

MELSEC-Q/L Structured Programming Manual

Special Instructions



(Always read these instructions before using this product.)

Before using the MELSEC-Q series and MELSEC-L series programmable controllers, thoroughly read the manuals attached to the products and the relevant manuals introduced in the attached manuals. Also pay careful attention to safety and handle the products properly.

Please keep this manual in a place where it is accessible when required and always forward it to the end user.

CONDITIONS OF USE FOR THE PRODUCT

(1) Mitsubishi programmable controller ("the PRODUCT") shall be used in conditions;
 i) where any problem, fault or failure occurring in the PRODUCT, if any, shall not lead to any major or serious accident; and

ii) where the backup and fail-safe function are systematically or automatically provided outside of the PRODUCT for the case of any problem, fault or failure occurring in the PRODUCT.

(2) The PRODUCT has been designed and manufactured for the purpose of being used in general industries.

MITSUBISHI SHALL HAVE NO RESPONSIBILITY OR LIABILITY (INCLUDING, BUT NOT LIMITED TO ANY AND ALL RESPONSIBILITY OR LIABILITY BASED ON CONTRACT, WARRANTY, TORT, PRODUCT LIABILITY) FOR ANY INJURY OR DEATH TO PERSONS OR LOSS OR DAMAGE TO PROPERTY CAUSED BY the PRODUCT THAT ARE OPERATED OR USED IN APPLICATION NOT INTENDED OR EXCLUDED BY INSTRUCTIONS, PRECAUTIONS, OR WARNING CONTAINED IN MITSUBISHI'S USER, INSTRUCTION AND/OR SAFETY MANUALS, TECHNICAL BULLETINS AND GUIDELINES FOR the PRODUCT. ("Prohibited Application")

Prohibited Applications include, but not limited to, the use of the PRODUCT in;

- Nuclear Power Plants and any other power plants operated by Power companies, and/or any other cases in which the public could be affected if any problem or fault occurs in the PRODUCT.
- Railway companies or Public service purposes, and/or any other cases in which establishment of a special quality assurance system is required by the Purchaser or End User.
- Aircraft or Aerospace, Medical applications, Train equipment, transport equipment such as Elevator and Escalator, Incineration and Fuel devices, Vehicles, Manned transportation, Equipment for Recreation and Amusement, and Safety devices, handling of Nuclear or Hazardous Materials or Chemicals, Mining and Drilling, and/or other applications where there is a significant risk of injury to the public or property.

Notwithstanding the above, restrictions Mitsubishi may in its sole discretion, authorize use of the PRODUCT in one or more of the Prohibited Applications, provided that the usage of the PRODUCT is limited only for the specific applications agreed to by Mitsubishi and provided further that no special quality assurance or fail-safe, redundant or other safety features which exceed the general specifications of the PRODUCTs are required. For details, please contact the Mitsubishi representative in your region.

The manual number is written at the bottom left of the back cover.

| Print date | Manual number | Revision |
|------------|--------------------|---|
| Jul., 2008 | SH(NA)-080785ENG-A | First edition |
| Jan., 2009 | SH(NA)-080785ENG-B | Model Addition Q00UJ, Q00U, Q01U, Q10UDH, Q10UDEH, Q20UDH, Q20UDEH Addition MANUALS, Section 2.2.5, Section 5.5 Correction GENERIC TERMS AND ABBREVIATIONS IN THIS MANUAL, Section 1.1, Section 1.2, Section 2.2.5 to 2.2.7 changed to Section 2.2.6 to Section 2.2.8, Section 5.5 to Section 5.7 changed to Section 5.6 to Section 5.8 |
| Jul., 2009 | SH(NA)-080785ENG-C | Model Addition Q00J, Q00, Q01 Correction PURPOSE OF THIS MANUAL is changed to Section 1.1, GENERIC TERMS AND ABBREVIATIONS IN THIS MANUAL is changed to Section 1.2, Section 1.1 is changed to Section 1.3, Section 1.2 is changed to Section 1.4, Chapter 4, Program examples are added in Chapter 5 |
| Jan., 2010 | SH(NA)-080785ENG-D | Model Addition L02, L26-BT Addition CONDITIONS OF USE FOR THE PRODUCT, Section 2.3, Section 2.4, Section 2.5, Section 2.6, Chapter 8, Chapter 9 Correction MANUALS, Section 1.1, Section 1.2, Section 1.3, Section 1.4, Section 2.2, Chapter 4, Section 5.1.2, Section 5.1.3, Section 5.6.1, Section 5.6.2, Section 5.6.3, Section 5.6.4. Section 5.7.1, Section 5.7.2, Section 5.7.3, Section 5.7.4, Section 5.4.10 to Section 5.4.25 are changed to Section 5.4.9 to Section 5.4.24, Section 5.8 is changed to Section 5.5, Section 5.6 to Section 5.7 are changed to Chapter 6, Section 5.5 is changed to Chapter 7 Deletion Section 5.4.9 |

| Print date | Manual number | Revision |
|------------|--------------------|--|
| Apr., 2010 | SH(NA)-080785ENG-E | Model AdditionQ50UDEH, Q100UDEHAdditionSection 2.2.2, Section 5.2.15, Section 5.4.15, Section 5.4.16, Section 5.4.17CorrectionMANUALS, Section 1.2, Section 1.4, Section 2.2.4, Chapter 4, Section 5.3.7,Section 5.4, Section 5.4.1, Section 5.4.3, Section 5.4.5, Section 5.4.6, Section 5.4.7,Section 5.4.8, Section 5.4.13, Section 5.4.14, Section 7.1,Section 5.4.15 to Section 5.4.24 are changed to Section 5.4.18 to Section 5.4.27. |
| Sep., 2010 | SH(NA)-080785ENG-F | AdditionSection 2.7, Section 5.4.16, Section 5.4.17, Chapter 10CorrectionMANUALS, Section 1.1, Section 1.3, Section 1.4, Section 2.1,Chapter 4, Section 5.1.2, Section 5.1.3, Section 5.4.1, Section 5.4.2, Section 5.4.4,Section 5.4.5. Section 5.4.8, Section 5.4.10, Section 5.4.12, Section 5.4.13,Section 5.4.14, Section 5.4.15, Section 5.4.18, Section 5.4.19, Section 5.4.20,Section 5.4.21, Section 5.4.22, Section 5.4.23, Section 5.4.29, Section 5.4.33,Section 2.2.2 to Section 2.2.4 are changed to Section 2.2.3 to Section 2.2.4,Section 5.2 to Section 5.4 are changed to Section 5.3 to Section 5.4,Section 5.5 is changed to Section 5.2DeletionSection 5.4.17, Section 5.4.18 |
| Jan., 2011 | SH(NA)-080785ENG-G | Correction Section 1.1, Section 5.4.16, Section 5.4.17 |
| Mar., 2011 | SH(NA)-080785ENG-H | Correction MANUALS, Section 1.1, Section 1.4, Section 2.1, Section 2.2.3, Section 5.1.1, Section 5.1.2, Section 5.1.3, Section 5.3.5, Section 5.3.7, Section 5.3.14, Section 5.3.15, Section 5.4.1, Section 5.4.2, Section 5.4.5, Section 5.4.6, Section 5.4.8, Section 5.4.9, Section 5.4.10, Section 5.4.11, Section 5.4.12, Section 5.4.13, Section 5.4.14, Section 5.1.15, Section 5.4.18, Section 5.4.19, Section 5.4.20, Section 5.4.21, Section 5.4.28, Section 5.4.31, Section 5.4.32, Section 5.4.33, Section 7.1, Section 7.2, Section 7.5, Section 7.8, Section 7.9 |

| Print date | Manual number | Revision |
|------------|--------------------|--|
| Jul., 2011 | SH(NA)-080785ENG-I | Model Addition L02-P, L26-PBT Correction Section 1.2, Section 1.4, Chapter 4, Section 5.3.2, Section 5.3.4, Section 5.3.15, Section 5.4.16, Section 5.4.17, Section 5.4.18, Section 5.4.21, Section 5.4.22, Section 5.4.23, Section 5.4.24, Section 5.4.25, Section 5.4.26, Section 5.4.27, Section 5.4.29, Section 5.4.32, Section 5.4.34, Section 7.1, Section 7.2, Section 8.1.5, Section 10.1.1, Section 10.1.2 |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |

Japanese manual version SH-080738-L

This manual confers no industrial property rights or any rights of any other kind, nor does it confer any patent licenses. Mitsubishi Electric Corporation cannot be held responsible for any problems involving industrial property rights which may occur as a result of using the contents noted in this manual.

© 2008 MITSUBISHI ELECTRIC CORPORATION

INTRODUCTION

Thank you for purchasing the Mitsubishi MELSEC-Q series/MELSEC-L series programmable controller. Before using the product, thoroughly read this manual to develop full familiarity with the programming specifications to ensure correct use.

Please forward this manual to the end user.

CONTENTS

| SAFETY PRECAUTIONS | A - 1 |
|-----------------------------------|--------|
| CONDITIONS OF USE FOR THE PRODUCT | |
| REVISIONS | A - 3 |
| NTRODUCTION | A - 6 |
| CONTENTS | A - 6 |
| MANUALS | A - 10 |

| 1. OVERVIEW | 1 - 1 to 1 - 10 |
|---|------------------|
| 1.1 Purpose of This Manual | 1 - 2 |
| 1.2 Generic Terms and Abbreviations in This Manual | 1 - 5 |
| 1.3 Explanation Content in This Manual | 1 - 6 |
| 1.4 Modules and Versions Applicable to Instructions | 1 - 8 |
| 2. INSTRUCTION TABLES | 2 - 1 to 2 - 16 |
| 2.1 How to Read Instruction Tables | 2 - 2 |
| 2.2 Module Dedicated Instruction | 2 - 3 |
| 2.2.1 Analog instruction 2.2.2 Positioning instruction | |
| 2.2.3 Serial communication 2.2.4 Network dedicated instruction | |
| 2.3 PID Control Instruction | 2 - 10 |
| 2.3.1 PID control instruction (inexact differential)2.3.2 PID control instruction (exact differential) | |
| 2.4 Socket Communication Function Instruction | 2 - 11 |
| 2.5 Built-in I/O Function Instruction | 2 - 12 |
| 2.5.1 Positioning function dedicated instruction2.5.2 Counter function dedicated instruction | |
| 2.6 Data Logging Function Instruction | 2 - 15 |
| 2.7 SFC Control Instruction | 2 - 15 |
| 3. CONFIGURATION OF INSTRUCTIONS | 3 - 1 to 3 - 4 |
| 3.1 Configuration of Instructions | 3 - 2 |
| 4. HOW TO READ INSTRUCTIONS | 4 - 1 to 4 - 4 |
| 5. MODULE DEDICATED INSTRUCTION | 5 - 1 to 5 - 256 |
| 5.1 Analog Instruction | 5 - 2 |

