4		Station 3				D9143	Station 6		Station 5				↓							D9171	Station 62		Station 61				D9172	Station 64		Station 63				b(n+7)	b(n+6)	b(n+5)	b(n+4)	b(n+3)	b(n+2)	b(n+1)	b(n+0)	0/1								Solo per CPU A2C
	b15	-	b8	b7	-	b0																																																															
D9141	Station 2		Station 1																																																																		
D9142	Station 4		Station 3																																																																		
D9143	Station 6		Station 5																																																																		
↓																																																																					
D9171	Station 62		Station 61																																																																		
D9172	Station 64		Station 63																																																																		
b(n+7)	b(n+6)	b(n+5)	b(n+4)	b(n+3)	b(n+2)	b(n+1)	b(n+0)																																																														
0/1																																																																					
D9173	Impostazione modo	0: Ripristino automatico online abilitato 1: Ripristino automatico online disabilitato 2: Arresto trasmissione su errore online 3: Controllo linea	<p>Impostazione modo</p> <p>Modo 0 (ripristino automatico online abilitato): Se un modulo di I/O o un modulo terminale remoto provoca un errore di comunicazione, la stazione viene posta offline. La comunicazione con le altre stazioni prosegue normalmente. Se la stazione in avaria viene ripristinata, viene posta online.</p> <p>Modo 1 (Ripristino automatico online disabilitato): Se un modulo di I/O o un modulo terminale remoto provoca un errore di comunicazione, la stazione viene posta offline. La comunicazione con le altre stazioni prosegue normalmente. Anche se la stazione in avaria viene ripristinata, la comunicazione non viene riavviata a meno che il modulo della stazione non venga riavviato.</p> <p>Modo 2 (Arresto trasmissione con errore online): Se un modulo di I/O o un modulo terminale remoto provoca un errore di comunicazione, la comunicazione con tutte le stazioni viene interrotta. Anche se una stazione in avaria viene ripristinata, la comunicazione non viene riavviata a meno che il modulo della stazione non venga riavviato.</p> <p>Modo 3 (controllo linea) Controlla l'hardware e i cavi di collegamento dei moduli di I/O e dei moduli terminale remoti.</p>	Solo per CPU A2C																																																																	
D9174	Impostazione del numero di tentativi	Numero tentativi	<p>Imposta il numero di tentativi eseguiti con i moduli di I/O remoto o terminali remoti che provocano gli errori di comunicazione. Impostato su 5 all'accensione.</p> <p>Campo di impostazione: da 0 a 32</p> <p>Se la comunicazione con un modulo di I/O remoto o un modulo terminale remoto non viene ripristinata dopo il numero di tentativi specificato, il modulo viene considerato come stazione in avaria.</p>	Solo per CPU A2C																																																																	
D9175	Contatore tentativi errore di linea	Numero tentativi	<p>Memorizza il numero di tentativi eseguiti per un errore di linea (timeout).</p> <p>Il dato diventa 0 quando la linea viene ripristinata e la comunicazione con i moduli di I/O remoti e con i moduli terminale remoti riavviata.</p>	Solo per CPU A2C																																																																	

Tabella dei registri speciali

Tabella registri speciali (serie A)

Numero	Significato	Descrizione	Dettagli	CPU																																																																																					
D9180 ? D9193	Numero di errore modulo terminale remoto	Codice di errore (0: Normale)	<p>Memorizza il codice di errore di un modulo terminale remoto in avaria, se M9060 è attivo.</p> <p>Le aree di memorizzazione del codice di errore per ciascun modulo terminale remoto sono mostrate di seguito.</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>D9180</td> <td>Remote special module No.1</td> </tr> <tr> <td>D9181</td> <td>Remote special module No.2</td> </tr> <tr> <td>D9182</td> <td>Remote special module No.3</td> </tr> <tr> <td style="text-align: center;">↓</td> <td style="text-align: center;">:</td> </tr> <tr> <td>D9192</td> <td>Remote special module No.13</td> </tr> <tr> <td>D9193</td> <td>Remote special module No.14</td> </tr> </table> <p>Il codice di errore viene cancellato nei casi seguenti. Commutatore di RUN portato da STOP a RUN. I registri da D9180 a D9193 sono tutti azzerati. Quando l'uscita Yn4 di ciascun modulo terminale remoto passa da OFF a ON.</p>	D9180	Remote special module No.1	D9181	Remote special module No.2	D9182	Remote special module No.3	↓	:	D9192	Remote special module No.13	D9193	Remote special module No.14	Solo per CPU A2C																																																																									
D9180	Remote special module No.1																																																																																								
D9181	Remote special module No.2																																																																																								
D9182	Remote special module No.3																																																																																								
↓	:																																																																																								
D9192	Remote special module No.13																																																																																								
D9193	Remote special module No.14																																																																																								
D9196 ? D9199	Rilevamento stazione in avaria	Maschera di bit per le stazioni in avaria	<p>Il bit corrispondente al modulo I/O o modulo terminale remoto in avaria è impostato a 1.</p> <p>Il bit corrispondente ad una stazione in avaria viene attivato quando non è possibile ripristinare la normale comunicazione dopo aver eseguito il numero di tentativi impostato in D9174.</p> <p>Se è abilitato il ritorno automatico online, il bit corrispondente alla stazione in avaria viene azzerato quando la stazione viene ripristinata.</p> <p>Configurazione dati</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th></th> <th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th> </tr> </thead> <tbody> <tr> <td>D9196</td> <td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td> </tr> <tr> <td>D9197</td> <td>32</td><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td> </tr> <tr> <td>D9198</td> <td>48</td><td>47</td><td>46</td><td>45</td><td>44</td><td>43</td><td>42</td><td>41</td><td>40</td><td>39</td><td>38</td><td>37</td><td>36</td><td>35</td><td>34</td><td>33</td> </tr> <tr> <td>D9199</td> <td>64</td><td>63</td><td>62</td><td>61</td><td>60</td><td>59</td><td>58</td><td>57</td><td>56</td><td>55</td><td>54</td><td>53</td><td>52</td><td>51</td><td>50</td><td>49</td> </tr> </tbody> </table>		b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	D9196	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	D9197	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	D9198	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	D9199	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	Solo per CPU A2C
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																									
D9196	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1																																																																									
D9197	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17																																																																									
D9198	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33																																																																									
D9199	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49																																																																									

NOTA

Tutti i registri speciali vengono azzerati dopo aver tolto l'alimentazione, dopo una cancellazione memoria o RESET.

Se il commutatore RUN viene portato su STOP, il contenuto dei registri viene mantenuto.

Il contenuto dei registri speciali contrassegnati con ❶ viene conservato, se lo stato normale è ripristinato. Possono essere disattivati come segue:

- Inserire una riga nel programma di sequenza per azzerare i registri speciali con una istruzione RST condizionata opportunamente.
- Forzare un RESET con il terminale di programmazione.
- Azzerare la CPU portando l'interruttore della CPU su RESET.

I registri speciali contrassegnati ❷ possono essere modificati solo con istruzioni del programma.

I registri speciali contrassegnati ❸ vengono modificati tramite terminale di programmazione in modo test..

A.5.3 Tabella registri di comunicazione (solo serie A)

I registri di comunicazione vengono sovrascritti durante la comunicazione dati in una rete, a seconda di diverse condizioni. I registri contengono lo stato della comunicazione e gli errori della rete come valori numerici. Controllando un registro di comunicazione è possibile leggere da qualsiasi numero di stazione diagnosticata come in avaria.

L'elaborazione dei registri di comunicazione è diversa a seconda che la CPU sia inserita in una stazione master o locale.

Registri di comunicazione nella stazione master

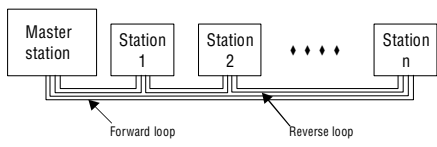
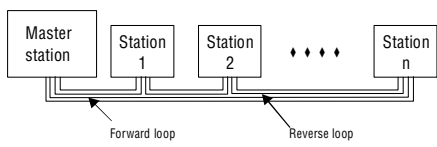
Numero	Significato	Descrizione	Dettagli
D9200	Risultato esecuzione LRDP	0: Normale 2: Errore impostazione istruzione LRDP 3: Stazione in errore 4: LRDP non può essere eseguita nella stazione corrispondente	Memorizza il risultato dell'esecuzione dell'istruzione LRDP (lettura operandi a word). Errore impostazione istruzione LRDP: Impostazione errata dell'istruzione LRDP: costanti, sorgente e/o destinazione. Stazione in errore corrispondente: una delle stazioni non sta comunicando. LRDP non può essere eseguita nella stazione corrispondente: La stazione specificata è una stazione di I/O remoto.
D9201	Risultato esecuzione LWTP	0: Normale 2: Errore impostazione istruzione LWTP 3: Stazione in errore 4: LWTP non può essere eseguita nella stazione corrispondente	Memorizza il risultato dell'esecuzione dell'istruzione LWTP (scrittura operandi a word). Errore impostazione istruzione LWTP: Impostazione errata dell'istruzione LWTP: costanti, sorgente e/o destinazione. Stazione in errore corrispondente: una delle stazioni non sta comunicando. LWTP non può essere eseguita nella stazione corrispondente: La stazione specificata è una stazione di I/O remoto.
D9202	Tipo di collegamento di una stazione locale (vedi anche D9241, D9242)	Stato delle stazioni da 1 a 16	I registri dati memorizzano la compatibilità delle stazioni slave con le reti MELSECNET o MELSECNET/II. Se una stazione slave è compatibile con MELSECNET/II il bit corrispondente del registro speciale contiene "1". Se è compatibile con MELSECNET, il contenuto è "0".
D9203		Stato delle stazioni da 17 a 32	
D9204 (Continua)	Stato comunicazione	0: Collegamento nel loop avanti 1: Collegamento nel loop indietro 2: Loopback in direzione avanti o indietro 3: Loopback in direzione avanti 4: Loopback in direzione indietro 5: Collegamento impossibile	Memorizza lo stato del percorso attuale del collegamento dati. Collegamento nel loop avanti  Collegamento nel loop indietro 