| | OFFGAN instruction | 5 - 2 |
|--|--|---|
| 5.1.2 | OGLOAD instruction | 5 - 4 |
| 5.1.3 | OGSTOR instruction | 5 - 27 |
| 5.2 Pos | sitioning Instruction | 5 - 51 |
| - | ABRST instruction | |
| | PSTRT instruction | |
| | TEACH instruction | |
| | PFWRT instruction | |
| | PINIT instruction | |
| 5.3 Ser | ial Communication | 5 - 64 |
| 5.3.1 | ONDEMAND instruction | |
| 5.3.2 | OUTPUT instruction | 5 - 68 |
| 5.3.3 | INPUT instruction | 5 - 71 |
| 5.3.4 | BIDOUT instruction | 5 - 74 |
| 5.3.5 | BIDIN instruction | 5 - 77 |
| 5.3.6 | SPBUSY instruction | 5 - 79 |
| 5.3.7 | CSET instruction (receive data clear) | 5 - 80 |
| 5.3.8 | BUFRCVS instruction | |
| | PRR instruction | |
| | CSET instruction (initial setting) | |
| | CSET instruction (programmable controller CPU monitor) | |
| | PUTE instruction | |
| | GETE instruction | |
| | UINI instruction | |
| | CPRTCL instruction | |
| | | 5 - 115 |
| | work Dedicated Instruction | |
| | RIRD instruction | |
| | RIWT instruction | |
| | | |
| | RIRCV instruction | 5 - 125 |
| 5.4.4 | RIRCV instruction | 5 - 125 5 - 129 |
| 5.4.4 5.4.5 | RIRCV instruction RISEND instruction RIFR instruction | 5 - 125 5 - 129 5 - 133 |
| 5.4.4 5.4.5 5.4.6 | RIRCV instruction RISEND instruction RIFR instruction RITO instruction | 5 - 125 5 - 129 5 - 133 5 - 135 |
| 5.4.4 5.4.5 5.4.6 5.4.7 | RIRCV instruction RISEND instruction RIFR instruction RITO instruction RLPASET instruction | 5 - 125 5 - 129 5 - 133 5 - 135 5 - 137 |
| 5.4.4 5.4.5 5.4.6 5.4.7 5.4.8 | RIRCV instruction | 5 - 125 5 - 129 5 - 133 5 - 135 5 - 137 5 - 144 |
| 5.4.4 5.4.5 5.4.6 5.4.7 5.4.8 5.4.9 | RIRCV instruction | 5 - 125 5 - 129 5 - 133 5 - 135 5 - 137 5 - 144 5 - 150 |
| 5.4.4 5.4.5 5.4.6 5.4.7 5.4.8 5.4.9 5.4.10 | RIRCV instruction | 5 - 125 5 - 129 5 - 133 5 - 135 5 - 137 5 - 137 5 - 144 5 - 150 5 - 154 |
| 5.4.4 5.4.5 5.4.6 5.4.7 5.4.8 5.4.9 5.4.10 5.4.11 | RIRCV instruction | 5 - 125 5 - 129 5 - 133 5 - 135 5 - 137 5 - 144 5 - 150 5 - 154 5 - 161 |
| 5.4.4 5.4.5 5.4.6 5.4.7 5.4.8 5.4.9 5.4.10 5.4.11 5.4.12 | RIRCV instruction RISEND instruction RIFR instruction RITO instruction RLPASET instruction READ instruction SREAD instruction WRITE instruction SWRITE instruction SEND instruction | 5 - 125 5 - 129 5 - 133 5 - 135 5 - 137 5 - 144 5 - 150 5 - 154 5 - 161 5 - 165 |
| 5.4.4 5.4.5 5.4.6 5.4.7 5.4.8 5.4.9 5.4.10 5.4.10 5.4.12 5.4.13 | RIRCV instruction | 5 - 125 5 - 129 5 - 133 5 - 135 5 - 137 5 - 144 5 - 150 5 - 154 5 - 161 5 - 165 5 - 173 |
| 5.4.4 5.4.5 5.4.6 5.4.7 5.4.8 5.4.9 5.4.10 5.4.11 5.4.12 5.4.13 5.4.14 | RIRCV instruction RISEND instruction RIFR instruction RITO instruction RLPASET instruction READ instruction SREAD instruction WRITE instruction SWRITE instruction SEND instruction RECV instruction RECV instruction | 5 - 125 5 - 129 5 - 133 5 - 135 5 - 137 5 - 144 5 - 150 5 - 154 5 - 161 5 - 165 5 - 173 5 - 178 |
| 5.4.4 5.4.5 5.4.6 5.4.7 5.4.8 5.4.9 5.4.10 5.4.11 5.4.12 5.4.13 5.4.14 5.4.15 | RIRCV instruction RISEND instruction RIFR instruction RITO instruction RLPASET instruction READ instruction SREAD instruction WRITE instruction SWRITE instruction SEND instruction RECV instruction RECVS instruction RECVS instruction | 5 - 125 5 - 129 5 - 133 5 - 135 5 - 137 5 - 144 5 - 150 5 - 154 5 - 161 5 - 165 5 - 173 5 - 178 5 - 181 |
| 5.4.4 5.4.5 5.4.6 5.4.7 5.4.8 5.4.9 5.4.10 5.4.10 5.4.12 5.4.12 5.4.13 5.4.14 5.4.15 5.4.16 | RIRCV instruction | 5 - 125 5 - 129 5 - 133 5 - 135 5 - 137 5 - 144 5 - 150 5 - 154 5 - 161 5 - 165 5 - 173 5 - 178 5 - 181 5 - 190 |
| 5.4.4 5.4.5 5.4.6 5.4.7 5.4.8 5.4.9 5.4.10 5.4.11 5.4.12 5.4.13 5.4.14 5.4.15 5.4.16 5.4.17 | RIRCV instruction | 5 - 125 5 - 129 5 - 133 5 - 135 5 - 137 5 - 144 5 - 150 5 - 154 5 - 161 5 - 165 5 - 173 5 - 178 5 - 181 5 - 190 5 - 193 |
| 5.4.4 5.4.5 5.4.6 5.4.7 5.4.8 5.4.9 5.4.10 5.4.10 5.4.12 5.4.13 5.4.14 5.4.15 5.4.16 5.4.17 5.4.18 | RIRCV instruction RISEND instruction RIFR instruction RITO instruction RLPASET instruction READ instruction SREAD instruction WRITE instruction SWRITE instruction SEND instruction RECV instruction RECVS instruction REQ Instruction | 5 - 125 5 - 129 5 - 133 5 - 135 5 - 137 5 - 144 5 - 150 5 - 154 5 - 161 5 - 165 5 - 173 5 - 178 5 - 181 5 - 190 5 - 193 5 - 197 |
| 5.4.4 5.4.5 5.4.6 5.4.7 5.4.8 5.4.9 5.4.10 5.4.12 5.4.12 5.4.13 5.4.13 5.4.15 5.4.16 5.4.17 5.4.18 5.4.19 | RIRCV instruction | 5 - 125 5 - 129 5 - 133 5 - 135 5 - 137 5 - 144 5 - 150 5 - 154 5 - 161 5 - 165 5 - 173 5 - 178 5 - 181 5 - 190 5 - 193 5 - 197 5 - 200 |
| 5.4.4 5.4.5 5.4.6 5.4.7 5.4.8 5.4.9 5.4.10 5.4.12 5.4.12 5.4.13 5.4.14 5.4.15 5.4.16 5.4.17 5.4.18 5.4.20 | RIRCV instruction | 5 - 125 5 - 129 5 - 133 5 - 135 5 - 137 5 - 144 5 - 150 5 - 154 5 - 161 5 - 165 5 - 173 5 - 178 5 - 178 5 - 181 5 - 190 5 - 193 5 - 197 5 - 200 5 - 203 |
| 5.4.4 5.4.5 5.4.6 5.4.7 5.4.8 5.4.9 5.4.10 5.4.10 5.4.12 5.4.13 5.4.14 5.4.15 5.4.16 5.4.17 5.4.18 5.4.19 5.4.20 5.4.21 | RIRCV instruction RISEND instruction RIFR instruction RITO instruction RLPASET instruction READ instruction SREAD instruction WRITE instruction SWRITE instruction SEND instruction RECV instruction RECVS instruction REQ instruction RTMRD instruction RTMRD instruction RTMRD instruction | 5 - 125 5 - 129 5 - 133 5 - 135 5 - 137 5 - 144 5 - 150 5 - 154 5 - 161 5 - 165 5 - 173 5 - 178 5 - 178 5 - 181 5 - 190 5 - 193 5 - 197 5 - 200 5 - 203 5 - 205 |
| 5.4.4 5.4.5 5.4.6 5.4.7 5.4.8 5.4.9 5.4.10 5.4.12 5.4.12 5.4.13 5.4.13 5.4.14 5.4.15 5.4.16 5.4.16 5.4.17 5.4.18 5.4.19 5.4.20 5.4.21 | RIRCV instruction RISEND instruction RIFR instruction RITO instruction RLPASET instruction READ instruction SREAD instruction WRITE instruction SWRITE instruction SEND instruction RECV instruction RECV instruction REQ INSTRUCTION R | 5 - 125 5 - 129 5 - 133 5 - 135 5 - 137 5 - 144 5 - 150 5 - 154 5 - 161 5 - 165 5 - 173 5 - 173 5 - 178 5 - 178 5 - 190 5 - 190 5 - 193 5 - 197 5 - 200 5 - 203 5 - 205 5 - 208 |
| 5.4.4 5.4.5 5.4.6 5.4.7 5.4.8 5.4.9 5.4.10 5.4.12 5.4.12 5.4.13 5.4.14 5.4.15 5.4.16 5.4.16 5.4.17 5.4.18 5.4.20 5.4.21 5.4.21 | RIRCV instruction | 5 - 125 5 - 129 5 - 133 5 - 135 5 - 137 5 - 144 5 - 150 5 - 154 5 - 161 5 - 165 5 - 173 5 - 178 5 - 178 5 - 190 5 - 193 5 - 193 5 - 197 5 - 200 5 - 203 5 - 208 5 - 210 |
| 5.4.4 5.4.5 5.4.6 5.4.7 5.4.8 5.4.9 5.4.10 5.4.10 5.4.12 5.4.13 5.4.14 5.4.15 5.4.16 5.4.16 5.4.17 5.4.18 5.4.19 5.4.20 5.4.21 5.4.21 5.4.23 5.4.24 | RIRCV instruction RISEND instruction RIFR instruction RITO instruction RLPASET instruction READ instruction SREAD instruction WRITE instruction SWRITE instruction SEND instruction RECV instruction RECV instruction REQ INSTRUCTION R | 5 - 125 5 - 129 5 - 133 5 - 135 5 - 137 5 - 144 5 - 150 5 - 154 5 - 161 5 - 165 5 - 173 5 - 178 5 - 178 5 - 178 5 - 178 5 - 190 5 - 193 5 - 193 5 - 200 5 - 203 5 - 208 5 - 210 5 - 212 |

| 5.4.26 CLOSE instruction | |
|----------------------------|--|
| 5.4.27 BUFRCV instruction | |
| 5.4.28 BUFRCVS instruction | |
| 5.4.29 BUFSND instruction | |
| 5.4.30 ERRCLR instruction | |
| 5.4.31 ERRRD instruction | |
| 5.4.32 UINI instruction | |
| 5.4.33 MRECV instruction | |
| 5.4.34 MSEND instruction | |
| | |

6. PID CONTROL INSTRUCTION

6 - 1 to 6 - 30

| 6.1 PI | D Control Instruction (Inexact Differential) | 6 - 2 |
|--------|--|--------|
| 6.1.1 | PIDINIT instruction | |
| 6.1.2 | PIDCONT instruction | |
| 6.1.3 | PIDSTOP instruction and PIDRUN instruction | |
| 6.1.4 | PIDPRMW instruction | |
| 6.2 PI | D Control Instruction (Exact Differential) | 6 - 16 |
| 6.2.1 | PIDINIT instruction | |
| 6.2.2 | PIDCONT instruction | |
| 6.2.3 | PIDSTOP instruction and PIDRUN instruction | |
| 6.2.4 | PIDPRMW instruction | |

| 7. SOCKET COMMUNICATION FUNCTION INSTRUCTION | 7 - 1 to 7 - 26 |
|--|-----------------|
| 7.1 SOCOPEN Instruction | 7 - 2 |
| 7.2 SOCCLOSE Instruction | 7 - 5 |
| 7.3 SOCRCV Instruction | 7 - 8 |
| 7.4 SOCRCVS Instruction | 7 - 11 |
| 7.5 SOCSND Instruction | 7 - 13 |
| 7.6 SOCCINF Instruction | 7 - 16 |
| 7.7 SOCCSET Instruction | 7 - 19 |
| 7.8 SOCRMODE Instruction | 7 - 22 |
| 7.9 SOCRDATA Instruction | 7 - 24 |

| 8. BUIL | T-IN I/O FUNCTION INSTRUCTION | 8 - 1 to 8 - 30 |
|---------|--|-----------------|
| 8.1 Pc | sitioning Function Dedicated Instruction | 8 - 2 |
| 8.1.1 | IPPSTRT instruction | |
| 8.1.2 | IPDSTRT instruction | |
| 8.1.3 | IPSIMUL instruction | |
| 8.1.4 | IPOPR instruction | |
| 8.1.5 | IPJOG instruction | |
| 8.1.6 | IPABRST instruction | |
| 8.1.7 | IPSTOP instruction | |
| 8.1.8 | IPSPCHG instruction | |
| 8.1.9 | IPTPCHG instruction | |
| 8.2 Co | ounter Function Dedicated Instruction | 8 - 18 |
| 8.2.1 | ICCNTRD instruction | |
| 8.2.2 | ICRNGWR instruction | |
| 8.2.3 | ICPREWR instruction | |

I

| 8.2.4 ICLI HRD Instruction | |
|---|------------------------|
| 8.2.5 ICSMPRD instruction | |
| | |
| 8.2.7 ICFCNT instruction | |
| 8.2.8 ICRCNT instruction | |
| 8.2.9 ICPLSRD instruction | |
| 8.2.10 ICPWM instruction | |
| 9. DATA LOGGING FUNCTION INSTRUCTION | 9 - 1 to 9 - 4 |
| 9.1 LOGTRG Instruction, LOGTRGR Instruction | 9 - 2 |
| 10. SFC CONTROL INSTRUCTION | 10 - 1 to 10 - 6 |
| 10.1 SFC Control Instruction | 10 - 2 |
| 10.1.1 SFCSCOMR instruction | |
| 10.1.2 SFCTCOMR instruction | |
| INDEX | Index - 1 to Index - 4 |

MANUALS

Related manuals

The manuals related to this product are shown below.

Refer to the following tables when ordering required manuals.

(1) Structured programming

| Manual name | Manual number (Model code) |
|---|-------------------------------|
| MELSEC-Q/L/F Structured Programming Manual (Fundamentals) | |
| Explains the programming method, types of programming languages, and other information required to create | SH-080782ENG |
| structured programs. | (13JW06) |
| (Sold separately) | |
| MELSEC-Q/L Structured Programming Manual (Common Instructions) | |
| Explains the specifications and functions of common instructions such as sequence instructions, basic instructions, | SH-080783ENG |
| and application instructions that can be used in structured programs. | (13JW07) |
| (Sold separately) | |
| MELSEC-Q/L Structured Programming Manual (Application Functions) | SH-080784ENG |
| Explains the specifications and functions of application functions that can be used in structured programs. | (13JW08) |
| (Sold separately) | (155000) |

(2) Operation of GX Works2

| Manual name | Manual number (Model code) |
|--|-------------------------------|
| GX Works2 Version 1 Operating Manual (Common) Explains the system configuration of GX Works2 and the functions common to Simple project and Structured project such as parameter setting, operation method for the online function. (Sold separately) | SH-080779ENG (13JU63) |
| GX Works2 Version 1 Operating Manual (Structured Project) Explains operation methods such as creating and monitoring programs in Structured project of GX Works2. (Sold separately) | SH-080781ENG (13JU65) |
| GX Works2 Beginner's Manual (Structured Project) Explains fundamental operation methods such as creating, editing, and monitoring programs in Structured project for users inexperienced with GX Works2. (Sold separately) | SH-080788ENG (13JZ23) |

⊠POINT -

The Operating Manuals are included on the CD-ROM of the software package in a PDF file format. Manuals in printed form are sold separately. Order a manual by quoting the manual number (model code) listed in the table above.

(3) Detailed specifications of instructions

Analog instruction

| Manual name | Manual number (Model code) |
|---|---------------------------------------|
| Analog-Digital Converter Module User's Manual Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the Q64AD, Q68ADV, and Q68ADI. | (Model code) SH-080055 (13JR03) |
| (Sold separately) | (133703) |
| Channel Isolated High Resolution Analog-Digital Converter Module Channel Isolated High Resolution Analog-Digital Converter Module (With Signal Conditioning Function) User's Manual | SH-080277 |
| Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the Q64AD-GH and Q62AD-DGH. | (13JR51) |
| (Sold separately) | |
| Channel Isolated Analog-Digital Converter Module/Channel Isolated Analog-Digital Converter Module (With Signal Conditioning Function) User's Manual Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the Q68AD-G and Q66AD-DG. | SH-080647ENG (13JR96) |
| (Sold separately) | |
| Digital-Analog Converter Module User's Manual Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the Q62DAN, Q64DAN, Q68DAVN, and Q68DAIN. | SH-080054 (13JR02) |
| (Sold separately) | |
| Channel Isolated Digital-Analog Converter Module User's Manual Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the Q62DA-FG. | SH-080281E (13JR52) |
| (Sold separately) | |
| Channel Isolated Digital-Analog Converter Module User's Manual Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the Q66DA-G. | SH-080648ENG (13JR97) |
| (Sold separately) | |
| RTD Input Module Channel Isolated RTD Input Module User's Manual Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the Q64RD and Q64RD-G. | SH-080142 (13JR31) |
| (Sold separately) | |
| Thermocouple Input Module Channel Isolated Thermocouple/Micro Voltage Input Module User's Manual Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the Q64TD and Q64TDV-GH. | SH-080141 (13JR30) |
| (Sold separately) | |
| Channel Isolated Thermocouple Input Module User's Manual Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the Q68TD-G-H01/Q68TD-G-H02. | SH-080795ENG (13JZ26) |
| (Sold separately) | |
| Channel Isolated RTD Input Module User's Manual Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the Q68RD3-G. | SH-080722ENG (13JZ06) |
| (Sold separately) | - |
| Q61LD Load Cell Input Module User's Manual Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the Q61LD. | SH-080821ENG (13JZ31) |
| (Sold separately) | · |
| MELSEC-L Analog-Digital Converter Module User's Manual Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the analog-digital converter module. | SH-080899ENG (13JZ42) |
| (Sold separately) | |
| MELSEC-L Digital-Analog Converter Module User's Manual Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the digital-analog converter module. | SH-080900ENG (13JZ43) |
| (Sold separately) | |

Positioning instruction

| Manual name | Manual number (Model code) |
|--|-------------------------------|
| Type QD75P/QD75D Positioning Module User's Manual (Details) | |
| Explains the system configuration, performance specifications, functions, handling, procedures before operation, and | SH-080058 |
| troubleshooting of the QD75P1/QD75P2/QD75P4 and QD75D1/QD75D2/QD75D4. | (13JR09) |
| (Sold separately) | |
| Type QD75M Positioning Module User's Manual (Details) | |
| Explains the system configuration, performance specifications, functions, handling, procedures before operation, and | IB-0300062 |
| troubleshooting of the QD75M1/QD75M2/QD75M4. | (1XB752) |
| (Sold separately) | |
| Type QD75MH Positioning Module User's Manual (Details) | |
| Explains the system configuration, performance specifications, functions, handling, procedures before operation, and | IB-0300117 |
| troubleshooting of the QD75MH1/QD75MH2/QD75MH4. | (1XB917) |
| (Sold separately) | |

Serial communication

| Manual name | Manual number (Model code) |
|--|-------------------------------|
| Q Corresponding Serial Communication Module User's Manual (Basic) | |
| Explains the overview for use of the module, applicable system configuration, specifications, procedures before | SH-080006 |
| operation, fundamental data communication with external devices, maintenance, inspection, and troubleshooting. | (13JL86) |
| (Sold separately) | |
| MELSEC-L Serial Communication Module User's Manual (Basic) | |
| Explains the overview for use of the module, applicable system configuration, specifications, procedures before | SH-080894ENG |
| operation, fundamental data communication with external devices, maintenance, inspection, and troubleshooting. | (13JZ40) |
| (Sold separately) | |
| MELSEC-Q/L Serial Communication Module User's Manual (Application) | |
| Explains the specifications and usage of special functions of the module, settings for special functions, and data | SH-080007 |
| communication with external devices. | (13JL87) |
| (Sold separately) | |

Network dedicated instruction

| Control & Communication Link System Master/Local Module User's Manual SH-080394E Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the SH-080394E QJ61BT11N. (Sold separately) MELSEC-L CC-Link System Master/Local Module User's Manual SH-080395ENG Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the SH-080895ENG built-in CC-Link and CC-Link system master/local modules. (Sold separately) CC-Link IE Controller Network Reference Manual SH-080668ENG Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the SH-080668ENG CC-Link IE Controller Network Reference Manual SH-080668ENG Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the SH-080668ENG MELSEC-Q CC-Link IE Field Network Master/Local Module User's Manual SH-080917ENG Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local SH-080972ENG MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual SH-080972ENG Explains the specifications, procedures before operation, system configura |
|--|
| QJ61BT11N. (13JR64) (Sold separately) MELSEC-L CC-Link System Master/Local Module User's Manual SH-080895ENG Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the built-in CC-Link and CC-Link system master/local modules. SH-080895ENG CC-Link IE Controller Network Reference Manual SH-080668ENG Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the CC-Link IE Controller Network. SH-080668ENG MELSEC-Q CC-Link IE Field Network Master/Local Module User's Manual SH-080668ENG Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network Master/Local Module User's Manual SH-080917ENG MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual (Sold separately) (13JZ47) Module. (Sold separately) (13JZ47) (13JZ47) (13JZ47) <t< td=""></t<> |
| (Sold separately) MELSEC-L CC-Link System Master/Local Module User's Manual Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the built-in CC-Link and CC-Link system master/local modules. SH-080895ENG (13JZ41) CC-Link IE Controller Network Reference Manual (Sold separately) Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the CC-Link IE Controller Network. SH-080668ENG (13JV16) CC-Link IE Controller Network. (Sold separately) MELSEC-Q CC-Link IE Field Network Master/Local Module User's Manual SH-080917ENG (13JZ47) module. (Sold separately) MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual SH-080917ENG (13JZ47) module. (Sold separately) MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual SH-080917ENG (13JZ47) MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual (Sold separately) MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual (Sold separately) MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual (Sold separately) Melsech-L CC-Link IE Field Network Master/Local Module User's Manual (Sold separately) Melsech-L CC-Link IE Field Network Master/Local Module User's Manual <t< td=""></t<> |
| MELSEC-L CC-Link System Master/Local Module User's Manual SH-080895ENG Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the built-in CC-Link and CC-Link system master/local modules. SH-080895ENG CC-Link IE Controller Network Reference Manual SH-080668ENG Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the CC-Link IE Controller Network. SH-080668ENG CC-Link IE Controller Network. (Sold separately) MELSEC-Q CC-Link IE Field Network Master/Local Module User's Manual SH-080068ENG Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network Master/Local Module User's Manual SH-080917ENG MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual SH-080917ENG module. (Sold separately) MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual SH-080972ENG module. (Sold separately) MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual SH-080972ENG module. (Sold separately) (13JZ47) MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual SH-080972ENG module. (Sold separately) (13JZ5 |
| Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the built-in CC-Link and CC-Link system master/local modules. SH-080895ENG (13JZ41) CC-Link IE Controller Network Reference Manual Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the CC-Link IE Controller Network. SH-080668ENG (13JV16) MELSEC-Q CC-Link IE Field Network Master/Local Module User's Manual Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network master/local module. SH-080917ENG (13JZ47) MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local module. SH-080917ENG (13JZ47) MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local module. SH-080972ENG (13JZ54) Q Corresponding MELSECNET/H Network System Reference Manual (PLC to PLC network) Explains the specifications, settings and procedures before operation, parameter setting, programming, and SH-080049 |
| Implanted to CC-Link and CC-Link system master/local modules. (13JZ41) (13JZ41) (13JZ41) CC-Link IE Controller Network Reference Manual SH-080668ENG Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the CC-Link IE Controller Network. SH-080668ENG MELSEC-Q CC-Link IE Field Network Master/Local Module User's Manual SH-080917ENG Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network Master/Local Module User's Manual SH-080917ENG MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual SH-080917ENG Mellesec-L CC-Link IE Field Network Master/Local Module User's Manual SH-080917ENG Mellesec-L CC-Link IE Field Network Master/Local Module User's Manual SH-080972ENG Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local SH-080972ENG Module. (Sold separately) (13JZ47) Oddiesecomming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local (13JZ47) Module. (Sold separately) (13JZ47) Q Corresponding MELSECNET/H Network System Refere |
| (Sold separately)CC-Link IE Controller Network Reference Manual Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the CC-Link IE Controller Network.SH-080668ENG (13JV16)MELSEC-Q CC-Link IE Field Network Master/Local Module User's Manual Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network master/local module.SH-080917ENG (13JZ47)MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local module.SH-080917ENG (13JZ47)MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local (13JZ54)SH-080972ENG (13JZ54)Q Corresponding MELSECNET/H Network System Reference Manual (PLC to PLC network) Explains the specifications, settings and procedures before operation, parameter setting, programming, and SH-080049SH-080049 |
| CC-Link IE Controller Network Reference Manual SH-080668ENG Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the SH-080668ENG CC-Link IE Controller Network. (Sold separately) MELSEC-Q CC-Link IE Field Network Master/Local Module User's Manual SH-080917ENG Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local SH-080917ENG MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual SH-080917ENG (13JZ47) MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual SH-080917ENG (13JZ47) MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual SH-080917ENG (13JZ47) Methodule. (Sold separately) (13JZ47) Methodule. (Sold separately) (13JZ47) Methodule. (Sold separately) (13JZ54) Module. (Sold separately) (13JZ54) Module. (Sold separately) (13JZ54) Q Corresponding MELSECNET/H Network System Reference Manual (PLC to PLC network) SH-080049 Explains the specifications, settings and procedures before operation, parameter setting, programming, |
| Explains the system configuration, performance specifications, functions, handling, wiring, and troubleshooting of the SH-080668ENG CC-Link IE Controller Network. (Sold separately) MELSEC-Q CC-Link IE Field Network Master/Local Module User's Manual SH-080917ENG Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local (13JZ47) SH-080917ENG MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual (Sold separately) MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual SH-080972ENG Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local (13JZ47) Meture (Sold separately) Q Corresponding Meture (Sold separately) Q Corresponding Meture (Sold separately) Q C |
| CC-Link IE Controller Network. (13JV16) MELSEC-Q CC-Link IE Field Network Master/Local Module User's Manual SH-080917ENG Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local (13JZ47) SH-080917ENG MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual (Sold separately) MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual SH-080972ENG Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local (13JZ54) SH-080972ENG Module. (Sold separately) (13JZ54) Q Corresponding MELSECNET/H Network System Reference Manual (PLC to PLC network) SH-080049 Explains the specifications, settings and procedures before operation, parameter setting, programming, and SH-080049 |
| (Sold separately)MELSEC-Q CC-Link IE Field Network Master/Local Module User's ManualExplains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local (Sold separately)SH-080917ENG (13JZ47)MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local module.SH-080972ENG (13JZ54)Q Corresponding MELSECNET/H Network System Reference Manual (PLC to PLC network) Explains the specifications, settings and procedures before operation, parameter setting, programming, andSH-080049 |
| MELSEC-Q CC-Link IE Field Network Master/Local Module User's Manual SH-080917ENG Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local (13JZ47) SH-080917ENG MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual (Sold separately) MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual SH-080972ENG Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local (13JZ54) SH-080972ENG Module. (Sold separately) (13JZ54) Q Corresponding MELSECNET/H Network System Reference Manual (PLC to PLC network) SH-080049 Explains the specifications, settings and procedures before operation, parameter setting, programming, and SH-080049 |
| Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local (13JZ47) SH-080917ENG (13JZ47) Mell SEC-L CC-Link IE Field Network Master/Local Module User's Manual (Sold separately) Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local (13JZ54) SH-080972ENG (13JZ54) Module. (Sold separately) (13JZ54) Q Corresponding MELSECNET/H Network System Reference Manual (PLC to PLC network) SH-080049 |
| programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local (13JZ47) (13JZ47) module. (Sold separately) MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual SH-080972ENG Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local (13JZ54) SH-080972ENG Q Corresponding MELSECNET/H Network System Reference Manual (PLC to PLC network) SH-080049 Explains the specifications, settings and procedures before operation, parameter setting, programming, and SH-080049 |
| module.(Sold separately)MELSEC-L CC-Link IE Field Network Master/Local Module User's ManualExplains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local (Sold separately)SH-080972ENG (13JZ54)Q Corresponding MELSECNET/H Network System Reference Manual (PLC to PLC network)SH-080049Explains the specifications, settings and procedures before operation, parameter setting, programming, andSH-080049 |
| MELSEC-L CC-Link IE Field Network Master/Local Module User's Manual SH-080972ENG Explains the specifications, procedures before operation, system configuration, installation, settings, functions, programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local (13JZ54) SH-080972ENG Meter of the control of the |
| Explains the specifications, procedures before operation, system configuration, installation, settings, functions, SH-080972ENG programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local (13JZ54) module. (Sold separately) Q Corresponding MELSECNET/H Network System Reference Manual (PLC to PLC network) SH-080049 Explains the specifications, settings and procedures before operation, parameter setting, programming, and SH-080049 |
| programming, and troubleshooting of the CC-Link IE Field Network and the CC-Link IE Field Network master/local (13JZ54) (13JZ54) module. (Sold separately) Q Corresponding MELSECNET/H Network System Reference Manual (PLC to PLC network) Explains the specifications, settings and procedures before operation, parameter setting, programming, and SH-080049 |
| module. (Sold separately) Q Corresponding MELSECNET/H Network System Reference Manual (PLC to PLC network) SH-080049 Explains the specifications, settings and procedures before operation, parameter setting, programming, and SH-080049 |
| Q Corresponding MELSECNET/H Network System Reference Manual (PLC to PLC network) Explains the specifications, settings and procedures before operation, parameter setting, programming, and SH-080049 |
| Explains the specifications, settings and procedures before operation, parameter setting, programming, and SH-080049 |
| |
| troubleshooting of the MELSECNET/H PLC-to-PLC network system. (13JF92) |
| |
| (Sold separately) |
| Q Corresponding MELSECNET/H Network System Reference Manual (Remote I/O network) |
| Explains the system configuration, performance specifications, and programming of the MELSECNET/H network SH-080124 |
| system (remote I/O network). (13JF96) |
| (Sold separately) |
| Q Corresponding Ethernet Interface Module User's Manual (Basic) |
| Explains the specifications of the Ethernet module, data communication procedure with external devices, line SH-080009 |
| connection (open/close), fixed buffer communication, random access buffer communication, and troubleshooting. (13JL88) |
| (Sold separately) |
| Q Corresponding Ethernet Interface Module User's Manual (Application) |
| Explains the e-mail function of the Ethernet module, programmable controller CPU status monitoring, communication |
| function using the MELSECNET/H or MELSECNET/10 as a relay station, communication with data link instructions. |
| and the use of file transfer (FTP server) function. (13JL89) |
| (Sold separately) |