Tabella dei registri speciali

Registri di comunicazione nella stazione master

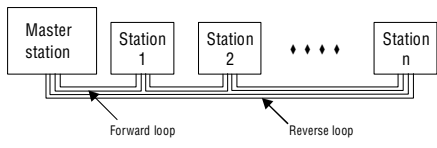
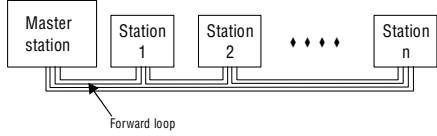
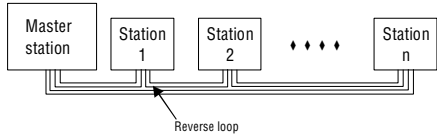
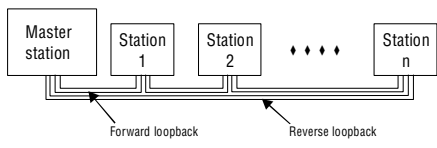
Numero	Significato	Descrizione	Dettagli
D9204	Stato comunicazione		<p>Loopback nei loop avanti/indietro</p>  <p>Loopback nel solo loop avanti</p>  <p>Loopback nel solo loop indietro</p> 
D9205*	Stazione che esegue il loop	Stazione che esegue il loopback avanti	Memorizza il numero di stazione locale o di I/O remoto su cui viene eseguito il loopback.
D9206*	Stazione che esegue il loop	Stazione che esegue il loopback indietro	 <p>Nell'esempio precedente, 1 viene scritto in D9205 e 3 in D9206. Se il collegamento torna nel suo stato normale (collegamento nel loop avanti) i valori in D9205 e D9206 rimangono 1 e 3. L'azzeramento avviene con il programma utente o il tasto RESET.</p>
D9207*	Tempo di scansione del collegamento	Valore massimo	Memorizza il tempo di esecuzione del collegamento con tutte le stazioni locali e di I/O remoto.
D9208*	Tempo di scansione del collegamento	Valore minimo	In ogni ciclo di comunicazione vengono scambiati ingressi (X), uscite (Y), relé di comunicazione (B) e registri di comunicazione (W) con le stazioni corrispondenti.
D9209*	Tempo di scansione del collegamento	Valore attuale	Un ciclo di comunicazione è il periodo di tempo in cui viene eseguito il collegamento con tutte le stazioni slave collegate, indipendentemente dal tempo di scansione del programma di sequenza.
D9210*	Conteggio tentativi	Numero totale memorizzato	Memorizza il numero di tentativi dovuti a errori di trasmissione. Il conteggio si arresta al valore massimo di "FFFF _H ". Eseguire RESET per riportare a 0 il conteggio.
D9211*	Conteggio delle commutazioni di loop	Numero totale memorizzato	Memorizza il numero di volte che la linea è stata commutata fra loop avanti e loop indietro o su loopback.

Tabella dei registri speciali

Registri di comunicazione nella stazione master

Numero	Significato	Descrizione	Dettagli																																																																																					
D9212	Stato operativo stazione locale	Memorizza lo stato delle stazioni da 1 a 16	Memorizza i numeri delle stazioni locali che sono in STOP o PAUSE. <table border="1" style="font-size: small; margin: 10px 0;"> <thead> <tr> <th></th> <th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th> </tr> </thead> <tbody> <tr> <td>D9212</td> <td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td> </tr> <tr> <td>D9213</td> <td>32</td><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td> </tr> <tr> <td>D9214</td> <td>48</td><td>47</td><td>46</td><td>45</td><td>44</td><td>43</td><td>42</td><td>41</td><td>40</td><td>39</td><td>38</td><td>37</td><td>36</td><td>35</td><td>34</td><td>33</td> </tr> <tr> <td>D9215</td> <td>64</td><td>63</td><td>62</td><td>61</td><td>60</td><td>59</td><td>58</td><td>57</td><td>56</td><td>55</td><td>54</td><td>53</td><td>52</td><td>51</td><td>50</td><td>49</td> </tr> </tbody> </table> Numeri di stazione Se una stazione locale viene portata in modo STOP o PAUSE, il bit corrispondente al numero di stazione nel registro diventa "1". Esempio: se la stazione 7 commuta in modo STOP, il bit 6 in D9212 diventa "1", e se D9212 viene controllato, il suo valore è "64 (49h)".		b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	D9212	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	D9213	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	D9214	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	D9215	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49
		b15		b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																						
D9212		16		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1																																																																						
D9213		32		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17																																																																						
D9214	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33																																																																								
D9215	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49																																																																								
D9213	Memorizza lo stato delle stazioni da 17 a 32																																																																																							
D9214*	Memorizza lo stato delle stazioni da 33 a 48																																																																																							
D9215*	Memorizza lo stato delle stazioni da 49 a 64																																																																																							
D9216	Rilevamento errori stazione locale	Memorizza lo stato delle stazioni da 1 a 16	Memorizza i numeri delle stazioni locali che sono in errore. <table border="1" style="font-size: small; margin: 10px 0;"> <thead> <tr> <th></th> <th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th> </tr> </thead> <tbody> <tr> <td>D9216</td> <td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td> </tr> <tr> <td>D9217</td> <td>32</td><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td> </tr> <tr> <td>D9218</td> <td>48</td><td>47</td><td>46</td><td>45</td><td>44</td><td>43</td><td>42</td><td>41</td><td>40</td><td>39</td><td>38</td><td>37</td><td>36</td><td>35</td><td>34</td><td>33</td> </tr> <tr> <td>D9219</td> <td>64</td><td>63</td><td>62</td><td>61</td><td>60</td><td>59</td><td>58</td><td>57</td><td>56</td><td>55</td><td>54</td><td>53</td><td>52</td><td>51</td><td>50</td><td>49</td> </tr> </tbody> </table> Numeri di stazione Se una stazione locale rileva un errore, il bit corrispondente al numero di stazione diventa "1". Esempio: se le stazioni 6 e 12 rilevano un errore, i bit 5 e 11 in D9216 diventano "1", e se D9216 viene controllato, il suo valore è "2080 (820h)".		b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	D9216	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	D9217	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	D9218	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	D9219	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49
		b15		b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																						
D9216		16		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1																																																																						
D9217		32		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17																																																																						
D9218	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33																																																																								
D9219	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49																																																																								
D9217	Memorizza lo stato delle stazioni da 17 a 32																																																																																							
D9218*	Memorizza lo stato delle stazioni da 33 a 48																																																																																							
D9219*	Memorizza lo stato delle stazioni da 49 a 64																																																																																							
D9220	Errore di parametrizzazione di una stazione locale o errore di assegnazione di una stazione I/O remota	Memorizza lo stato delle stazioni da 1 a 16	Memorizza i numeri delle stazioni locali che contengono parametri errati o di stazioni remote per cui è stata fatta una assegnazione errata degli indirizzi di I/O. <table border="1" style="font-size: small; margin: 10px 0;"> <thead> <tr> <th></th> <th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th> </tr> </thead> <tbody> <tr> <td>D9220</td> <td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td> </tr> <tr> <td>D9221</td> <td>32</td><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td> </tr> <tr> <td>D9222</td> <td>48</td><td>47</td><td>46</td><td>45</td><td>44</td><td>43</td><td>42</td><td>41</td><td>40</td><td>39</td><td>38</td><td>37</td><td>36</td><td>35</td><td>34</td><td>33</td> </tr> <tr> <td>D9223</td> <td>64</td><td>63</td><td>62</td><td>61</td><td>60</td><td>59</td><td>58</td><td>57</td><td>56</td><td>55</td><td>54</td><td>53</td><td>52</td><td>51</td><td>50</td><td>49</td> </tr> </tbody> </table> Numeri di stazione Se una stazione locale che funziona come stazione master di livello 3 rileva un errore di parametrizzazione, oppure se una stazione remota contiene una assegnazione di indirizzi di I/O non valida, il bit corrispondente al numero di stazione diventa "1". Esempio: se la stazione locale 5 e la stazione di I/O remoto 14 rilevano un errore, i bit 4 e 13 in D9220 diventano "1", e se viene controllato D9220, il valore visualizzato è "8208 (2010h)".		b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	D9220	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	D9221	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	D9222	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	D9223	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49
		b15		b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																						
D9220		16		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1																																																																						
D9221		32		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17																																																																						
D9222	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33																																																																								
D9223	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49																																																																								
D9221	Memorizza lo stato delle stazioni da 17 a 32																																																																																							
D9222*	Memorizza lo stato delle stazioni da 33 a 48																																																																																							
D9223*	Memorizza lo stato delle stazioni da 49 a 64																																																																																							
D9224	Comunicazione iniziale fra stazioni locali o di I/O remote	Memorizza lo stato delle stazioni da 1 a 16	Memorizza i numeri delle stazioni locali o remote durante la comunicazione dei dati di inizializzazione con le rispettive stazioni master. <table border="1" style="font-size: small; margin: 10px 0;"> <thead> <tr> <th></th> <th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th> </tr> </thead> <tbody> <tr> <td>D9224</td> <td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td> </tr> <tr> <td>D9225</td> <td>32</td><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td> </tr> <tr> <td>D9226</td> <td>48</td><td>47</td><td>46</td><td>45</td><td>44</td><td>43</td><td>42</td><td>41</td><td>40</td><td>39</td><td>38</td><td>37</td><td>36</td><td>35</td><td>34</td><td>33</td> </tr> <tr> <td>D9227</td> <td>64</td><td>63</td><td>62</td><td>61</td><td>60</td><td>59</td><td>58</td><td>57</td><td>56</td><td>55</td><td>54</td><td>53</td><td>52</td><td>51</td><td>50</td><td>49</td> </tr> </tbody> </table> Numeri di stazione Il bit corrispondente con il numero della stazione che sta attualmente comunicando i dati di inizializzazione diventa "1". Esempio: Se le stazioni 23 e 45 stanno comunicando, il bit 6 di D9225 e il bit 12 di D9226 diventano "1", e se viene controllato D9225, il suo valore viene visualizzato come "64 (40h)", e se viene controllato D9226, il suo valore è "4096 (1000h)".		b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	D9224	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	D9225	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	D9226	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	D9227	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49
		b15		b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																						
D9224		16		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1																																																																						
D9225		32		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17																																																																						
D9226	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33																																																																								
D9227	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49																																																																								
D9225	Memorizza lo stato delle stazioni da 17 a 32																																																																																							
D9226*	Memorizza lo stato delle stazioni da 33 a 48																																																																																							
D9227*	Memorizza lo stato delle stazioni da 49 a 64																																																																																							

Tabella dei registri speciali

Registri di comunicazione nella stazione master

Numero	Significato	Descrizione	Dettagli																																																																																																																																																																																																																																																																																
D9228	Errore stazione locale o di I/O remoto	Memorizza lo stato delle stazioni da 1 a 16	<p>Memorizza i numeri delle stazioni locali o remote che sono in errore.</p> <table border="1"> <thead> <tr> <th></th> <th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th> </tr> </thead> <tbody> <tr> <td>D9228</td> <td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td> </tr> <tr> <td>D9229</td> <td>32</td><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td> </tr> <tr> <td>D9230</td> <td>48</td><td>47</td><td>46</td><td>45</td><td>44</td><td>43</td><td>42</td><td>41</td><td>40</td><td>39</td><td>38</td><td>37</td><td>36</td><td>35</td><td>34</td><td>33</td> </tr> <tr> <td>D9231</td> <td>64</td><td>63</td><td>62</td><td>61</td><td>60</td><td>59</td><td>58</td><td>57</td><td>56</td><td>55</td><td>54</td><td>53</td><td>52</td><td>51</td><td>50</td><td>49</td> </tr> </tbody> </table> <p>Il bit corrispondente al numero di stazione in errore diventa "1".</p> <p>Esempio: se la stazione locale 3 e la stazione di I/O remoto 14 rilevano un errore, i bit 2 e 13 in D9228 diventano "1", e se viene controllato D9228, il valore visualizzato è "8196 (2004h)".</p>		b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	D9228	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	D9229	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	D9230	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	D9231	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49																																																																																																																																																																																											
		b15		b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																																																																																																																																																																																																																	
D9228		16		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1																																																																																																																																																																																																																																																																	
D9229		32		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17																																																																																																																																																																																																																																																																	
D9230	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33																																																																																																																																																																																																																																																																			
D9231	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49																																																																																																																																																																																																																																																																			
D9229	Memorizza lo stato delle stazioni da 17 a 32																																																																																																																																																																																																																																																																																		
D9230*	Memorizza lo stato delle stazioni da 33 a 48																																																																																																																																																																																																																																																																																		
D9231*	Memorizza lo stato delle stazioni da 49 a 64																																																																																																																																																																																																																																																																																		
D9232	Errore loop stazione locale o di I/O remoto	Memorizza lo stato delle stazioni da 1 a 8	<p>Memorizza i numeri delle stazioni locali o remote in cui si è verificato un errore nel loop avanti o indietro.</p> <table border="1"> <thead> <tr> <th></th> <th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th> </tr> </thead> <tbody> <tr> <td>D9232</td> <td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td> </tr> <tr> <td></td> <td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> <tr> <td>D9233</td> <td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td> </tr> <tr> <td>D9234</td> <td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td> </tr> <tr> <td></td> <td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> <tr> <td>D9235</td> <td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td> </tr> <tr> <td></td> <td>32</td><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> <tr> <td>D9136</td> <td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td> </tr> <tr> <td></td> <td>40</td><td>39</td><td>38</td><td>37</td><td>36</td><td>35</td><td>34</td><td>33</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> <tr> <td>D9237</td> <td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td> </tr> <tr> <td></td> <td>48</td><td>47</td><td>46</td><td>45</td><td>44</td><td>43</td><td>42</td><td>41</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> <tr> <td>D9238</td> <td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td> </tr> <tr> <td></td> <td>56</td><td>55</td><td>54</td><td>53</td><td>52</td><td>51</td><td>50</td><td>49</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> <tr> <td>D9239</td> <td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td><td>R</td><td>V</td> </tr> <tr> <td></td> <td>64</td><td>63</td><td>62</td><td>61</td><td>60</td><td>59</td><td>58</td><td>57</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td> </tr> </tbody> </table> <p>Nella tabella precedente, "F" indica il loop avanti, mentre "R" indica il loop indietro. Il bit corrispondente al numero di stazione in cui si è verificato l'errore sul loop avanti o indietro, diventa "1".</p> <p>Esempio: Se la linea del loop avanti della stazione 5 presenta un errore, il bit 8 di D9232 diventa "1", e se viene visualizzato D9232, il suo valore è "256 (100h)".</p>		b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	D9232	R	V	R	V	R	V	R	V	R	V	R	V	R	V	R	V		8	7	6	5	4	3	2	1									D9233	R	V	R	V	R	V	R	V	R	V	R	V	R	V	R	V	D9234	R	V	R	V	R	V	R	V	R	V	R	V	R	V	R	V		24	23	22	21	20	19	18	17									D9235	R	V	R	V	R	V	R	V	R	V	R	V	R	V	R	V		32	31	30	29	28	27	26	25									D9136	R	V	R	V	R	V	R	V	R	V	R	V	R	V	R	V		40	39	38	37	36	35	34	33									D9237	R	V	R	V	R	V	R	V	R	V	R	V	R	V	R	V		48	47	46	45	44	43	42	41									D9238	R	V	R	V	R	V	R	V	R	V	R	V	R	V	R	V		56	55	54	53	52	51	50	49									D9239	R	V	R	V	R	V	R	V	R	V	R	V	R	V	R	V		64	63	62	61	60	59	58	57								
		b15		b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																																																																																																																																																																																																																	
D9232		R		V	R	V	R	V	R	V	R	V	R	V	R	V	R	V																																																																																																																																																																																																																																																																	
		8		7	6	5	4	3	2	1																																																																																																																																																																																																																																																																									
D9233		R		V	R	V	R	V	R	V	R	V	R	V	R	V	R	V																																																																																																																																																																																																																																																																	
D9234		R		V	R	V	R	V	R	V	R	V	R	V	R	V	R	V																																																																																																																																																																																																																																																																	
		24		23	22	21	20	19	18	17																																																																																																																																																																																																																																																																									
D9235		R		V	R	V	R	V	R	V	R	V	R	V	R	V	R	V																																																																																																																																																																																																																																																																	
		32		31	30	29	28	27	26	25																																																																																																																																																																																																																																																																									
D9136	R	V	R	V	R	V	R	V	R	V	R	V	R	V	R	V																																																																																																																																																																																																																																																																			
	40	39	38	37	36	35	34	33																																																																																																																																																																																																																																																																											
D9237	R	V	R	V	R	V	R	V	R	V	R	V	R	V	R	V																																																																																																																																																																																																																																																																			
	48	47	46	45	44	43	42	41																																																																																																																																																																																																																																																																											
D9238	R	V	R	V	R	V	R	V	R	V	R	V	R	V	R	V																																																																																																																																																																																																																																																																			
	56	55	54	53	52	51	50	49																																																																																																																																																																																																																																																																											
D9239	R	V	R	V	R	V	R	V	R	V	R	V	R	V	R	V																																																																																																																																																																																																																																																																			
	64	63	62	61	60	59	58	57																																																																																																																																																																																																																																																																											
D9233	Memorizza lo stato delle stazioni da 9 a 16																																																																																																																																																																																																																																																																																		
D9234	Memorizza lo stato delle stazioni da 17 a 24																																																																																																																																																																																																																																																																																		
D9235	Memorizza lo stato delle stazioni da 25 a 32																																																																																																																																																																																																																																																																																		
D9236	Memorizza lo stato delle stazioni da 33 a 40																																																																																																																																																																																																																																																																																		
D9237	Memorizza lo stato delle stazioni da 41 a 48																																																																																																																																																																																																																																																																																		
D9238	Memorizza lo stato delle stazioni da 49 a 56																																																																																																																																																																																																																																																																																		
D9239	Memorizza lo stato delle stazioni da 57 a 64																																																																																																																																																																																																																																																																																		
D9240	Contatore errori di trasmissione	Numero totale memorizzato	<p>Memorizza il numero di volte che si sono verificati i seguenti errori di trasmissione: CRC, OVER, AB.IF</p> <p>Il conteggio arriva fino al massimo di FFFF. Eseguire RESET per riportare a 0 il conteggio.</p>																																																																																																																																																																																																																																																																																
D9241*	Tipo di collegamento di una stazione locale (vedi anche D9202, D9203)	Memorizza lo stato delle stazioni da 33 a 48	<p>I registri dati memorizzano la compatibilità delle stazioni slave con le reti MELSECNET o MELSECNET/II.</p> <p>Se una stazione slave è compatibile con MELSECNET/II il bit corrispondente del registro speciale contiene "1". Se è compatibile con MELSECNET, il contenuto è "0".</p>																																																																																																																																																																																																																																																																																
D9242*		Memorizza lo stato delle stazioni da 49 a 64																																																																																																																																																																																																																																																																																	
D9243	Controllo del numero della propria stazione	Memorizza un numero di stazione (da 0 a 64)	Consente a una stazione locale di confermare il proprio numero di stazione.																																																																																																																																																																																																																																																																																
D9244	Numero totale di stazioni slave	Memorizza il numero di stazioni Slave	Indica il numero di stazioni slave in un loop.																																																																																																																																																																																																																																																																																
D9245	Contatore errori di trasmissione	Numero totale memorizzato	<p>Memorizza il numero di volte che si sono verificati i seguenti errori di trasmissione: CRC, OVER, AB.IF</p> <p>Il conteggio arriva fino al massimo di FFFFh. Eseguire RESET per riportare a 0 il conteggio.</p>																																																																																																																																																																																																																																																																																