• PID control instruction

| Manual name | Manual number (Model code) |
|--|-------------------------------|
| MELSEC-Q/L/QnA Programming Manual (PID Control Instructions) Explains the dedicated instructions for PID control. | SH-080040 (13JF59) |
| (Sold separately) | (1551 58) |

Socket communication function instruction

| Manual name | | Manual number (Model code) |
|---|-------------------|-------------------------------|
| QnUCPU User's Manual (Communication via Built-in Ethernet Port) Explains the specifications and functions of the built-in Ethernet port communication. | (Sold separately) | SH-080811 (13JZ29) |
| MELSEC-L CPU Module User's Manual (Built-In Ethernet Function) Explains the specifications and functions of the built-in Ethernet port communication. | (Sold separately) | SH-080891ENG (13JZ37) |

Built-in I/O function instruction

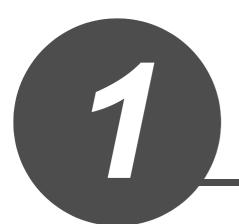
| Manual name | Manual number (Model code) |
|---|-------------------------------|
| MELSEC-L CPU Module User's Manual (Built-In I/O Function) | |
| Explains the general input/output function, interrupt input function, pulse catch function, positioning function, and high- | SH-080892ENG |
| speed counter function of CPU module. | (13JZ38) |
| (Sold separately) | |

Data logging function instruction

| Manual name | Manual number (Model code) |
|---|-------------------------------|
| MELSEC-L CPU Module User's Manual (Data Logging Function) | |
| Explains the specifications of the LCPU module data logging function, and the method for using the LCPU logging configuration tool. | SH-080893ENG (13JZ39) |
| (Sold separately) | |

SFC control Instruction

| Manual name | | Manual number (Model code) |
|---|-------------------|-------------------------------|
| MELSEC-Q/L/QnA Programming Manual (SFC) Explains the programming methods required to create SFC program, specifications and functions. | (Sold separately) | SH-080041ENG (13JF60) |



OVERVIEW

| 1.1 | Purpose of This Manual | 1-2 |
|-----|---|-----|
| 1.2 | Generic Terms and Abbreviations in This Manual | 1-5 |
| 1.3 | Explanation Content in This Manual | 1-6 |
| 1.4 | Modules and Versions Applicable to Instructions | 1-8 |

This manual explains the instructions for the network module, intelligent function module, PID control, socket communication function, built-in I/O function, and data logging function among common instructions and special instructions necessary for creating programs using the structured programming technique.

Manuals for reference are listed in the following table according to their purpose.

For information such as the contents and number of each manual, refer to the list of 'Related manuals'.

| Purpose | | GX Works2 | GX Works2 Beginner's Manual | | GX Works2 Version 1 Operating Manual | | | |
|---------------------------|--|------------------------------|--------------------------------|-----------------------|---|-------------------|-----------------------|-----------------------------------|
| | | Installation Instructions | Simple Project | Structured Project | Common | Simple Project | Structured Project | Intelligent Function Module |
| Installation | Learning the operating environment and installation method | Details | | | | | | |
| | Learning a USB driver installation method | | | | Details | | | |
| | Learning all functions of GX Works2 | | | | Outline | | | |
| Operation of GX Works2 | Learning the project types and available languages in GX Works2 | | | | Outline | | | |
| | Learning the basic operations and operating procedures when creating a simple project for the first time | | Details | | | | | |
| | Learning the basic operations and operating procedures when creating a structured project for the first time | | | Details | | | | |
| | Learning the operations of available functions regardless of project type. | | | | Details | | | |
| | Learning the functions and operation methods for programming | | | | Outline | Details | Details | |
| | Learning data setting methods for intelligent function module | | | | | | | Details |

(1) Operation of GX Works2

1

(2) Operations in each programming language

For details of instructions used in each programming language, refer to the section 3 on the next page

| Purpose | | | orks2 's Manual | GX Works2 Version 1 Operating Manual | | |
|-------------------|---------------------------|-------------------|-----------------------|---|-----------------------|--|
| | | Simple Project | Structured Project | Simple Project | Structured Project | |
| | Ladder | Outline | | Details | | |
| Simple Project | SFC | *1 Outline | | Details | | |
| | ST | | Outline | | Details | |
| | Ladder | Outline | | Details | | |
| Structured | SFC | *1 Outline | | Details | | |
| Project | Structured ladder/ FBD | | Outline | | Details | |
| | ST | | Outline | | Details | |

*1: MELSAP3 and FX series SFC only

| (3) | Details of instructions | in | each | programming | language |
|-----|-------------------------|----|-------|-------------|-----------|
| (0) | Botano or motiono | | 00011 | programming | , angaage |

| Purpose | | MELSEC- Q/L/F Structured Programming Manual | MELSEC-Q/L Structured Programming Manual | | | MELSEC- Q/L MELSEC-Q/L/QnA Programming Manual Manual | | | Manual for module to be used |
|---|--|---|---|-------------------------|--------------------------|---|-----------------------------|---------|------------------------------------|
| | | Fundamentals | Common Instructions | Special Instructions | Application Functions | Common Instructions | PID Control Instructions | SFC | - |
| All languages | Learning details of programmable controller CPU error codes, special relays, and special registers | | | | | Details | | | |
| | Learning the types and details of common instructions | | | | | Details | | | |
| Using ladder | Learning the types and details of instructions for intelligent function modules | | | | | | | | Details |
| language | Learning the types and details of instructions for network modules | | | | | | | | Details |
| | Learning the types and details of instructions for the PID control function | | | | | | Details | | |
| Using SFC language | Learning details of specifications, functions, and instructions of SFC (MELSAP3) | | | | | | | Details | |
| | Learning the fundamentals for creating a structured program | Details | | | | | | | |
| | Learning the types and details of common instructions | | Details | | | | | | |
| Using structured ladder/FBD/ ST language | Learning the types and details of instructions for intelligent function modules | | | Outline | | | | | Details |
| | Learning the types and details of instructions for network modules | | | Outline | | | | | Details |
| | Learning the types and details of instructions for the PID control function | | | Outline | | | Details | | |
| | Learning the types and details of application functions | | | | Details | | | | |

This manual uses the generic terms and abbreviations listed in the following table to discuss the software packages and programmable controller CPUs. Corresponding module models are also listed if needed.

| Generic term and abbreviation | Description |
|----------------------------------|--|
| GX Works2 | Generic product name for SWnDNC-GXW2-E (n: version) |
| Basic model QCPU | Generic term for Q00J, Q00, and Q01 |
| High Performance model QCPU | Generic term for Q02, Q02H, Q06H, Q12H, and Q25H |
| Universal model QCPU | Generic term for Q00UJ, Q00U, Q01U, Q02U, Q03UD, Q03UDE, Q04UDH, Q04UDEH, Q06UDH, Q06UDEH, Q10UDEH, Q13UDH, Q13UDEH, Q20UDEH, Q20UDEH, Q26UDH, Q26UDEH, Q26U |
| | O50UDEH, and Q100UDEH |
| Built-in Ethernet port | Generic term for Q03UDE, Q04UDEH, Q06UDEH, Q10UDEH, Q13UDEH, Q20UDEH, Q26UDEH, |
| QCPU | O50UDEH, and Q100UDEH |
| QCPU (Q mode) | Generic term for Basic model QCPU, High Performance model QCPU, and Universal model QCPU |
| LCPU | Generic term for L02, L02-P, L26-BT and L26-PBT |
| CPU module | Generic term for QCPU (Q mode) and LCPU |
| CC-Link IE Controller | Abbreviation for CC-Link IE Controller Network system |
| Network | |
| CC-Link IE Field Network | Abbreviation for CC-Link IE Field Network system |
| CC-Link IE | Generic term for CC-Link IE Controller Network system and CC-Link IE Field Network system |
| MELSECNET/H | Abbreviation for MELSECNET/H network system |
| Ethernet | Abbreviation for Ethernet network system |
| CC-Link | Abbreviation for Control & Communication Link |
| Personal computer | Generic term for personal computer on which Windows [®] operates |
| | Generic term for the sequence instructions, basic instructions, application instructions, data link |
| Common instruction | instructions, multiple CPU dedicated instructions, and multiple CPU high-speed transmission dedicated instructions |
| Special instruction | Generic term for module dedicated instructions, PID control instructions, socket communication function instructions, built-in I/O function instructions, and data logging function instructions |

This manual explains the programming methods and data used for control of the following modules and PID control using structured programming technique.

| Function/module for explaining an instruction | Processing performed by the instruction | Reference |
|--|--|-------------|
| Analog module | Switches the mode. (Offset/gain setting mode or normal mode) Reads the user range setting offset/gain value. Restores the user range setting offset/gain value. | Section 5.1 |
| Positioning module | Restores the absolute position of the specified axis. Starts positioning of the specified axis. Executes teaching of the specified axis. Writes parameters/positioning data and block start data to a flash ROM. Initializes setting data. | Section 5.2 |
| Serial communication module | Sends and receives data to and from an external device.Registers and reads user frames. | Section 5.3 |
| CC-Link system master/local module | Reads and writes data from and to an intelligent device station on the CC-Link system. Reads and writes data from and to the auto-refresh buffer memory at the master station. Sets the network parameters. | |
| CC-Link IE network module | Sends and receives data to and from an external device. Reads and writes data from and to another station on the | Section 5.4 |
| MELSECNET/H network module | CC-Link IE or MELSECNET/H network system. • Reads and clears error information. | |
| Ethernet interface module | Sends and receives e-mails. | |
| PID control instruction | Sets PID control data and performs PID operation for inexact differential and exact differential. Stops and starts operation of the specified loop. Changes the parameter of the specified loop. | Chapter 6 |
| Socket communication function (Built-in Ethernet port QCPU, LCPU) | Opens/closes a connection. Reads receive data. Changes the receive mode. | Chapter 7 |

| Function/module for explaining an instruction | | Processing performed by the instruction | Reference | |
|--|----------------------|--|------------|--|
| | Positioning function | Starts positioning of the specified axis. Starts OPR of the specified axis. Starts JOG operation of the specified axis. Restores the absolute position of the specified axis. Stops the operating axis. Changes the speed and the target position of the specified axis. | | |
| Built-in I/O function | Counter function | Updates the current value of the specified CH. Sets a ring counter lower limit value and a ring counter upper limit value. Sets a preset value/latch counter value/sampling counter value. Sets the coincidence output No. n point. Measures the frequency/rotation speed. Stores the measured pulse value. Outputs the PWM wave form. | Chapter 8 | |
| Data logging function | | Generates a trigger on the data logging of the specified data logging configuration number. Resets the LOGTRG instruction of the specified data logging configuration number. | Chapter 9 | |
| SFC control | | Reads comment of an active step in the specified SFC block. Reads comment of transition condition associated with an active step in the specified SFC block. | Chapter 10 | |

• Precautions on using instructions

For details of the specifications, functions, and operating timing of each instruction, refer to the related manuals of each module.

🖙 'MANUALS'

This section explains the modules and versions applicable to the instructions explained in this manual.

| Function/module for explaining | an instruction | Applicable version/serial number |
|--------------------------------------|--------------------|--|
| | Q64AD | |
| | Q68ADV | |
| | Q68ADI | |
| | Q64AD-GH | |
| | Q62AD-DGH | |
| | Q68AD-G | |
| | Q66AD-DG | |
| | Q62DAN Q64DAN | |
| | Q64DAN Q68DAVN | |
| | Q68DAVN Q68DAIN | |
| | Q62DA | |
| | Q64DA | |
| Analog module | Q68DAV | Applicable to all versions |
| | Q68DAI | |
| | Q62DA-FG | |
| | Q66DA-G | |
| | Q64RD | |
| | Q64RD-G | |
| | Q64TD | |
| | Q64TDV-GH | |
| | Q68TD-G-H01 | |
| | Q68TD-G-H02 | |
| | Q68RD3-G | |
| | Q61LD | |
| | L60AD4 | |
| | L60DA4 | |
| | QD75P1 | |
| | QD75P2 | |
| | QD75P4 | |
| | QD75D1 | |
| | QD75D2 | |
| | QD75D4 | |
| Positioning module | | Applicable to all versions |
| | QD75M1 | |
| | QD75M2 | |
| | QD75M4 | |
| | QD75MH1 | |
| | QD75MH2 | |
| | QD75MH4 | |
| | QJ71C24N | |
| | QJ71C24N-R2 | |
| | QJ71C24N-R4 | The modules that can use the UINI instruction are |
| Serial communication module | QJ71C24 | limited. |
| | QJ71C24-R2 | For details F Section 5.3.14 |
| | LJ71C24 | |
| | LJ71C24-R2 | |
| | QJ61BT11N | Applicable to all versions |
| | LJ61BT11 | |
| | | The modules that can use the RLPASET instruction are |
| | | limited. |
| CC-Link system master/local module | | The instruction is applicable to the module of which the |
| | QJ61BT11 | function version is B and the first five digits of the serial |
| | | - |
| | | number are '03042' or higher. |
| | | For details F Section 5.4.7 |
| CC-Link IE Controller Network module | QJ71GP21-SX | Applicable to all versions |
| | QJ71GP21S-SX | - The second sec |

For details of applicable versions, refer to each instruction in Chapter 5.

| Function/module for explainin | - | Applicable version/serial number |
|---------------------------------------|----------------------------|--|
| CC-Link IE Field Network module | QJ71GF11-T2 LJ71GF11-T2 | Applicable to all versions |
| | QJ71LP21 | |
| | QJ71LP21-25 | |
| | QJ71LP21S-25 | |
| MELSECNET/H network module | QJ71LP21G | Applicable to all versions |
| MELSECINE I/H Hetwork Houdule | QJ71BR11 | Applicable to all versions |
| | QJ72LP25-25 | |
| | QJ72LP25G | |
| | QJ72BR15 QJ71E71-100 | |
| Ethernet interface module | QJ71E71-100 QJ71E71-B5 | Applicable to all versions |
| | QJ71E71-B2 | |
| | Q00J | |
| | Q00UJ | |
| | Q00 | |
| | Q00U | |
| | Q01 | |
| | Q01U | |
| | | |
| | Q02 | |
| | Q02H | |
| | Q02U | |
| | Q03UD | |
| | Q03UDE | |
| | Q04UDH | |
| | Q04UDEH | |
| | Q06H | |
| | Q06UDH | |
| CPU module supporting the PID control | Q06UDEH | The modules that can use the instruction are limited. |
| nstruction | Q10UDH | For details F Section 6.1, Section 6.2 |
| | Q10UDEH | |
| | Q12H | |
| | Q13UDH | |
| | Q13UDEH | |
| | Q20UDH | |
| | | |
| | Q20UDEH | |
| | Q25H | |
| | Q26UDH | |
| | Q26UDEH | |
| | Q50UDEH | |
| | Q100UDEH | |
| | L02 | |
| | L02-P | |
| | L26-BT | |
| | L26-PBT | |
| | Q03UDE | |
| | Q04UDEH | |
| | Q06UDEH | |
| | Q10UDEH | The modules that can use the socket communication |
| | Q13UDEH | function instruction are limited when using the Built-in |
| | Q20UDEH | Ethernet port QCPU. |
| Built-in Ethernet port QCPU/LCPU | Q26UDEH | The instruction is applicable to the module of which the |
| (Built-in Ethernet function) | Q50UDEH | function version is B and the first five digits of the seria |
| | Q100UDEH | number are '11012' or higher. |
| | L02 | |
| | | Applicable to all versions for LCPU. |
| | L02-P | |
| | L26-BT | |
| | L26-PBT | I |

| Function/module for explaining | | Applicable version/serial number |
|--------------------------------|----------|---|
| | L02 | |
| LCPU (Built-in I/O function) | L02-P | Applicable to all versions |
| | L26-BT | |
| | L26-PBT | |
| | L02 | |
| LCPU (Data logging function) | L02-P | Applicable to all versions |
| | L26-BT | |
| | L26-PBT | |
| | Q00J | |
| | Q00UJ | |
| | Q00 | |
| | Q00U | |
| | Q01 | |
| | Q01U | |
| | Q02 | |
| | Q02H | |
| | Q02U | |
| | Q03UD | |
| | Q03UDE | |
| | Q04UDH | |
| | Q04UDEH | |
| CPU module supporting the SFC | Q06H | The modules that can use the instruction are limited. |
| instruction | Q06UDH | For details F Section 10.1 |
| | Q06UDEH | |
| | Q10UDH | |
| | Q10UDEH | |
| | Q12H | |
| | Q13UDH | |
| | Q13UDEH | |
| | Q20UDH | |
| | Q20UDEH | |
| | Q25H | |
| | Q26UDH | |
| | Q26UDEH | |
| | Q50UDEH | |
| | Q100UDEH | |
| | | I |

| How to check the applicable ve | rsion or serial number |
|--|--|
| Intelligent function modules | : User's Manual or Reference Manual for |
| | the module listed in 'Manuals' |
| CPU modules supporting PID c | ontrol: User's Manual (Function Explanation, |
| | Program Fundamentals) of the CPU |
| | module to be used |
| Built-in Ethernet port QCPU | : QnUCPU User's Manual |
| | (Communication via Built-in Ethernet |
| | Port) |
| Manual for reference | |
| S 'MANUALS' | |



INSTRUCTION TABLES

| 2.1 | How to Read Instruction Tables 2 | -2 |
|-----|--|----|
| 2.2 | Module Dedicated Instruction | -3 |
| 2.3 | PID Control Instruction | 10 |
| 2.4 | Socket Communication Function Instruction 2- | 11 |
| 2.5 | Built-in I/O Function Instruction | 12 |
| 2.6 | Data Logging Function Instruction | 15 |
| 2.7 | SFC Control Instruction | 15 |

| Classification | Instruction name | Argument | Processing details | Executing condition | Applicable module | Page |
|---------------------------------------|------------------|--------------------|-------------------------------------|---------------------|----------------------|------|
| On-demand function transmission | G_ONDEMAND | Un), \$1, \$2, d | Sends data using the on-demand | | Serial | 5.64 |
| | GP_ONDEMAND | Un), 61, 62, d | function of MC protocol. | | | 5-64 |
| Nonprocedural | G_OUTPUT | (Ln), §1, §2, d | Sends the specified number of data. | | | 5-68 |
| protocol | GP_OUTPUT | (m), (s), (s2, (d) | Sends the specified number of data. | | Serial | 5-08 |
| communication | G_INPUT | Un, s, d), d2 | Reads the received data. | | | 5-71 |
| ↑ ① | | 3 | (4) | † 5 | ↑ 6 | |

Instruction tables in Section 2.2 have the following form:

Description

- ① Classifies instructions by application.
- ② Indicates the instructions used in a program.
- ③ Indicates the arguments of the instruction.
 - (s), (d): Source...... Stores data before operation.
 - (a), (b): Destination Indicates the destination of data after operation.
 - n, n1: Specifies the number of devices and the number of transfers.
 - (in): Specifies the network number.
- ④ Indicates the processing details of each instruction.

(5) Details of executing condition of each instruction are as follows:

| Symbol | Executing condition |
|--------|---|
| | Indicates an 'executed while ON' type instruction that is executed only while the precondition is ON. When the precondition is OFF, the instruction is not executed and does not perform processing. |
| | Indicates an 'executed once at ON' type instruction that is executed only at the rising pulse (OFF \rightarrow ON) of the precondition of the instruction. The instruction is not executed afterwards even when the condition is ON and thus does not perform processing. |

(6) Indicates the execution target module of each instruction.

For details of the icons, refer to Chapter 4.

O Indicates the pages on which the instructions are explained.

2.2.1 Analog instruction

| Classification | Instruction name | Argument | Processing details | Executing condition | Page |
|---------------------------|------------------|----------------|---|---------------------|-------|
| Mode switching | G_OFFGAN | Un), s | Moves to the offset/gain setting mode. Moves to the normal mode. | | 5-2 |
| | GP_OFFGAN | Un), (s) | | | |
| Setting value | G_OGLOAD | (Un), (s), (d) | Reads the user range settings offset/gain value to the programmable controller CPU. | | - 5-4 |
| reading | GP_OGLOAD | Un), (s), (d) | | | |
| Setting value restoration | G_OGSTOR | (Ln), (s), (d) | Restores the user range settings offset/gain value stored in the programmable controller CPU. | | 5-27 |
| | GP_OGSTOR | (m), (s), (d) | | | 5-27 |

2.2.2 Positioning instruction

| Classification | Instruction name | Argument | Processing details | Executing condition | Page |
|-----------------------------|------------------|----------------|--|---------------------|------|
| | Z_ABRST1 | (Un), (s), (d) | | | |
| Absolute position | Z_ABRST2 | Un), (s), (d) | Restores the absolute position of the specified axis. | | 5-51 |
| restoration | Z_ABRST3 | (Un), (s), (d) | | | 0-01 |
| | Z_ABRST4 | (Un), (s), (d) | | | |
| Positioning start | ZP_PSTRT1 | (Un), (s), (d) | | | |
| | ZP_PSTRT2 | (Un), (S), (d) | Starts positioning of the specified axis. | | 5-55 |
| | ZP_PSTRT3 | (un), (s), (d) | | | 5-55 |
| | ZP_PSTRT4 | (Un), (s), (d) | | | |
| | ZP_TEACH1 | (Un), (s), (d) | | | |
| Teaching | ZP_TEACH2 | (Un), (s), (d) | Performs teaching for the specified axis. | | 5-57 |
| readining | ZP_TEACH3 | (Un), (s), (d) | | | 0.01 |
| | ZP_TEACH4 | Un), (s), (d) | | | |
| Writing to flash ROM | ZP_PFWRT | un), (s), (d) | Writes the QD75 parameters, positioning data, and block start data to the flash ROM. | | 5-60 |
| Setting data initialization | ZP_PINIT | (m), (s), (d) | Initializes the QD75 setting data. | | 5-62 |

2.2.3 Serial communication

| Classification | Instruction name | Argument | Processing details | Executing condition | Applicable module | Page |
|---|------------------|-----------------------|---|---------------------|----------------------|-------|
| On-demand | G_ONDEMAND | (m), s1, s2, d | Sends data using the on-demand | | | |
| function transmission | GP_ONDEMAND | un), 61, 62, 0 | function of MC protocol. | Serial | Serial | 5-64 |
| Nonprocedural | G_OUTPUT | Un, s1, s2, d | | | | 5.00 |
| protocol | GP_OUTPUT | Un, 1, 1, 2, 0 | Sends the specified number of data. | | Serial | 5-68 |
| communication | G_INPUT | Un, s, d), @ | Reads the received data. | | | 5-71 |
| | G_BIDOUT | (m), (s1), (s2), (d) | Sends the specified number of data. | | | 5-74 |
| Bidirectional protocol | GP_BIDOUT | Un, s1, s2, d | Serius the specified humber of data. | | Contal | 5-74 |
| communication | G_BIDIN | (m), (s), (d), (d) | Reads the received data. | | Serial | 5-77 |
| | GP_BIDIN | (Un), (S), (d), (d2) | Reads the received data. | | | 5-11 |
| Communication | G_SPBUSY | Un, d | Reads the data transmission/ | | Covial | 5-79 |
| status check | GP_SPBUSY | Un, d | reception status using the instruction. | | Serial | 5-79 |
| Receive data clear | ZP_CSET | (m), s), 2, (1), (2) | Clears receive data without stopping transmission using the nonprocedural protocol. | | Serial | 5-80 |
| Data | Z_BUFRCVS | (un), (s), (d) | Receives data with an interrupt program using the nonprocedural protocol or bidirectional protocol. | | | 5-83 |
| transmission/ reception | G_PRR | (Un), (S), (d) | Sends data by user frame according to the specification in user frame | | Serial | |
| | GP_PRR | (un), (s), (d) | specification area for transmission using the nonprocedural protocol. | | | 5-85 |
| Initial setting | ZP_CSET | un), s), ©, t), t2 | Sets the unit (word/byte) of the number of the data to be sent or received. | | Serial | 5-89 |
| Programmable controller CPU monitor | ZP_CSET | ☞, ᢒ, ֎, ๗, ֎ | Registers and cancels the programmable controller CPU monitoring for using the programmable controller CPU monitoring function. | | Serial | 5-93 |
| | G_PUTE | (m), (s1), (s2), (d) | Registers a user frames to the flash | | | 5-101 |
| Flash ROM user frame | GP_PUTE | (m), (s1), (s2), (d) | ROM. | | Contal | 5-101 |
| registration/ reading | G_GETE | Un, 1, 2, 0 | Reads a user frames from the flash | | Serial | 5 404 |
| 3 | GP_GETE | Un, 1, 2, 0 | ROM. | | | 5-104 |
| Mode switching | ZP_UINI | (un), (s), (d) | Switches the mode, transmission specification, and host station number. | | Serial | 5-107 |
| Pre-defined | G_CPRTCL | (m), n1, n2, (s), (d) | Executes the protocols and | | | E 440 |
| protocol communication | GP_CPRTCL | un), n1, n2, s, d | functional protocols written to the flash ROM. | | Serial | 5-112 |