Registri di comunicazione nella stazione master

Numero	Significato	Descrizione	Dettagli																																																																																					
D9248	Stato operativo stazione locale	Memorizza lo stato delle stazioni da 1 a 16	Memorizza i numeri delle stazioni locali che sono in STOP o PAUSE. <table border="1" style="font-size: small; margin: 10px 0;"> <thead> <tr> <th></th> <th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th> </tr> </thead> <tbody> <tr> <td>D9196</td> <td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td> </tr> <tr> <td>D9197</td> <td>32</td><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td> </tr> <tr> <td>D9198</td> <td>48</td><td>47</td><td>46</td><td>45</td><td>44</td><td>43</td><td>42</td><td>41</td><td>40</td><td>39</td><td>38</td><td>37</td><td>36</td><td>35</td><td>34</td><td>33</td> </tr> <tr> <td>D9199</td> <td>64</td><td>63</td><td>62</td><td>61</td><td>60</td><td>59</td><td>58</td><td>57</td><td>56</td><td>55</td><td>54</td><td>53</td><td>52</td><td>51</td><td>50</td><td>49</td> </tr> </tbody> </table>		b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	D9196	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	D9197	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	D9198	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	D9199	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49
		b15		b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																						
D9196		16		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1																																																																						
D9197		32		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17																																																																						
D9198	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33																																																																								
D9199	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49																																																																								
D9249	Memorizza lo stato delle stazioni da 17 a 32																																																																																							
D9250*	Memorizza lo stato delle stazioni da 33 a 48																																																																																							
D9251*	Memorizza lo stato delle stazioni da 49 a 64																																																																																							
D9252	Errore stazione locale	Memorizza lo stato delle stazioni da 1 a 16	Memorizza il numero di stazione locale, diversa dalla stazione host, che è in errore. <table border="1" style="font-size: small; margin: 10px 0;"> <thead> <tr> <th></th> <th>b15</th><th>b14</th><th>b13</th><th>b12</th><th>b11</th><th>b10</th><th>b9</th><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>b0</th> </tr> </thead> <tbody> <tr> <td>D9196</td> <td>16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td> </tr> <tr> <td>D9197</td> <td>32</td><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td> </tr> <tr> <td>D9198</td> <td>48</td><td>47</td><td>46</td><td>45</td><td>44</td><td>43</td><td>42</td><td>41</td><td>40</td><td>39</td><td>38</td><td>37</td><td>36</td><td>35</td><td>34</td><td>33</td> </tr> <tr> <td>D9199</td> <td>64</td><td>63</td><td>62</td><td>61</td><td>60</td><td>59</td><td>58</td><td>57</td><td>56</td><td>55</td><td>54</td><td>53</td><td>52</td><td>51</td><td>50</td><td>49</td> </tr> </tbody> </table>		b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	D9196	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	D9197	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	D9198	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	D9199	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49
		b15		b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0																																																																						
D9196		16		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1																																																																						
D9197		32		31	30	29	28	27	26	25	24	23	22	21	20	19	18	17																																																																						
D9198	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33																																																																								
D9199	64	63	62	61	60	59	58	57	56	55	54	53	52	51	50	49																																																																								
D9253	Memorizza lo stato delle stazioni da 17 a 32																																																																																							
D9254*	Memorizza lo stato delle stazioni da 33 a 48																																																																																							
D9255*	Memorizza lo stato delle stazioni da 49 a 64																																																																																							

* I registri speciali contrassegnati non possono essere utilizzati con MELSECNET/B.

A

ACOS, ACOSP. 7-356
 Aggregazione dati a 16-bit. 7-99
 Altre istruzioni utili, parte 1. 6-178
 Altre istruzioni utili, parte 2. 7-466
 Altre istruzioni (panoramica) 2-59
 ANB. 5-11
 AND. 5-4
 logico. 7-6
 ANDF. 5-8
 ANDP 5-8
 ANI 5-4
 ANY (tipo dati) 3-9
 Area di memorizzazione
 Programma microcomputer 12-1
 Area retentiva
 Registri speciali A-102
 relé speciali A-57
 Array (indirizzamento). 3-20
 ASC (serie A) 7-316
 ASC, ASCP (serie Q) 7-313
 ASIN, ASINP 7-353
 ATAN, ATANP 7-359
 Attivazione e cancellazione
 di bit singoli 7-67
 memoria di stato. 7-240
 Attivazione e disattivazione
 segnale di riporto 7-470
 traccia campionamento 7-242
 Attivazione, disattivazione ed esecuzione
 traccia di programma 7-244

B

BACOS, BACOSP 7-396
 BAND, BANDP, DBAND, DBANDP 7-406
 BASIN, BASINP 7-393
 BATAN, BATANP 7-399
 BBLKRD, BBLKRDP. 11-27
 BBLKWR, BBLKWRP 11-30
 BCD 4-cifre
 operazioni di moltiplicazione e divisione 6-54
 operazioni di somma e sottrazione. 6-44
 BCD 8-cifre
 operazioni di moltiplicazione e divisione 6-58
 operazioni di somma e sottrazione. 6-49
 BCDDA, BCDDAP, DBCDDA, DBCDDAP 7-258
 BCD, BCDP, DBCD, DBCDP. 6-82
 BCOS, BCOSP. 7-386
 BIN 16-bit
 moltiplicazione e divisione 6-37
 operazioni di incremento e decremento 6-78
 operazioni di somma e sottrazione. 6-29
 BIN 32-bit
 moltiplicazione e divisione 6-41
 operazioni di incremento e decremento 6-81

operazioni di somma e sottrazione 6-34
 BINDA, BINDAP, DBINDA, DBINDAP 7-248
 BINHA, BINHAP, DBINHA, DBINHAP 7-253
 BIN, BINP, DBIN, DBINP 6-85
 BKAND, BKANDP 7-11
 BKBCD, BKBCDP 6-110
 BKBIN, BKBINP. 6-113
 BKCMP, BKCMP_P 6-20
 BKOR, BKORP 7-20
 BKRST, BKRSTP. 7-73
 BKXNR, BKXNRP 7-39
 BKXOR, BKXORP 7-29
 BK+, BK+P, BK-, BK-P 6-69
 Blocchi BIN
 confronto dati. 6-21
 operazioni di somma e sottrazione 6-70
 scambio dati 6-141
 trasferimento dati 6-132
 Blocchi di bit 3-12
 di dati doppia word 3-14
 di word dati 3-13
 Blocchi funzionali per contatori A-39
 BMOV, BMOVP 6-131
 BREAK, BREAKP 7-129
 BSET, BSETP, BRST, BRSTP 7-66
 BSFR, BSFRP, BSFL, BSFLP 7-59
 BSIN, BSINP 7-383
 BSQR, BSQRP, BDSQR, BDSQRP 7-379
 BTAN, BTANP 7-389
 BUFRCV 11-34
 BUFRCVS (moduli comunicazione seriale) 11-3
 BUFRCVS (modulo interfaccia ETHERNET) 11-39
 BUFSND 11-42
 BXCH, BXCHP 6-140
 Bx, BxP, B/, B/P. 6-53
 B+, B+P, B-, B-P 6-43

C

Calcolo		Serie QnA e System Q (tranne CPU Q00J, Q00, Q01)	13-12
del totale di blocchi dati BIN a 16 bit.	7-122	Codifica da 256 a 8 bit	7-90
del totale di blocchi dati BIN a 32 bit.	7-124	Collegamento in parallelo di blocco a contatti	5-12
Calcolo arcocoseno		Collegamento in serie di blocco di contatti	5-12
da dato BCD.	7-397	Collegamento parallelo	5-5
da valori in virgola mobile	7-357	COM	6-172
Calcolo arcoseno		Comandi di confronto IEC	6-3
da dato BCD.	7-394	Commutazione fra programma	
da valori in virgola mobile	7-354	MAIN e SUB	7-148
Calcolo arcotangente		COMRD, COMRDP	7-277
da dato BCD.	7-400	Condizioni di esecuzione	2-6
da valori in virgola mobile	7-360	per istruzioni di confronto	6-3
Calcolo del coseno		per un rinfresco di collegamento	6-176
da dato BCD.	7-387	Condizioni di esecuzione delle istruzioni	3-34
da valori in virgola mobile	7-348	Configurazione area memoria dati	12-4
Calcolo del logaritmo (ln) da valori in virgola mobile	7-375	Configurazione indirizzi dell'area memorizzazione dati	12-3
Calcolo del seno		Confronto	
da dato BCD.	7-384	Contatori	A-38
da valori in virgola mobile	7-345	CPU QnA e CPU System Q	A-41
Calcolo della tangente		di moduli CPU	A-32
da dato BCD.	7-390	Istruzioni di visualizzazione	A-39
da valori in virgola mobile	7-351	Temporizzatore	A-35
Calcolo radice quadrata		Confronto dati	
da valori in virgola mobile	7-369	BIN a 16 bit	6-6
di dati BCD a 4 o 8 cifre.	7-380	BIN a 32 bit	6-9
CALL, CALLP	7-132	in virgola mobile	6-12
Campi di lettura e scrittura dei dati	8-3	operazioni di moltiplicazione e divisione	6-67
Cancellazione		operazioni di somma e sottrazione	6-62
spie e segnalazioni di errore	7-214	Confronto dati a	
timer di watchdog	7-468	stringhe di caratteri	6-16
Cancellazione di un programma		Connessione (ETHERNET)	
dalla memoria.	9-37	apertura	11-50
Cancellazione e caricamento di		chiusura	11-58
un programma	9-39	contatore avanti/indietro a fase doppia	6-183
Cancellazione e inserimento di blocchi		contatore avanti/indietro a fase singola	6-180
dati specificati in una tabella	7-181	Controllo anomalie per operazioni bidirezionali	
Capacità di memorizzazione ed aree		Serie QnA e System Q	7-219
di memoria	12-1	Solo serie A	7-227
Categorie di istruzioni	2-1	Controllo bit dati	7-85
CGMODE	10-4	Conversione	
CHG	7-147	da blocchi dati BCD a blocchi dati BIN	6-114
Chiamata a programma microcomputer	7-157	da blocchi dati BIN a blocchi dati BCD	6-111
CHK (solo serie A)	7-226	da dati BCD a dati BIN	6-86
CHKCIR, CHKEND	7-234	da dati BIN a dati BCD	6-83
CHKST, CHK (Solo serie Q e System Q):	7-218	da dati BIN a dati in codice Gray	6-100
CJ, SCJ, JMP	6-148	da dati BIN a dati in virgola mobile	6-90
CLOSE	11-56	da dati BIN a 16 bit a dati BIN a 32 bit	6-96
CML, CMLP, DCML, DCMLP	6-126	da dati BIN a 32 bit a dati BIN a 16 bit	6-98
Codice di errore (modulo ETHERNET)		da dati in codice Gray a dati BIN	6-103
cancellazione	11-63	da dati in virgola mobile a dati BIN	6-93
lettura	11-69	da gradi a radianti con valori in virgola mobile	7-363
Codici di errore		da radianti a gradi con valori in virgola mobile	7-366
CPU AnA e AnAS	13-43	Conversione da indirizzo array a indirizzo iniziale	3-21
Q00JCPU, Q00CPU, Q01CPU	13-2		
Serie A (tranne AnA e AnAS)	13-39		