STRUCTION ABLES **2**

2.2.4 Network dedicated instruction

| Classification | Instruction name | Argument | Processing details | Executing condition | Applicable module | Page |
|--|------------------|----------------------------------|---|---------------------|---|-------|
| | J_RIRD | (m [°]), (s), (d), (d2 | | | CC IE C | |
| Reading from the | JP_RIRD | £, \$, #, @ | | | CC IE C CC IE F | |
| buffer memory of an intelligent device station | G_RIRD | un, s, d), @ | Reads data for the specified number of points from the buffer memory or device of the specified station. | | CC-Link CC IE C | 5-115 |
| | GP_RIRD | ur, s, t, æ | | | CC-Link CC IE C CC IE F | |
| Writing to the buffer memory of an intelligent device station | J_RIWT | (m [°]), (s), (d), (d) | | | CC IE C | |
| | JP_RIWT | ☞, ७, ๗, ๗ | | | CC IE C CC IE F | |
| | G_RIWT | un, s), ©, 0 | Writes data for the specified number of points to the buffer memory or device of the specified station. | | CC-Link CC IE C CC-Link CC IE C CC IE F | 5-120 |
| | GP_RIWT | ☞, ☞, ֎, @ | | | | |
| Reading from the buffer memory of | G_RIRCV | (m), s1, s2, d1, d2 | Automatically performs handshaking with the specified station and reads | | CC-Link | |
| an intelligent device station (with handshake) | GP_RIRCV | ₩, 9, 2, 0, @ | data from the buffer memory of the specified station. This instruction is applicable with a module having a handshake signal, such as the AJ65BT-R2(N). | | | 5-125 |
| Writing to the buffer memory of | G_RISEND | ☞, ᢒ, 2, 1, 2 | Automatically performs handshaking with the specified station and writes data to the buffer memory of the | | | |
| an intelligent device station (with handshake) | GP_RISEND | ☞, ᢒ, ୧, ๗, ๗ | specified station. This instruction is applicable with a module having a handshake signal, such as the AJ65BT-R2(N). | | CC-Link | 5-129 |
| Reading from the | G_RIFR | un), n1, n2, n3, d | Reads data from the auto-refresh buffer memory of the specified | | | |
| Reading from the auto-refresh buffer memory of the master station | GP_RIFR | ௴, n1, n2, n3, | station | CC-Link | 5-133 | |
| Writing to the | G_RITO | un), n1, n2, n3, @ | Writes data to the auto-refresh buffer memory of the specified station. | | | |
| auto-refresh buffer memory of the master station | GP_RITO | ሆ), n1, n2, n3, | This instruction is applicable with a module having an auto-refresh buffer, such as the AJ65BT-R2. | | CC-Link | 5-135 |

| Classification | Instruction name | Argument | Processing details | Executing condition | Applicable module | Page | |
|-------------------------|------------------|--|---|---------------------|----------------------|-------|-----------------------|
| Network parameter | G_RLPASET | (m ²), (1), (12), (3), (4), (13), (10), (| Sets network parameter to the master station and starts up the data | | CC-Link | 5-137 | |
| setting | GP_RLPASET | (Ln), (1), (2), (3), (4), (5), (d) | link. | | CC-LINK | 5-157 | |
| | J_READ | Jn), 61, 62, 61, 62 | | | | | 2 |
| | JP_READ | (m), (s), (s2, (d), (d2 | Reads data from a word device of | | | | z |
| | G_READ | Un), \$1, \$2, 61, 62 | another station. | | | 5-144 | INSTRUCTION TABLES |
| | GP_READ | Un), 61, 62, 61, 62 | | | | | NSTR |
| | J_SREAD | Jn), 61, 62, 61, 62, 63 | | | 1 | | |
| Device data read/ | JP_SREAD | (In [°]), (s ¹), (s ²), (d ¹), (d ²), (d ³) | Reads data from a device of another | | | 5 450 | |
| | G_SREAD | Un, \$1, \$2, \$1, \$2, \$3 | station (with completion device). | | CC IE C | 5-150 | |
| | GP_SREAD | (m), s1, s2, d1, d2, d3 | | | CC IE F | | |
| write | J_WRITE | (In [°]), (s1), (s2), (s3), (d1) | | | NET/H Ether | | |
| - | JP_WRITE | (In [®]), (s1), (s2), (s3), (d1) | Writes data to a device of another station. | | | | |
| | G_WRITE | Un), 61, 62, 63, 61 | | | - | 5-154 | |
| | GP_WRITE | (Un), (s1, (s2, (s3, (d1 | | | - | | |
| | J_SWRITE | (In [°]), (s ¹), (s ²), (d ¹), (d ²), (d ³) | | | | | |
| | JP_SWRITE | (In [°]), (s1, (s2, (d1, (d2, (d3 | Writes data to a device of another | | | 5-161 | |
| | G_SWRITE | Un), \$1, \$2, \$1, \$2, \$3 | station (with completion device). | | | | |
| | GP_SWRITE | Un, s1, s2, d1, d2, d3 | - | | | | |
| | J_SEND | Jn, \$1, \$2, d | | | | | |
| | JP_SEND | Jn), \$1, \$2, d | | | - | | |
| | G_SEND | Un, \$1, \$2, d | Sends data to another station. | | - | 5-165 | |
| | GP_SEND | Un, 11, 12, 0 | - | | CC IE C | | |
| Message (user-specified | J_RECV | Jn), (6), (1), (2) | | | CC IE F | | |
| data) communication | JP_RECV | Jn [*]), (s), (d), (2) | Reads received data from another | | NET/H | 5-173 | |
| | G_RECV | Un, 6, 10, 12 | station (for main program). | | Ether | | |
| | GP_RECV | Un), (6), (1), (2) | | | | | |
| | Z_RECVS | Un), (1), (2), (d) | Reads received data from another station (for interrupt program). | | | 5-178 | |

| Classification | Instruction name | Argument | Processing details | Executing condition | Applicable module | Page |
|---|--|--|---|---------------------|--------------------------------------|-------|
| | J_REQ | Jn), \$1, \$2, \$1, \$2 | | | CC IE C | |
| Transient request | JP_REQ | ₲,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,, | Executes remote RUN/STOP for another station. | | CC IE C CC IE F NET/H Ether | 5-181 |
| to another station | G_REQ | Un), s), s), d), a | Reads/writes clock data from another station. | | CC IE C | 5-161 |
| | GP_REQ | ☞, ᢒ, ֎, ๗, ֎ | | | CC IE C CC IE F NET/H Ether | |
| Read from other | J_ZNRD | (Ln), n1, (S), n2, (d), (d) | ent Processing details condition modul 9, @ | | 5-190 | |
| Read from other station devices Write to other station devices Remote RUN | JP_ZNRD | un, n1, s, n2, d), @ | | | | 5-190 |
| Write to other | J_ZNWR | un, n1, s, n2, d, d | | | | 5-193 |
| station devices | JP_ZNWR | Un), n1, (s), n2, (d), (d) | | | | 5-195 |
| | Z_RRUN_J | (m [°]), (s ¹), (s ²), (s ³), (s ⁴), (d) | | | CC IE C NET/H | |
| Remote | ZP_RRUN_J | (m [°]), (s1), (s2), (s3), (s4), (d) | | | | 5-197 |
| Remote RUN | Z_RRUN_U | Un, \$1, \$2, \$3, \$4, d | | | | 5-197 |
| | ZP_RRUN_U | Un, s1, s2, s3, s4, d | | | | |
| | Z_RSTOP_J | (m [°]), (s1), (s2), (s3), (s4), (d) | | | | |
| Remote | ZP_RSTOP_J | Jn, \$1, \$2, \$3, \$4, d | Executes remote STOP for a CPU | | CC IE C | 5-200 |
| STOP | GP_REQ GP_REQ GP_REQ GP_REQ GP_REQ GP_REQ Image: Second | NET/H | 5-200 | | | |
| | ZP_RSTOP_U | Un, \$1, \$2, \$3, \$4, d | | | | |
| | Z_RTMRD_J | Jn), \$1, \$2, \$3, @, @ | | | | |
| Reading clock data from another | ZP_RTMRD_J | Jn ² , s ¹ , s ² , s ³ , d ¹ , d ² | Reads clock data from a CPU | | CC IE C | 5-203 |
| station | Z_RTMRD_U | Un), s1, s2, s3, d1, d2 | module on another station. | | NET/H | 5-203 |
| | ZP_RTMRD_U | un), s1, s2, s3, d1, d2 | | | | |
| | Z_RTMWR_J | Jn), s1, s2, s3, s4, d | | | | |
| Writing clock data | ZP_RTMWR_J | Jn, S1, S2, S3, S4, d | Writes clock data to a CPU module | | CC IE C | 5 205 |
| to another station | Z_RTMWR_U | Un), 61, 62, 63, 64, 0 | on another station. | | NET/H | 5-205 |
| | ZP_RTMWR_U | (m ²), (s1), (s2), (s3), (s4), (d) | | | | |

| Classification | Instruction name | Argument | Processing details | Executing condition | Applicable module | Page |
|--|------------------|--|--|---------------------|----------------------|-------|
| Reading from buffer memory of intelligent function | Z_REMFR | (m), n1, n2, n3, n4, n5, (d), @ | Reads data from the buffer memory of an intelligent function module on the remote I/O station. | | NET/H | 5-208 |
| module on remote I/O station | ZP_REMFR | اله), n1, n2, n3, n4, n5, 1), @ | | | CC IE F NET/H | |
| Writing to buffer memory of intelligent function | Z_REMTO | ஸ், n1, n2, n3, n4, n5, ரி, @ | Writes data to the buffer memory of an intelligent function module on the | | NET/H | 5-210 |
| module on remote I/O station | ZP_REMTO | (m), n1, n2, n3, n4, n5, (d) , @ | remote I/O station. | | CC IE F NET/H | |
| Setting parameter | G_CCPASET | ☞, ᢒ, ֎, ֍, ଖ, d | master/local modules CC I | CC IE F | 5-212 | |
| | GP_CCPASET | (m [°]), s ¹), s ²), s ³), s ⁴), d | | | | |
| Connection opening | ZP_OPEN | Un [°] , 61, 62, 0 | Opens a connection. | | Ether | 5-218 |
| or closing | ZP_CLOSE | (m), 61, 62, 0 | Closes a connection. | | Luier | 5-222 |
| | ZP_BUFRCV | un), s), s2, d), d2 | Reads received data. (for main program) | | | 5-225 |
| Fixed buffer communication | Z_BUFRCVS | (m), (s), (d) | Reads received data. (for interrupt program) | | Ether | 5-229 |
| | ZP_BUFSND | Un), \$1, \$2, \$3, d | Sends data. | | | 5-231 |
| Reading or clearing | ZP_ERRCLR | Un), (s), (d) | Clears error information. | | | 5-235 |
| error information | ZP_ERRRD | (Ln ²), (S), (d) | Reads error information. | | Ether | 5-238 |
| Re-initialization/ station number setting/changing switch setting | Z_UINI | (m), (s), (d) | Executes re-initialization. | | CC IE C | |
| | ZP_UINI | Un, (s), (d) | Sets the host station number.Changes the switch setting. | | CC IE C Ether | 5-241 |
| E-mail | ZP_MRECV | Un), (s), (d), (d2) | Reads received e-mail. | | | 5-245 |
| communication | ZP_MSEND | (m), s1, s2, d | Sends an e-mail. | | Ether | 5-250 |

2.3.1 PID control instruction (inexact differential)

| Classification | Instruction name | Argument | Processing details | Executing condition | Page |
|---------------------|------------------|----------|---|---------------------|------|
| Data setting | S_PIDINIT | 6 | Sets data to be used for PID operation. | | 6-2 |
| Data Setting | SP_PIDINIT | S | | | 0-2 |
| PID operation | S_PIDCONT | 6 | Performs PID operation based on the set value (SV) | | 6-7 |
| | SP_PIDCONT | 6 | and process value (PV). | | 0-1 |
| PID operation stop | S_PIDSTOP | n | Stops the PID operation for the specified loop | | |
| | SP_PIDSTOP | n | number. | | 6-11 |
| PID operation start | S_PIDRUN | n | Starts the PID operation for the specified loop | | 0 11 |
| TID operation start | SP_PIDRUN | n | number. | | |
| Operation | S_PIDPRMW | n, | Changes operation parameter of the specified loop number. | | 6-12 |
| parameter change | SP_PIDPRMW | n, © | | | 0-12 |

2.3.2 PID control instruction (exact differential)

| Classification | Instruction name | Argument | Processing details | Executing condition | Page |
|---------------------|------------------|----------|---|---------------------|------|
| Data setting | PIDINIT | 6 | Sets data to be used for PID operation. | | 6-16 |
| Data setting | PIDINITP | s | | | 0-10 |
| PID operation | PIDCONT | 6 | Performs PID operation based on the set value (SV) | | 6-21 |
| | PIDCONTP | (\$) | and process value (PV). | | 0-21 |
| PID operation stop | PIDSTOP | n | Stops the PID operation for the specified loop | | |
| | PIDSTOPP | n | number. | | 6-26 |
| PID operation start | PIDRUN | n | Starts the PID operation for the specified loop | | 0-20 |
| | PIDRUNP | n | number. | | |
| Operation | PIDPRMW | n, © | Changes operation parameter of the specified loop number. | | 6-27 |
| parameter change | PIDPRMWP | n, ® | | | 0-27 |

2.4 Socket Communication Function Instruction

| Classification | Instruction name | Argument | Processing details | Executing condition | Page |
|--------------------------------------|------------------|--|--|---------------------|------|
| Opening/closing | SP_SOCOPEN | (Ln), (s1), (s2), (d) | Establishes a connection. | | 7-2 |
| connection | SP_SOCCLOSE | (Ln), (s1), (s2), (d) | Shuts a connection off. | | 7-5 |
| Reading receive | SP_SOCRCV | (11), \$1, \$2, €1, €2 | Reads receive data. (Reading at the end process) | | 7-8 |
| data | S_SOCRCVS | (m), (s), (d) | Reads receive data. (Reading at the instruction execution) | | 7-11 |
| Sending data | SP_SOCSND | (Ln [°]), (S), (S), (G), (G) | Sends data. | | 7-13 |
| Reading connection information | SP_SOCCINF | ur), (1), (2), (1) | Reads connection information. | | 7-16 |
| Changing destination | SP_SOCCSET | (m ²), (s1), (s2) | Changes a destination of a UDP/IP connection. | | 7-19 |
| Changing receive mode | SP_SOCRMODE | (m), (s1), (s2) | Changes the receive mode of a connection. | | 7-22 |
| Reading data from receive data area | S_SOCRDATA | un), @, @, n, @ | Reads data from the receive data area. | | 7-24 |
| | SP_SOCRDATA | ₩,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,, | | | 1 24 |