Conversione da serie A a System Q/serie QnA
 Registri speciali A-105
 relé speciali A-58
 Conversione di
 dati BCD a 4/8 cifre in codice ASCII 7-259
 dati binari a 16-/32-bit in stringa
 di caratteri 7-285
 dati binari a 16-/32-bit in valori
 decimali in codice ASCII 7-249
 dati binari a 16-/32-bit in valori
 esadecimali in codice ASCII 7-254
 dati decimali ASCII a dati BIN a 16/32 bit 7-264
 dati esadecimali ASCII a dati
 BIN a 16/32 bit 7-269
 stringa di carattere in codice ASCII 7-317
 stringhe di caratteri in dati binari
 a 16-/32-bit. 7-293
 un dato ASCII decimale in codice
 BCD a 4/8 cifre 7-273
 un dato BIN a 16 bit in codice ASCII ... 7-314
 un dato in virgola mobile in stringa
 di caratteri 7-299
 un numero in virgola mobile in
 formato BCD 7-337
 un numero in virgola mobile in
 formato decimale 7-340
 una stringa di caratteri in un dato
 in virgola mobile 7-308
 valori ASCII esadecimali in valori binari . 7-319
 Conversione risultato
 operazione in impulso 5-20
 Costanti (uso in una istruzione) 3-1
 COS, COSP 7-347
 CPU AnAS/AnUS 1-2
 CPU AnA/AnU 1-2
 CPU AnN 1-2
 CPU AnS 1-2
 CPU QnA 1-2
 CPU System Q 1-2

D

DABCD, DABCDP, DDABCD, DDABCDP .. 7-272
 DABIN, DABINP, DDABIN, DDABINP 7-263
 DATERD, DATERDP 7-425
 DATEWR, DATEWRP 7-429
 DATE-, DATE-P 7-438
 DATE+, DATE+P 7-433
 Dati BIN
 inversione 6-127
 istruzioni per operazioni aritmetiche 6-27
 scambio 6-138
 trasferimento 6-118
 Dati BIN 16/32-bit
 controllo zona morta 7-407
 somma offset 7-411
 Dati di instradamento (panoramica) 2-62
 Dati in virgola mobile
 trasferimento 6-121
 Dati orologio
 lettura 7-426
 modifica formato 7-445
 scrittura 7-430
 somma 7-434
 sottrazione 7-439
 Dato a 7 segmenti 7-93
 DBL, DBLP 6-95
 DBx, DBxP, DB/, DB/P 6-57
 DB+, DB+P, DB-, DB-P 6-48
 Debug
 Registri speciali A-101
 Relé speciali A-56
 Decodifica da 8 a 256 bit 7-88
 Decodifica 7 segmenti 7-92
 DECO, DECOP 7-87
 DEG, DEGP 7-365
 Destinazione dei dati (d) 3-2
 Diagnostica guasti e debug 7-217
 DINC, DINCP, DDEC, DDECP 6-80
 DINT (tipo dati) 3-19
 Disaggregazione dati a 16-bit 7-96
 Disaggregazione e aggregazione dati in byte 7-107
 Disaggregazione o aggregazione
 di dati in raggruppamenti di bit qualsiasi ... 7-102
 Disattivazione uscite
 in un sottoprogramma 7-138
 in un sottoprogramma di un file 7-145
 DIS, DISP 7-95
 DI, EI, IMASK 6-156
 DROL, DROLP, DRCL, DRCLP 7-52
 DROR, DRORP, DRCR, DRCRP 7-49
 DSFR, DSFRP, DSFL, DSFLP 7-62
 DUTY 7-471
 DWORD 3-19
 DWSUM, DWSUMP 7-123
 Dx, DxP, D/, D/P 6-40
 D+, D+P, D-, D-P 6-32
 D=, D, D>, D= 6-8

E		G	
ECALL, ECALLP.	7-141	GBIN, GBINP, DGBIN, DGBINP	6-102
EFCALL, EFCALLP.	7-144	Generazione circuiti di controllo	
EI, DI	6-175	per istruzione CHK	7-235
Elaborazione		Generazione e aggiornamento	
dati a bit	3-11	serie valori random	7-378
di dati a doppia word (32 bit)	3-14	Gerarchia dei tipi dati ANY	3-9
di dati a word (16 bit)	3-12	GETE, GETEP	11-6
Elaborazione risultato		GOEND	6-153
operazione	5-15	GRY, GRYP, DGRY, DGRYP	6-99
EMOD, EMODP	7-336		
EMOV, EMOVP	6-120	H	
ENCO, ENCO P	7-89	HABIN, HABINP, DHABIN, DHABINP	7-268
ENEG, ENEGP	6-108	HEX, HEXP	7-318
EREXP, EREXPP	7-339		
EROMWR, EROMWRP	7-492	I	
ERRCLR	11-61	Implementazione matrice di ingressi	6-204
Errori di esecuzione		Impostazione	
durante l'esecuzione di un'istruzione	3-31	blocchi file registri	7-416
nella spiegazione dell'istruzione	4-6	file commenti	7-422
ERRRD	11-67	nome file registri	7-419
Esempi di calcolo dei passi		Impostazione programma	
di programma	3-39	in modo scansione	7-463
Estrazione di parti di stringhe di caratteri	7-323	in modo standby	7-459
ESTR, ESTRP	7-298	in modo standby con azzeramento	
ETHERNET		delle uscite	7-461
lettura codice di errore	11-69	modo esecuzione a bassa velocità	7-465
Reinizializzazione	11-74	Impulso	
Spegnimento LED „ERR.“	11-63	collegato in parallelo	5-9
EVAL, EVALP	7-307	collegato in serie	5-9
EXP, EXPP	7-371	INC, INCP, DEC, DECP	6-77
Ex, ExP, E/, E/P	6-66	Indicizzazione	
E+, E+P, E-, E-P	6-61	CPU AnA, AnAS e AnU	3-28
E=, E, E>, E=	6-11	CPU System Q e QnA	3-26
		Generalità	3-24
		Indicizzazione di intere sezioni	
		di programma	7-160
		Indirizzamento di	
		array	3-20
		registri	3-19
		Indirizzamento indiretto	
		memorizzazione	7-482
		Indirizzi operandi indicizzati	
		in una tabella di indicizzazione	7-165
		Informazioni di sistema,	
		Registri speciali	A-83
		relé speciali	A-46
		Informazioni diagnostiche	
		Registri speciali	A-75
		Informazioni modulo, lettura	9-2
		Informazioni scansione	
		Registri speciali	A-94
		relé speciali	A-51
F			
FCALL, FCALLP	7-137		
FDEL, FDELP, FINS, FINSP	7-180		
FIFR, FIFRP	7-172		
FIFW, FIFWP	7-168		
Fine sottoprogramma	7-136		
FLT, FLTP, DFLT, DFLTP	6-89		
FMOV, FMOVP	6-134		
FOR, NEXT	7-126		
FPOP, FPOPP	7-176		
FREAD	9-20		
FROM, DFRO	7-186		
FROM/FROMP	9-50		
Funzioni	4-6		
Funzioni contatore	A-38		
Funzioni speciali	7-342		
Funzioni timer	A-35		
FWRITE	9-9		

Ingresso EN	3-35	Istruzioni di elaborazione stringhe	
Ingresso tastiera da dispositivo periferico	7-455	di caratteri (panoramica)	2-48
Inizio operazione	5-5	Istruzioni di ingresso	5-4
Inserimento tasti numerici	7-484	Istruzioni di ingresso (panoramica)	2-7
INSTR, INSTRP	7-332	Istruzioni di programma	
Intestazione AWL MELSEC	3-7	panoramica	2-58
INT, INTP, DINT, DINTP	6-92	per System Q	9-33
INV	5-17	Istruzioni di rotazione (panoramica)	2-36
Inversione di segno		Istruzioni di salto	6-149
per dati BIN (complemento a 2)	6-106	Istruzioni di salto di programma	6-147
per dati in virgola mobile	6-109	Istruzioni di salto programma (panoramica)	2-28
Inversione risultato		Istruzioni di scorrimento	7-55
operazione	5-18	Istruzioni di scorrimento (panoramica)	2-9, 2-37
Invio dati ad altre stazioni	8-42	Istruzioni di sequenza	5-1
Invio messaggi a dispositivi periferici	7-452	Istruzioni di terminazione programma	
Invio su un display a LED		(panoramica)	2-10
LED	7-206	Istruzioni di trasferimento dati (panoramica)	2-26
LEDA, LEDB	7-212	Istruzioni di trasferimento per System Q	9-41
LEDC	7-209	Istruzioni di uscita (panoramica)	2-9
IRET	6-163	Istruzioni di visualizzazione	7-194
Istruzione CHG		Istruzioni di visualizzazione (panoramica)	2-46
e conteggio dei contatori	7-152	Istruzioni miscelanee (panoramica)	2-10
e temporizzazioni dei timer	7-153	Istruzioni per comunicazione dati	8-1
ed elaborazione delle istruzioni OUT	7-153	Istruzioni per comunicazione dati	
insieme ad una istruzione impulsiva (xP)	7-151	compatibili con serie A	8-76
insieme all'istruzione PLS	7-150	Istruzioni per controllo dati	7-401
Istruzione di posizionamento		Istruzioni per controllo esecuzione	
per tavole rotanti	6-192	del programma	6-155
Istruzione (struttura)	3-1	Istruzioni per controllo interruzione	6-157
Istruzioni applicative, parte 1	6-1	Istruzioni per controllo interruzione	
Istruzioni applicative, parte 2	7-1	(panoramica)	2-28
Istruzioni controllo di programma	7-457	Istruzioni per debug (panoramica)	2-47
Istruzioni dedicate	3-6	Istruzioni per dispositivi periferici	7-450
Istruzioni dedicate per collegamenti dati		Istruzioni per dispositivi periferici	
Generalità	8-4	(panoramica)	2-58
Istruzioni dedicate per collegamenti dati		Istruzioni per elaborazione dati	7-76
(panoramica)	2-61	Istruzioni per funzioni speciali (panoramica)	2-52
Istruzioni dedicate per comunicazione dati		Istruzioni per moduli funzione speciali	11-1
per la serie QnA	8-11	Istruzioni per NOR esclusivo	7-40
Istruzioni di		Istruzioni per operazioni aritmetiche	6-25
debug	7-217	Istruzioni per operazioni aritmetiche	
debug per CPU System Q	9-7	su dati BCD	6-27
Istruzioni di accesso a buffer di memoria	7-185	Istruzioni per operazioni aritmetiche	
Istruzioni di accesso a buffer di memoria		(panoramica)	2-16
(panoramica)	2-45	Istruzioni per operazioni di confronto	6-2
Istruzioni di collegamento	5-11	Istruzioni per operazioni di confronto	
Istruzioni di collegamento (panoramica)	2-8	(panoramica)	2-11
Istruzioni di commutazione file registri	7-414	Istruzioni per operazioni logiche	7-2
Istruzioni di commutazione file registro		Istruzioni per operazioni logiche	
(panoramica)	2-56	(panoramica)	2-32
Istruzioni di controllo dati (panoramica)	2-55	Istruzioni per operazioni su tabelle dati	7-167
Istruzioni di controllo primario (panoramica)	2-10	Istruzioni per OR esclusivo	7-30
Istruzioni di conversione dati	6-81	Istruzioni per orologio	7-424
Istruzioni di conversione dati (panoramica)	2-23	Istruzioni per orologio (panoramica)	2-57
Istruzioni di elaborazione bit	7-65	Istruzioni per programmi strutturati	7-125
Istruzioni di elaborazione bit (panoramica)	2-38	Istruzioni per programmi	
Istruzioni di elaborazione dati (panoramica)	2-39	strutturati (panoramica)	2-42
Istruzioni di elaborazione stringhe di caratteri	7-245	Istruzioni per Q4ARCPU	10-1

Istruzioni per rinfresco collegamento	6-165
Istruzioni per rinfresco dati.	8-6
Istruzioni per rinfresco dati (panoramica)	2-29
Istruzioni per rinfresco rete (panoramica)	2-61
Istruzioni per rotazione dati	7-42
Istruzioni per rotazione dati (panoramica)	2-36
Istruzioni per tabelle dati (panoramica)	2-44
Istruzioni per trasferimento dati	6-116
Istruzioni per uso in un sistema multi CPU	9-46
IXDEV, IXSET	7-164
IX, IXEND	7-159

K

KEY.	7-483
--------------	-------

L

LD	5-4
LDF	5-8
LDI	5-4
LDP	5-8
LED	7-205
LEDA, LEDB	7-211
LEDC.	7-208
LEDR.	7-213
Legge primo dato	
inserito da una tabella dati.	7-173
LEN, LENP.	7-281
Letture	
dati di instradamento.	8-107
Letture commento operando	7-278
Letture dati	
da altre stazioni	8-78
da buffer fisso.	11-36
da moduli funzione speciali in	
stazioni I/O remote	8-65
da un modulo di comunicazione.	11-4
da un modulo funzione speciale.	7-187
da un modulo interfaccia PROFIBUS	11-28
da una diversa stazione di CC-Link	11-112
da una stazione intelligente (CC-Link)	11-140, 11-146
da una stazione locale	8-86
da una stazione remota.	8-95
da un'altro PLC (CC-Link)	11-119
dal buffer ad aggiornamento	
automatico (serie A)	11-170
dal buffer ad aggiornamento	
automatico (serie Q)	11-174
in un programma di interruzione.	11-40
Letture dati operandi a word da una	
diversa stazione (READ).	8-16
Letture dati operandi a word da una	
diversa stazione (SREAD).	8-22
Letture dati orologio	7-426

Letture di	
trame utente registrate	11-8
Letture diretta di un byte da un file registri	7-475
Letture e scrittura di informazioni di	
instradamento	8-105
Letture ultimo dato	
inserito da una tabella dati	7-177
Limitazione valori di uscita	7-403
LIMIT, LIMITP, DLIMIT, DLIMITP	7-402
LOG, LOGP.	7-374
LRDP	8-85
LWTP	8-89

M

Mappa di memoria	12-3
MAX, MAXP, DMAX, DMAXP	7-111
MEF	5-19
MELSECNET	
Impostazione accoppiamento di stazioni	11-80
Memoria condivisa CPU	
lettura	9-50
scrittura	9-47
Memorizzazione e spostamento di	
parti di stringhe di caratteri	7-327
Memory card	
Registri speciali	A-96
relé speciali	A-52
MEP	5-19
MIDR, MIDRP, MIDW, MIDWP	7-326
Misura densità di impulso	6-198
Modalità Microcomputer (AnN(S))	12-1
Modifica formato orologio	7-445
Modulazione larghezza impulso	6-202
Moduli con fusibile interrotto	
(registri speciali)	A-103
MOV, MOVP, DMOV, DMOVP	6-117
MPP	5-14
MPS	5-14
MRD	5-14
MSG	7-451
MTR	6-203
M_REAL_TO_REAL	3-18
M_REAL_TO_REAL_E	3-18

N

NDIS, NDISP, NUNI, NUNIP.	7-101
NEG, NEGp, DNEG, DNEGp	6-105
Nome istruzione.	2-5
Notazione delle istruzioni	3-3
Numero passi di programma	
Per una CPU AnA, AnAS e AnU	3-38
Numero (n)	3-2

O

OPEN 11-47

Operandi

 MELSEC A. 4-3

 MELSEC Q 4-4

Operandi a bit

 di dati a word 3-12

 di dati doppia word 3-14

 uso nelle istruzioni 3-11

Operandi a word

 disposizione dei bit 3-11

 uso in una istruzione 3-13

Operazione

 NOR esclusivo 7-33

OR. 5-4

 logico. 7-15

ORB 5-11

Ordina dati a 16-/32-bit 7-118

ORF. 5-8

ORI 5-4

Orologio di sistema/contatori

 Registri speciali A-93

 relé speciali A-50

ORP 5-8

P

PAIRSET 11-79

Panoramica

 Altre istruzioni. 2-59

 Blocchi funzionali timer A-37

 Codici di errore. 7-232

 Dati di instradamento 2-62

 Istruzioni dedicate per collegamenti dati. 2-61

 Istruzioni di accesso a buffer di memoria 2-45

 Istruzioni di collegamento 2-8

 Istruzioni di commutazione file registri 2-56

 Istruzioni di controllo primario 2-10

 Istruzioni di conversione dati 2-23

 Istruzioni di elaborazione bit 2-38

 Istruzioni di elaborazione stringhe

 di caratteri 2-48

 Istruzioni di ingresso 2-7

 Istruzioni di programma. 2-58

 Istruzioni di rotazione 2-36

 Istruzioni di salto di programma 2-28

 Istruzioni di scorrimento 2-9, 2-37

 Istruzioni di terminazione programma. 2-10

 Istruzioni di uscita. 2-9

 Istruzioni di visualizzazione 2-46

 Istruzioni miscelanee 2-10

 Istruzioni per controllo dati. 2-55

 Istruzioni per controllo interruzione. 2-28

 Istruzioni per CPU System Q. 9-1

 Istruzioni per debug 2-47

 Istruzioni per dispositivi periferici 2-58

Istruzioni per elaborazione dati 2-39

Istruzioni per funzioni speciali 2-52

Istruzioni per operazioni aritmetiche 2-16

Istruzioni per operazioni di confronto 2-11

Istruzioni per operazioni logiche 2-32

Istruzioni per orologio 2-57

Istruzioni per programmi strutturati 2-42

Istruzioni per rinfresco dati 2-29

Istruzioni per rinfresco rete 2-61

Istruzioni per tabelle dati. 2-44

Istruzioni per trasferimento dati 2-26

Istruzioni serie Q e System Q

 equivalenti alle istruzioni per serie A A-40

 Modi di controllo I/O A-34

Operandi per dati di controllo (READ) 8-14

Operandi per dati di controllo (RECV) 8-48

Operandi per dati di controllo (REQ) 8-54

Operandi per dati di controllo (SEND) 8-40

Operandi per dati di controllo (SREAD) 8-20

Operandi per dati di controllo (SWRITE) 8-34

Operandi per dati di controllo (WRITE) 8-27

Operandi per dati di controllo (ZNFR) 8-64

Operandi per dati di controllo (ZNT0) 8-70

Relé speciali A-43

Tabelle istruzioni 2-1

Tempi di elaborazione (definizione) A-1

Tipi di dati A-34

Ulteriori manuali. 1-1

Parametri PLC 1-3

PKEY 7-454

PLOADP 9-33

PLOW, PLOWP 7-464

PLSY 6-199

POFF, POFFP 7-460

PPC-CPU-CPU686(MS)-128 1-2

PPC-CPU686(MS)-64 1-2

PR 7-196

PRC 7-201

Preset del numero di scansioni di

 attivazione di un operando 7-472

 Prodotto logico con blocchi dati da 16 bit 7-12

Programma

 cancellazione dalla memoria 9-37

 Cancellazione e caricamento 9-39

 caricamento da una memory card 9-34

Programma microcomputer

 Applicazione 12-2

Programmazione

 di istruzioni dedicate. 3-6

 di variabili 3-7

PRR, PRRP 11-18

PSCAN, PSCANP 7-462

PSTOP, PSTOPP 7-458

PSWAPP. 9-38

PTRA, PTRAR, PTRAEXE, PTRAEXEP 7-243

PUNLOADP. 9-36

PUTE, PUTEF. 11-11

PWM. 6-201

Q

QCDSET, QCDSETP	7-421
QDRSET, QDRSETP	7-418
Q00CPU	1-2
Q00JCPU	1-2
Q01CPU	1-2
Q02CPU	1-2
Q02HCPU	1-2
Q06HCPU	1-2
Q12HCPU	1-2
Q4ARCPU	
Trasferimento dati	10-6
Q4ARCPU, impostazione modo	10-2