2.5.1 Positioning function dedicated instruction

| Classification | Instruction name | Argument | Processing details | Executing condition | Page |
|-------------------|------------------|----------|---|---------------------|------|
| | IPPSTRT1 | n | | | |
| | IPPSTRT1P | n | Specifies a data number to be executed from "Positioning Data" No. 1 to No. 10 which are | | 8-2 |
| | IPPSTRT2 | n | previously set in GX Works2, and starts the positioning. | | 0-2 |
| | IPPSTRT2P | n | | | |
| Positioning start | IPDSTRT1 | s | Regardless of "Positioning Data" No. 1 to No. 10 | | |
| Positioning start | IPDSTRT1P | s | which are previously set in GX Works2, starts the positioning using the data stored in the | | 8-3 |
| | IPDSTRT2 | s | devices starting from the one specified for | | 0-3 |
| | IPDSTRT2P | s | control data. | | |
| | IPSIMUL | n1, n2 | Starts the positioning of the axis 1 "Positioning | | 9 |
| | IPSIMULP | n1, n2 | Data" number and the axis 2 "Positioning Data" number simultaneously. | | 8-6 |
| | IPOPR1 (s) | | | | |
| OPR start | IPOPR1P | s | Specifies a method and starts the OPR of the | | 8-7 |
| | IPOPR2 | s | specified axis. | | |
| | IPOPR2P | s | | | |
| JOG start | IPJOG1 | 6), 62 | Starts the JOG operation of the specified axis. | | 8-9 |
| JOG start | IPJOG2 | 6), 62 | Starts the JOG operation of the specified axis. | | |
| Absolute position | IPABRST1 | (s), (d) | Executes the absolute position restoration of the | | 8-11 |
| restoration | IPABRST2 | (s), (d) | specified axis. | | 0-11 |
| Stop | IPSTOP1 | - | Stops the axis in operation. | | 8-13 |
| Stop | IPSTOP2 | - | | | 0-13 |
| | IPSPCHG1 | s | | | |
| Speed change | IPSPCHG1P | s | Changes the speed of the specified axis | | 8-14 |
| Speed change | IPSPCHG2 | s | Changes the speed of the specified axis. | | |
| | IPSPCHG2P | s | | | |
| | IPTPCHG1 | s | | | |
| Target position | IPTPCHG1P | s | Changes the target position of the specified | | 8-16 |
| change | IPTPCHG2 | s | axis. | | 0-10 |
| | IPTPCHG2P | s | | | |

2.5.2 Counter function dedicated instruction

| Classification | Instruction name | Argument | Processing details | Executing condition | Page | |
|-----------------------------------|------------------|----------|---|---------------------|--------|--|
| | ICCNTRD1 | - | | | | |
| | ICCNTRD1P | - | Stores the most recent value for the current | | 0.40 | |
| Current value read | ICCNTRD2 | - | value of the specified CH. | | 8-18 | |
| | ICCNTRD2P | - | | | | |
| | ICRNGWR1 | 61,62 | | | | |
| Ring counter upper/lower limit | ICRNGWR1P | 61, 62 | Sets a ring counter lower limit value and upper | | 8-19 | |
| value write | ICRNGWR2 | st, s2 | limit value of the specified CH. | | 0-19 | |
| | ICRNGWR2P | €1, €2 | | | | |
| | ICPREWR1 | (s) | | | | |
| Preset value write | ICPREWR1P | s | Sets a preset value of the specified CH. | | 8-21 | |
| Fleset value white | ICPREWR2 | s | Sets a preser value of the specified Ch. | | 0-21 | |
| | ICOREWR2P | s | | | | |
| | ICLTHRD1 | n, @ | | | | |
| Latch counter | ICLTHRD1P | n, @ | Stores a latch counter value of the specified CH. | | 8-22 | |
| value read | ICLTHRD2 | n, @ | | | | |
| | ICLTHRD2P | n, @ | | | | |
| | ICSMPRD1 | d | | | - 8-23 | |
| Sampling counter | ICSMPRD1P | d | Stores a sampling counter value of the specified | | | |
| value read | ICSMPRD2 | d | СН. | | | |
| | ICSMPRD2P | d | | | | |
| | ICCOVWR1 | n, | | | | |
| Coincidence | ICCOVWR1P | n, | Sets a coincidence output No. n point of the | | 8-24 | |
| output point write | ICCOVWR2 | n, ® | specified CH. | | 0-24 | |
| | ICCOVWR2P | n, | | | | |
| Frequency | ICFCNT1 | d | Measures the frequency of the specified CH. | | 8-25 | |
| measurement | ICFCNT2 | d | | | 8-25 | |
| Rotation speed | ICRCNT1 | d | Measures the rotation speed of the specified | | 8-26 | |
| measurement | ICRCNT2 | d | СН. | | - 8-26 | |

| Classification | Instruction name | Argument | Processing details | Executing condition | Page |
|------------------|------------------|----------|--|---------------------|------|
| | ICPLSRD1 | đ | | | |
| Pulse | ICPLSRD1P | đ | Stores the measured pulse value of the specified CH. | | 8-27 |
| measurement read | ICPLSRD2 | đ | | | |
| | ICPLSRD2P | Ø | | | |
| PWM output | ICPWM1 | \$1, \$2 | Outputs the PWM waveform of the specified CH. | | 8-28 |
| | ICPWM2 | s), s2 | | | 0-20 |

2.6 Data Logging Function Instruction

| Classification | Instruction name | Argument | Processing details | Executing condition | Page |
|------------------------------|------------------|----------|---|---------------------|------|
| Trigger logging set/reset | LOGTRG | n | Generates the trigger conditions in a trigger logging. Stores the data sampling results to the data logging file for the number of times specified in the trigger logging configuration of the programming tool. | | 9-2 |
| | LOGTRGR | n | Resets the trigger conditions | | |

2.7 SFC Control Instruction

| Classification | Instruction name | Argument | Processing details | Executing condition | Page |
|---|------------------|-------------------|--|---------------------|------|
| SFC step comment read | S_SFCSCOMR | n1, n2, n3, ⓓ), @ | Reads comment of an active step in the specified SFC block by the specified number. | | 10-2 |
| | SP_SFCSCOMR | n1, n2, n3, ⓓ , @ | | | |
| SFC transition condition comment read | S_SFCTCOMR | n1, n2, n3, @, @ | Reads comment of transition condition associated with an active step in the specified SFC block by the specified number. | | 10-4 |
| | SP_SFCTCOMR | n1, n2, n3, ⓓ , @ | | | |

MEMO



CONFIGURATION OF INSTRUCTIONS

| 3.1 | onfiguration of Instructions | 5-2 |
|-----|------------------------------|-----|
| 0.1 | | 1 |

1

Instructions available in the CPU module can be divided into an instruction name and an argument.

The application of an instruction name and an argument are as follows:

- Instruction name..... Indicates the function of the instruction.
- Argument Indicates the I/O data used in the instruction.

Arguments are classified into I/O number, source data, destination data, number of devices, executing condition, and execution result.

- (1) I/O number
 - (a) I/O number is data that set a module in which the instruction is to be executed. Set the I/O number by start I/O number or a network number of the module depending on the instruction.
 - (b) Setting the start I/O number (Un) of the module
 Set the higher two digits when expressing the start I/O number in three digits for the module in which the instruction is to be executed.
 Set the start I/O number in a numeric value or character string according to the data type available with the instruction.
 - Setting the start I/O number in word (unsigned)/16-bit string or word (signed) data type

Set the start I/O number of the module for 'n' of 'Un'. Example: For the module whose start I/O number is 020H: 02

 Setting the start I/O number in string data type Set the start I/O number in the format of "Un" (n: start I/O number of the module).
 Example: For the module whose start I/O number is 020H: "02"

(c) Network number (Jn) setting

Set the network number of the network module/Ethernet module in which the instruction is to be executed.

Set a network number indicated below, in word (unsigned)/16-bit string or word (signed) data type, for 'n' of 'Jn'.

- 1 to 239 : Network number
- 254 : Network specified in "Valid module during other station access" on the GX Works2 network parameter screen

Example: When the network number is 1:1

(2) Source (s)

•

- (a) A source is data used in an operation.
- (b) The following source types are available depending on the device specified in an instruction:

| Constant | Specifies a numeric value used in an operation. Constants are set during programming so that they cannot be changed while the program is being executed. Perform index setting when using them as variable data. |
|----------------------------|---|
| Bit device and word device | Specifies the device in which the data used in the operation are stored. Data must be stored to the specified device before executing the operation. By changing the data to be stored to the specified device while a program is being executed, the data used in the instruction can be changed. |

- (c) The instructions explained in this manual use special data. Refer to the explanation for each instruction and use data correctly.
- (3) Destination d
 - (a) Data after the operation are stored to a destination.
 - (b) Set a device in which data are to be stored to a destination.
 - (c) The instructions explained in this manual use special data. Refer to the explanation for each instruction and use data correctly.

For details of the configuration of instructions for labels and structures, refer to MELSEC-Q/L/F Structured Programming Manual (Fundamentals).

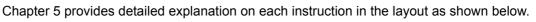
MEMO

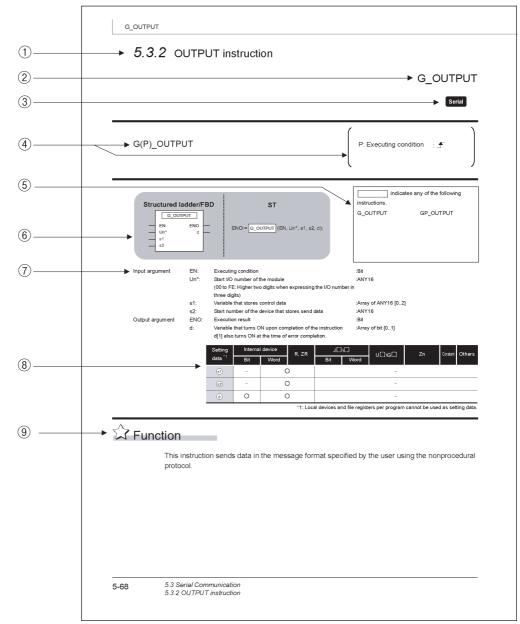


HOW TO READ INSTRUCTIONS



1





- ① Indicates a section number and an outline of an instruction.
- 2 Indicates an instruction to be explained.

③ Indicates the instruction execution target module.

If one instruction is to be executed in two or more modules, applicable modules are indicated using icons.

| Module | lcon |
|---|---------------------|
| Serial communication | Serial |
| CC-Link IE Controller Network | CC IE C |
| CC-Link IE Field Network | CC IE F |
| MELSECNET/H | NET/H |
| Ethernet | Ether |
| Universal model QCPU (Built-in Ethernet port QCPU) | QnUDE(H) |
| LCPU | L CPU |
| Universal model QCPU | Universal |
| High Performance model QCPU | High performance |

④ Indicates the instruction name and executing condition of the instruction.

| Executing condition | Non-conditional execution | Executed while ON | Executed once at ON | Executed while OFF | Executed once at OFF |
|-----------------------------------|------------------------------|-------------------|---------------------|--------------------|-------------------------|
| Symbols on the corresponding page | No symbol | | | | |

- (5) Indicates the instruction names that can be described.
- ⑥ Indicates the description format of the instruction in the structured ladder/FBD/ST language.
- ⑦ Indicates the names of input and output arguments, and the data type of each argument. For details of each data type, refer to MELSEC-Q/L/F structured programming manual (Fundamentals).
- (8) Devices that can be used in the instruction are marked with \bigcirc .

The following table shows applicable classification for usable devices.

| Device classification | Internal device (system, user) | | File register | | Link direct device ^{*4} | | Intelligent function module | | Others ^{*5} |
|--------------------------------|---|---|------------------|-----------------------------------|----------------------------------|---------|-----------------------------------|--------------|---|
| classification | Bit | Word | R, ZR | Bit | Word | U[]\G[] | Zn | | |
| Usable device ^{*1} | X, Y, M, L, SM, F, B, SB, FX, FY ^{*2} | T, ST, C, ^{*3} D, W, SD, SW, FD, @⊡ | R, ZR | J [] //SB] [] //Y] [] //X | J⊞/SM]∭/M | U[]\G[] | Z | K, H, E, \$, | P, I, J, U, DX, DY, N, BL, TR, BL\S, V |

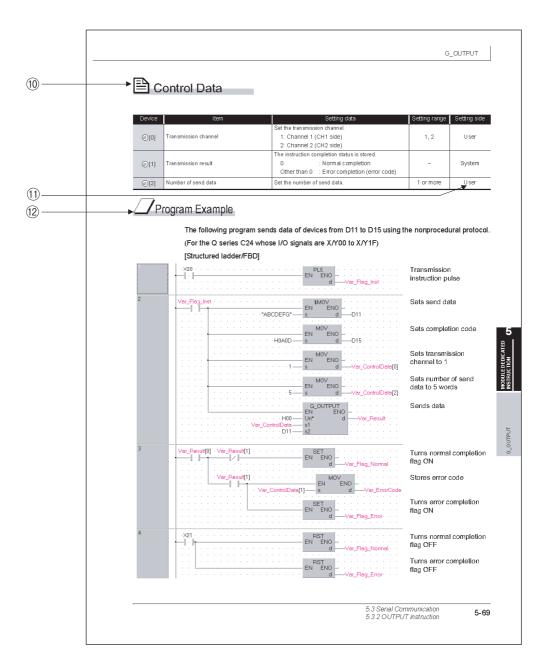
*1 : For description of each device, refer to the User's Manual (Function Explanation, Program Fundamentals) of the CPU module to be used.

*2 : FX and FY can be used in bit data only, and FD can be used in word data only in the PID control instruction.

*3 : T, ST, and C can be used in word data only (cannot be used in bit data).

- *4 : These devices can be used in CC-Link IE, MELSECNET/H, and MELSECNET/10.
- *5 : The Constant and Others columns describe settable devices.

(9) Indicates the processing performed by the instruction.



1 Indicates data such as control data, send data or receive data, that are used for an input argument or output argument in an instruction.

Example: Control data to be used in the CC-Link instruction 'GP_RIRD'

1 The setting side indicates the following:

User : Data set by user before dedicated instruction execution

System : Data stored by the programmable controller CPU after dedicated instruction execution

The setting does not need to be set by the user.

If the setting is set by the user, data cannot be read normally.

2 Indicates the program examples of structured ladder/FBD/ST.

The program examples are when the conditions are satisfied. The program example shown above shows that the conditions are satisfied in ladder block number 3.