R

RAD, RADP	7-362
RAMP	6-195
RBMOV, RBMOV P	9-41
READ	8-12
REAL (tipo dati)	3-17
REAL_TO_M_REAL	3-18
REAL_TO_M_REAL_E	3-18
RECV	8-47
Registri iniziali	3-19
Registri relativi alle istruzioni	A-99
REQ	8-52
Reset di gruppi	
di bit	7-74
RET	7-135
RFRP	8-93
RFS, RFSP	6-166
Ricerca	
valore massimo in dati a 16/32 bit	7-112
valore minimo in dati a 16/32 bit	7-115
Ricerca dati	7-79
Ricerca di stringhe di caratteri	7-333
Ricerca di una istruzione	1-3
Ricezione dati inviati da altre stazioni	8-49
Richiamo	
di un sottoprogramma	7-133
di un sottoprogramma in un file	7-142
Richiesta dati	
da altre stazioni	8-59
durante operazione RUN/STOP su	
stazione remota	8-58
Richiesta/risposta dati durante	
operazioni di scrittura/lettura di dati orologio	8-55
RIFR (serie A)	11-170
RIFR (serie QnA, System Q)	11-174
RIGHT, RIGHTP, LEFT, LEFTP	7-322
Rilevamento della lunghezza di stringhe	
di caratteri	7-282
Rinfresco collegamento	6-173
Rinfresco dati di rete	8-8
Rinfresco parziale I/O	

serie A	6-169
serie Q e System Q	6-167
RIRCV (serie A)	11-138
RIRCV (serie QnA, System Q)	11-144
RIRD (serie A)	11-109
RIRD (serie QnA, System Q)	11-115
RISEND (serie A)	11-150
RISEND (serie QnA, System Q)	11-156
RITO (serie A)	11-162
RITO (serie QnA, System Q)	11-166
Ritorno da un programma di interruzione	6-164
RIWT (serie A)	11-123
RIWT (serie QnA, System Q)	11-130
RLPA (serie A)	11-83
RLPASET, RLPASET_P	11-89
RND, RNDP, SRND, SRNDP	7-377
ROL, ROLP, RCL, RCLP	7-46
ROR, RORP, RCR, RCRP	7-43
Rotazione dati	
verso destra (16 bit)	7-44
verso destra (32 bit)	7-50
verso sinistra (16 bit)	7-47
verso sinistra (32 bit)	7-53
ROTC	6-191
RRPA (serie A)	11-101
RTOP	8-99
RTREAD	8-106
RTWRITE	8-108

S

Salti di programma	6-149
Salto a fine programma	6-154
Salvataggio e ripristino	
di gruppi di registri indice	7-490
Scambio byte	6-144
Scambio fra byte leggero e byte pesante	6-144
Scorrimento	
di una word dati da 16 bit per n bit	7-57
di 1 bit di n operandi a bit	7-60
di 1 indirizzo di n operandi a word	7-63
Scrittura	
dati di instradamento	8-109
di un gruppo di dati in un file	
registri EEPROM	7-493
Scrittura dati	
in un modulo interfaccia PROFIBUS	11-31
in una stazione intelligente	
(CC-Link)	11-152, 11-158
in una stazione su CC-Link	11-134
in una tabella dati	7-169
in un'altra stazione (CC-Link)	11-127
nel buffer ad aggiornamento	
automatico (serie A)	11-162
nel buffer ad aggiornamento	
automatico (serie Q)	11-166
su altre stazioni	8-82

su buffer fisso (ETHERNET) 11-44

su moduli funzione speciali in
 stazioni I/O remote 8-71

su una stazione locale 8-90

su una stazione remota 8-101

Scrittura dati operando a word su
 una diversa stazione (SWRITE) 8-36

Scrittura dati operando a word su
 una diversa stazione (WRITE) 8-29

Scrittura dati orologio 7-430

Scrittura diretta di un byte in un file registri . . 7-479

Scrittura e lettura da file (System Q) 9-9

Scrittura nel buffer di memoria di
 un modulo funzione speciale 7-191

SECOND, SECONDP, HOUR, HOURP 7-444

SEG. 6-168

Segnale rampa 6-196

SEG, SEGP 7-91

SEND 8-39

SER, SERP, DSER, DSERP 7-78

SFR, SFRP, SFL, SFLP 7-56

SIN, SINP 7-344

SLT, SLTR 7-239

Software 1-2

Somma
 logica con blocchi dati da 16 bit 7-21

Somma dati orologio 7-434

Sorgente dei dati (s) 3-1

SORT, SORTP, DSORT, DSORTP 7-117

Sottrazione dati orologio 7-439

SPD. 6-197

SPREF 10-11

SQR, SQRP 7-368

SREAD 8-18

STC, CLC 7-469

STMODE 10-2

STMR, STMRH 6-187

STRA, STRAR 7-241

Stringa caratteri
 Istruzione STRING 3-22

memorizzazione e spostamento 7-327

operazioni di collegamento 6-74

ricerca 7-333

Stringhe di caratteri
 trasferimento 6-124

STR, STRP, DSTR, DSTRP 7-284

SUB, SUBP 7-156

SUM, SUMP, DSUM, DSUMP 7-84

SWAP, SWAPP 6-143

SWRITE. 8-32

System Q
 Tipi di CPU a singolo processore 1-2

Tipi di CPU multi processore 1-2

S.TO, SP.TO 9-46

T

Tabella
 operandi disponibili A-32

registri di comunicazione (solo serie A) A-120

Registri speciali (D) (solo serie A) A-111

registri speciali (serie Q e System Q) A-74

relé di comunicazione (solo serie A) A-71

relé speciali diagnostici (SM)
 (serie Q e System Q) A-43

Relé speciali e relé diagnostici
 (conversione) A-58

relé speciali (M) (solo serie A) A-64

Tabella CPU della descrizione dell'istruzione . . 4-2

Tabella dei
 Codici di errore
 CPU AnA e AnAS 13-43

Serie A (tranne AnA e AnAS) 13-39

Serie QnA e System Q (CPU
 Q02, Q02H, Q06H, Q12H, Q25H) 13-12

System Q (CPU Q00J, Q00, Q01) 13-2

tempi di elaborazione A-2

TAN, TANP 7-350

Terminazione di un loop FOR/NEXT 7-130

Test dello stato di singoli bit in word
 dati da 16/32 bit 7-70

TEST, TESTP, DTEST, DTESTP 7-69

Timer programmabile (apprendimento) 6-186

Timer speciale 6-188

Tipi di CPU 1-2

Tipi di dati 3-9

TO, DTO 7-190

TRACE, TRACER 9-7

Trame utente
 cancellazione 11-13

lettura 11-8

registrazione 11-13

trasmissione 11-20

Trasferimento blocchi dati BIN identici 6-135

Trasferimento blocchi (System Q) 9-42

Trasferimento dati per Q4ARCPU 10-6

TRUCK 10-6

TTMR 6-185

U		Z	
UDCNT1	6-179	ZCOM	8-7
UDCNT2	6-182	ZNFR	8-63
UINI	11-72	ZNRD	8-77
UNIRD, UNIRDP	9-2	ZNTD	8-69
Unità organizzativa di programma (POU)	3-19	ZNWR	8-81
UNI, UNIP	7-98	ZONE, ZONEP, DZONE, DZONEP	7-410
Uscita ENO	3-35	ZPUSH, ZPUSHP, ZPOP, ZPOPP	7-489
Uscita impulsiva con numero di impulsi regolabile	6-200	ZRRDB, ZRRDBP	7-474
Uscita su dispositivo periferico		ZRWRB, ZRWRBP	7-478
PR	7-197		
PRC	7-202		
Utilizzo di programmi			
microcomputer creati dall'utente	12-2		
V		Simboli	
Valore in virgola mobile come		+, +P, -, -P	6-28
esponente della base e	7-372	=, , >, =	6-5
VAL, VALP, DVAL, DVALP	7-292	\$ =, \$, \$ >, \$ =	6-15
Variabili	4-5	\$MOV, \$MOV P	6-123
Variabili (uso in una istruzione)	3-1	\$+, \$+P	6-73
Verifica			
del campo operando	3-31		
del dato operando	3-33		
Verifica moduli di I/O (registri speciali)	A-104		
W			
WAND, WANDP, DAND, DANDP	7-4		
WDT, WDTP	7-467		
WORD, WORDP	6-97		
WOR, WORP, DOR, DORP	7-14		
WRITE	8-25		
WSUM, WSUMP	7-121		
WTOB, WTOBP, BTOW, BTOWP	7-106		
WXNR, WXNRP, DXNR, DXNRP	7-32		
WXOR, WXORP, DXOR, DXORP	7-23		
X			
XCH, XCHP, DXCH, DXCHP	6-137		
x, xP, /, /P	6-36		

SEDE CENTRALE		DISTRIBUTORI EUROPEI		DISTRIBUTORI EUROPEI		DISTRIBUTORI - EURASIA	
MITSUBISHI ELECTRIC EUROPE B.V. German Branch Gothaer Straße 8 D-40880 Ratingen Telefono: +49 (0)2102 / 486-0 Fax: +49 (0)2102 / 486-1120	EUROPA	GEVA Wiener Straße 89 AT-2500 Baden Telefono: +43 (0)2252 / 85 55 20 Fax: +43 (0)2252 / 488 60	AUSTRIA	HIFLEX AUTOMATISIERUNGSTECHNIK B.V. Wolweverstraat 22 NL-2984 CD Ridderkerk Telefono: +31 (0)180 - 46 60 04 Fax: +31 (0)180 - 44 23 55	OLANDA	Kazpromautomatics Ltd. Mustafina Str. 7/2 KAZ-470046 Karaganda Telefono: +7 7212 / 50 11 50 Fax: +7 7212 / 50 11 50	KAZAKISTAN
MITSUBISHI ELECTRIC EUROPE B.V. French Branch 25, Boulevard des Bouvets F-92741 Nanterre Cedex Telefono: +33 (0)1 / 55 68 55 68 Fax: +33 (0)1 / 55 68 57 57	FRANCIA	ESCO DRIVES & AUTOMATION Culliganlaan 3 BE-1831 Diegem Telefono: +32 (0)2 / 717 64 30 Fax: +32 (0)2 / 717 64 31	BELGIO	Koning & Hartman b.v. Haarlerbergweg 21-23 NL-1101 CH Amsterdam Telefono: +31 (0)20 / 587 76 00 Fax: +31 (0)20 / 587 76 05	OLANDA	DISTRIBUTORI - MEDIO ORIENTE	
MITSUBISHI ELECTRIC EUROPE B.V. Irish Branch Westgate Business Park, Ballymount IRL-Dublin 24 Telefono: +353 (0)1 4198800 Fax: +353 (0)1 4198890	IRLANDA	Koning & Hartman b.v. Woluwelaan 31 BE-1800 Vilvoorde Telefono: +32 (0)2 / 257 02 40 Fax: +32 (0)2 / 257 02 49	BELGIO	MPL Technology Sp. z o.o. Ul. Krakowska 50 PL-32-083 Balice Telefono: +48 (0)12 / 630 47 00 Fax: +48 (0)12 / 630 47 01	POLONIA	ILAN & GAVISH Ltd. 24 Shenkar Str., Kiryat Arie IL-49001 Petah-Tiqva Telefono: +972 (0)3 / 922 18 24 Fax: +972 (0)3 / 924 0761	ISRAELE
MITSUBISHI ELECTRIC EUROPE B.V. Italian Branch Viale Colleoni 7 I-20041 Agrate Brianza (MB) Telefono: +39 039 / 60 53 1 Fax: +39 039 / 60 53 312	ITALIA	TEHNIKON Oktyabrskaya 16/5, Off. 703-711 BY-220030 Minsk Telefono: +375 (0)17 / 210 46 26 Fax: +375 (0)17 / 210 46 26	BIELORUSSIA	AutoCont C.S. s.r.o. Technologická 374/6 CZ-708 00 Ostrava-Pustkovec Telefono: +420 595 691 150 Fax: +420 595 691 199	REP. CECA	TEXEL ELECTRONICS Ltd. 2 Ha'umanut, P.O.B. 6272 IL-42160 Netanya Telefono: +972 (0)9 / 863 39 80 Fax: +972 (0)9 / 885 24 30	ISRAELE
MITSUBISHI ELECTRIC EUROPE B.V. Czech Branch Avenir Business Park, Radlická 714/113a CZ-158 00 Praha 5 Telefono: +420 (0)251 551 470 Fax: +420 (0)251-551-471	REP. CECA	INEA BH d.o.o. Aleja Lipa 56 BA-71000 Sarajevo Telefono: +387 (0)33 / 921 164 Fax: +387 (0)33/ 524 539	BOSNIA E ERZEGOVINA	B:TECH A.S. U Borové 69 CZ-58001 Havlíčkův Brod Telefono: +420 (0)569 777 777 Fax: +420 (0)569-777 778	REP. CECA	CEG INTERNATIONAL Cebaco Center/Block A Autostrade DORA Lebanon - Beirut Telefono: +961 (0)1 / 240 430 Fax: +961 (0)1 / 240 438	LIBANO
MITSUBISHI ELECTRIC EUROPE B.V. Spanish Branch Carretera de Rubí 76-80 E-08190 Sant Cugat del Vallés (Barcelona) Telefono: +34 93563131 // +34 93563131 Fax: +34 935891579	SPAGNA	AKHNATON 4 Andrej Ljapchev Blvd. Pb 21 BG-1756 Sofia Telefono: +359 (0)2 / 817 6004 Fax: +359 (0)2 / 97 44 06 1	BULGARIA	Sirius Trading & Services srl Aleea Lacul Morii Nr. 3 RO-060841 Bucuresti, Sector 6 Telefono: +40 (0)21 / 430 40 06 Fax: +40 (0)21 / 430 40 02	ROMANIA	DISTRIBUTORI - AFRICA	
MITSUBISHI ELECTRIC EUROPE B.V. UK Branch Travellers Lane UK-Hatfield, Herts. AL10 8XB Telefono: +44 (0)1707 / 27 61 00 Fax: +44 (0)1707 / 27 86 95	UK	INEA CR d.o.o. Losinjska 4 a HR-10000 Zagreb Telefono: +385 (0)1 / 36 940 -01/-02/-03 Fax: +385 (0)1 / 36 940 -03	CROAZIA	Craft Con. & Engineering d.o.o. Bulevar Svetog Cara Konstantina 80-86 SER-18106 Nis Telefono: +381 (0)18 / 292-24-4/5 Fax: +381 (0)18 / 292-24-4/5	SERBIA	CBI Ltd. Private Bag 2016 ZA-1600 Isando Telefono: +27 (0)11 / 928 2000 Fax: +27 (0)11 / 392 2354	AFRICA DEL SUD
MITSUBISHI ELECTRIC CORPORATION Office Tower "Z" 14 F 8-12, 1 chome, Harumi Chuo-Ku Tokyo 104-6212 Telefono: +81 3 622 160 60 Fax: +81 3 622 160 75	GIAPPONE	Beijer Electronics A/S Lykkegårdsvej 17, 1. DK-4000 Roskilde Telefono: +45 (0)46 / 75 76 66 Fax: +45 (0)46 / 75 56 26	DANIMARCA	INEA SR d.o.o. Izletnicka 10 SER-113000 Smederevo Telefono: +381 (0)26 / 617 163 Fax: +381 (0)26 / 617 163	SERBIA		
MITSUBISHI ELECTRIC AUTOMATION, Inc. 500 Corporate Woods Parkway Vernon Hills, IL 60061 Telefono: +1 847 478 21 00 Fax: +1 847 478 22 53	USA	Beijer Electronics Eesti OÜ Pärnu mnt.160i EE-11317 Tallinn Telefono: +372 (0)6 / 51 81 40 Fax: +372 (0)6 / 51 81 49	ESTONIA	AutoCont Control s.r.o. Radlinského 47 SK-02601 Dolny Kubin Telefono: +421 (0)43 / 5868210 Fax: +421 (0)43 / 5868210	SLOVACCHIA		
		Beijer Electronics OY Jaakonkatu 2 FIN-01620 Vantaa Telefono: +358 (0)207 / 463 500 Fax: +358 (0)207 / 463 501	FINLANDIA	CS MTrade Slovensko, s.r.o. Vajanskeho 58 SK-92101 Piestany Telefono: +421 (0)33 / 7742 760 Fax: +421 (0)33 / 7735 144	SLOVACCHIA		
		UTECA A.B.E.E. 5, Mavrogenous Str. GR-18542 Piraeus Telefono: +30 211 / 1206 900 Fax: +30 211 / 1206 999	GRECIA	INEA d.o.o. Stegne 11 SI-1000 Ljubljana Telefono: +386 (0)1 / 513 8100 Fax: +386 (0)1 / 513 8170	SLOVENIA		
		Beijer Electronics SIA Vestienas iela 2 LV-1035 Riga Telefono: +371 (0)784 / 2280 Fax: +371 (0)784 / 2281	LETTONIA	Beijer Electronics AB Box 426 SE-20124 Malmö Telefono: +46 (0)40 / 35 86 00 Fax: +46 (0)40 / 35 86 02	SVEZIA		
		Beijer Electronics UAB Savanoriu Pr. 187 LT-02300 Vilnius Telefono: +370 (0)5 / 232 3101 Fax: +370 (0)5 / 232 2980	LITUANIA	Econotec AG Hinterdorfstr. 12 CH-8309 Nürensdorf Telefono: +41 (0)44 / 838 48 11 Fax: +41 (0)44 / 838 48 12	SVIZZERA		
		ALFATRADA Ltd. 99, Paola Hill Malta- Paola PLA 1702 Telefono: +356 (0)21 / 697 816 Fax: +356 (0)21 / 697 817	MALTA	GTS Bayraktar Bulvari Nutuk Sok. No:5 TR-34775 Yukari Dudullu-Umraniye-ISTANBUL Telefono: +90 (0)216 526 39 90 Fax: +90 (0)216 526 39 95	TURCHIA		
		INTEHSIS srl bld. Traian 23/1 MD-2060 Kishinev Telefono: +373 (0)22 / 66 4242 Fax: +373 (0)22 / 66 4280	MOLDAVIA	CSC Automation Ltd. 4-B, M. Raskovoyi St. UA-02660 Kiev Telefono: +380 (0)44 / 494 33 55 Fax: +380 (0)44 / 494-33-66	UCRAINA		
		Beijer Electronics AS Postboks 487 NO-3002 Drammen Telefono: +47 (0)32 / 24 30 00 Fax: +47 (0)32 / 84 85 77	NORVEGIA	MELTRADE Ltd. Fertő utca 14. HU-1107 Budapest Telefono: +36 (0)1 / 431-9726 Fax: +36 (0)1 / 431-9727	UNGHERIA		