The processing when the conditions are not satisfied, create appropriate programs as necessary.



MODULE DEDICATED INSTRUCTION

| 5.1 | Analog Instruction | 5-2 |
|-----|-------------------------------|--------|
| 5.2 | Positioning Instruction | . 5-51 |
| 5.3 | Serial Communication | . 5-64 |
| 5.4 | Network Dedicated Instruction | 5-115 |

1

5.1 Analog Instruction

5.1.1 OFFGAN instruction

G_OFFGAN

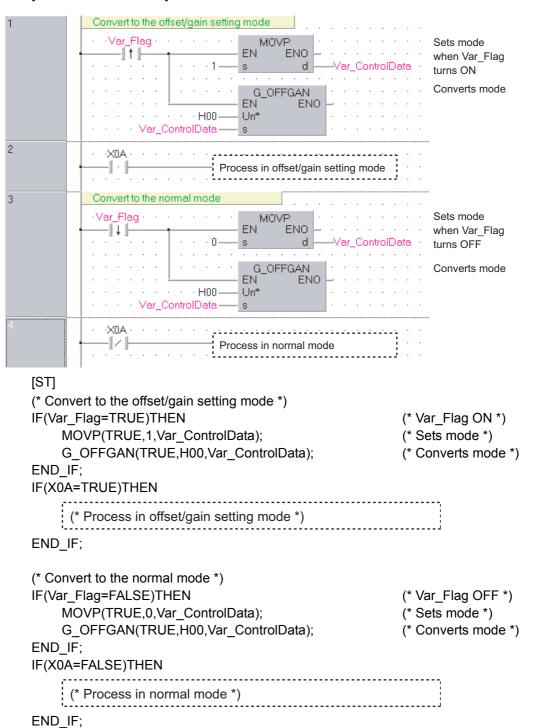
| G(P)_OFFG | AN | | P: Executing condition : |
|--------------------------------------|-------------|---|-------------------------------------|
| Structured G_OF EN Un* s | | D ST ENO:= <u>G_OFFGAN</u> (EN, Un*, s); | instructions. G_OFFGAN GP_OFFGAN |
| Input argument | EN: Un*: | Executing condition Start I/O number of the module (00 to FE: Higher two digits when expressing the I/O | :Bit :ANY16 O number in |
| | S: | three digits) Mode switching 0: To normal mode 1: To offset/gain setting mode | :ANY16 |
| Output argument | ENO: | Execution result | :Bit |
| | Setting | Internal device | |
| | data *1 | Bit Word R, ZR Bit W | Jiii\Giii Zn Constant Othe |



This instruction converts the mode of analog modules. (normal mode to offset/gain setting mode, offset/gain setting mode to normal mode)

Program Example

The following program converts the mode of the A/D converter module mounted on the I/O numbers from X/Y00 to X/Y0F to the offset/gain setting mode when Var_Flag turns ON, and gets it back to the normal mode when Var_Flag turns OFF.



[Structured ladder/FBD]

5.1.2 OGLOAD instruction

G_OGLOAD

| G(P)_OGLO | AD | | P: Executing condition : |
|---|------|--|--|
| Structured I G_OGI EN Un* s | | BD ST ENO:= G_OGLOAD (EN, Un*, s, d); | indicates any of the following instructions. G_OGLOAD GP_OGLOAD |
| Input argument | EN: | Executing condition | :Bit |
| | Un*: | Start I/O number of the module (00 to FE: Higher two digits when expressing the I/O number three digits) | :ANY16 r in |
| | S: | Variable that stores control data | :Array of ANY16 [035] |
| Output argument | ENO: | Execution result | :Bit |
| | d: | Variable that turns ON upon completion of the instruction d[1] also turns ON at the time of error completion. | :Array of bit [01] |
| | | Setting data *1 Internal device R, ZR Jiii/iiii Bit Word Bit Bit | UIII\GIII Zn Constant Other |
| | | b | - |
| | • | *1: Local devices and f | ile registers per program cannot be used as setting da |

Grant Function

This instruction reads the user range settings offset/gain values of analog modules to the CPU.

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|---------------|---|--|-------------------|--------------|
| s [0] | System area | - | - | - |
| ঙ[1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| ঙ [2] | Pass data classification setting | Specify the voltage/current of the offset/gain values to be read. 0: Voltage specified 1: Current specified b15 b4 b3 b2 b1 b0 0 to 0 CH4 CH3 CH2 CH1 | 0000н to 000Fн | User |
| s [3] | System area | - | - | - |
| ঙ [4] | CH1 Industrial shipment settings offset value | _ | - | System |
| s [5] | CH1 Industrial shipment settings gain value | _ | - | System |
| ⑤[6] | CH2 Industrial shipment settings offset value | _ | - | System |
| ⑤ [7] | CH2 Industrial shipment settings gain value | _ | - | System |
| §[8] | CH3 Industrial shipment settings offset value | _ | - | System |
| ⑤[9] | CH3 Industrial shipment settings gain value | _ | - | System |
| s [10] | CH4 Industrial shipment settings offset value | _ | - | System |
| s[11] | CH4 Industrial shipment settings gain value | _ | - | System |
| s [12] | CH1 User range settings offset value | _ | - | System |
| <u></u> জ[13] | CH1 User range settings gain value | _ | - | System |
| <u>জ</u> [14] | CH2 User range settings offset value | _ | - | System |
| <u></u> জ[15] | CH2 User range settings gain value | _ | _ | System |
| s [16] | CH3 User range settings offset value | _ | - | System |
| s [17] | CH3 User range settings gain value | _ | - | System |
| s [18] | CH4 User range settings offset value | _ | - | System |
| s [19] | CH4 User range settings gain value | _ | _ | System |

(1) Q64AD/Q64DAN/Q64DA/L60AD4/L60DA4 *1

*1 : Set the data only to the Pass data classification setting \circledast [2].

(2) Q68ADV/Q68ADI/Q68DAVN/Q68DAV/Q68DAIN/Q68DAI *1

| Device | Item | Setting data | Setting range | Setting side |
|----------------|---|--|---------------|--------------|
| s [0] | System area | _ | - | _ |
| 011 | | The instruction completion status is stored. | | |
| <u></u> \$[1] | Completion status | 0 : Normal completion | - | System |
| \$[2] | | Other than 0 : Error completion (error code) | | |
| ⑤[2] ⑤[3] | System area | - | - | - |
| ⑤[4] | CH1 Industrial shipment settings offset value | | _ | System |
| ⑤ [5] | CH1 Industrial shipment settings gain value | | - | System |
| <u> </u> | CH2 Industrial shipment settings offset value | _ | _ | System |
| <u></u> (§[7] | CH2 Industrial shipment settings gain value | _ | _ | System |
| <u>\$</u> [8] | CH3 Industrial shipment settings offset value | _ | - | System |
| s [9] | CH3 Industrial shipment settings gain value | _ | - | System |
| s [10] | CH4 Industrial shipment settings offset value | _ | - | System |
| §[11] | CH4 Industrial shipment settings gain value | _ | - | System |
| s [12] | CH5 Industrial shipment settings offset value | _ | - | System |
| s[13] | CH5 Industrial shipment settings gain value | _ | - | System |
| s [14] | CH6 Industrial shipment settings offset value | _ | - | System |
| <u>জ</u> [15] | CH6 Industrial shipment settings gain value | _ | - | System |
| s[16] | CH7 Industrial shipment settings offset value | _ | - | System |
| জ [17] | CH7 Industrial shipment settings gain value | _ | - | System |
| <u></u> \$[18] | CH8 Industrial shipment settings offset value | _ | - | System |
| s[19] | CH8 Industrial shipment settings gain value | - | - | System |
| s [20] | CH1 User range settings offset value | - | - | System |
| s[21] | CH1 User range settings gain value | - | - | System |
| s [22] | CH2 User range settings offset value | _ | - | System |
| s [23] | CH2 User range settings gain value | _ | - | System |
| s[24] | CH3 User range settings offset value | _ | - | System |
| s [25] | CH3 User range settings gain value | _ | - | System |
| s [26] | CH4 User range settings offset value | _ | - | System |
| \$[27] | CH4 User range settings gain value | _ | - | System |
| s [28] | CH5 User range settings offset value | _ | - | System |
| s [29] | CH5 User range settings gain value | - | - | System |
| s [30] | CH6 User range settings offset value | _ | - | System |
| S[31] | CH6 User range settings gain value | _ | - | System |
| s [32] | CH7 User range settings offset value | _ | - | System |
| \$[33] | CH7 User range settings gain value | _ | - | System |
| \$[34] | CH8 User range settings offset value | _ | - | System |
| s [35] | CH8 User range settings gain value | _ | - | System |

*1 : Setting is unnecessary. If setting is configured, offset/gain setting value is not read properly.

(3) Q64AD-GH^{*1}

| Device | Item | Setting data | Setting range | Setting side |
|---------------|---|--|---------------|--------------|
| ⑤ [0] | System area | - | _ | - |
| | | The instruction completion status is stored. | | |
| s[1] | Completion status | 0 : Normal completion | - | System |
| | | Other than 0 : Error completion (error code) Specify the voltage/current of the offset/gain values to | | |
| | | be read. | | |
| ⑤ [2] | Pass data classification setting | 0: Voltage specified | 0000н to | User |
| ©[2] | | 1: Current specified b15 b4 b3 b2 b1 b0 | 000Fн | |
| | | 0 to 0 CH4 CH3 CH2 CH1 | | |
| \$[3] | System area | _ | _ | _ |
| \$[4] | CH1 Industrial shipment settings offset value (L) | | | |
| \$[5] | CH1 Industrial shipment settings offset value (H) | - | - | System |
| \$[6] | CH1 Industrial shipment settings gain value (L) | | | |
| ⑤ [7] | CH1 Industrial shipment settings gain value (H) | - | - | System |
| \$[8] | CH2 Industrial shipment settings offset value (L) | | | 0 |
| ⑤ [9] | CH2 Industrial shipment settings offset value (H) | - | - | System |
| <u>©</u> [10] | CH2 Industrial shipment settings gain value (L) | | | Custom |
| ⑤ [11] | CH2 Industrial shipment settings gain value (H) | - | - | System |
| S [12] | CH3 Industrial shipment settings offset value (L) | | | Custom |
| <u></u> জ[13] | CH3 Industrial shipment settings offset value (H) | - | - | System |
| s [14] | CH3 Industrial shipment settings gain value (L) | | | System |
| s [15] | CH3 Industrial shipment settings gain value (H) | _ | - | System |
| <u>ি</u> [16] | CH4 Industrial shipment settings offset value (L) | | | System |
| S [17] | CH4 Industrial shipment settings offset value (H) | _ | _ | Oystern |
| ⑤ [18] | CH4 Industrial shipment settings gain value (L) | _ | _ | System |
| s [19] | CH4 Industrial shipment settings gain value (H) | _ | | Oystern |
| s [20] | CH1 User range settings offset value (L) | _ | _ | System |
| s[21] | CH1 User range settings offset value (H) | _ | _ | Oystern |
| s [22] | CH1 User range settings gain value (L) | _ | _ | System |
| \$ [23] | CH1 User range settings gain value (H) | | | - Jotom |
| \$ [24] | CH2 User range settings offset value (L) | | _ | System |
| \$ [25] | CH2 User range settings offset value (H) | | | - , |
| \$ [26] | CH2 User range settings gain value (L) | _ | | System |
| S [27] | CH2 User range settings gain value (H) | | | - , |
| s [28] | CH3 User range settings offset value (L) | _ | - | System |
| s [29] | CH3 User range settings offset value (H) | | | , |
| s [30] | CH3 User range settings gain value (L) | _ | _ | System |
| s [31] | CH3 User range settings gain value (H) | | | |
| s [32] | CH4 User range settings offset value (L) | _ | _ | System |
| s [33] | CH4 User range settings offset value (H) | | | , |
| s [34] | CH4 User range settings gain value (L) | _ | _ | System |
| s [35] | CH4 User range settings gain value (H) | | | |

*1 : Set the data only to the Pass data classification setting [2].

When the data is written to the area to be set by system, offset/gain setting value is not read properly.

G_OGLOAD

5-7

(4) Q62AD-DGH^{*1}

| Device | Item | Setting data | Setting range | Setting side |
|----------------|---|---|---------------|--------------|
| s [0] | System area | - | - | - |
| | | The instruction completion status is stored. | | Quatant |
| <u></u> জ[1] | Completion status | 0 : Normal completion Other than 0 : Error completion (error code) | - | System |
| s[2] | | | | |
| <u> </u> | System area | - | - | - |
| § [4] | CH1 Industrial shipment settings offset value (L) | | | |
| s [5] | CH1 Industrial shipment settings offset value (H) | - | - | System |
| §[6] | CH1 Industrial shipment settings gain value (L) | | | Quatant |
| ⑤ [7] | CH1 Industrial shipment settings gain value (H) | - | - | System |
| § [8] | CH2 Industrial shipment settings offset value (L) | | | Quatara |
| s [9] | CH2 Industrial shipment settings offset value (H) | - | - | System |
| s [10] | CH2 Industrial shipment settings gain value (L) | | | Sustam |
| s[11] | CH2 Industrial shipment settings gain value (H) | | - | System |
| ⑤[12] | | | | |
| to | System area | - | - | - |
| ঙ [19] | | | | |
| s [20] | CH1 User range settings offset value (L) | _ | _ | System |
| ি [21] | CH1 User range settings offset value (H) | | | |
| s [22] | CH1 User range settings gain value (L) | _ | _ | System |
| ঙ [23] | CH1 User range settings gain value (H) | | | |
| ঙ [24] | CH2 User range settings offset value (L) | _ | _ | System |
| <u></u> জ [25] | CH2 User range settings offset value (H) | | | , |
| ঙ [26] | CH2 User range settings gain value (L) | _ | _ | System |
| § [27] | CH2 User range settings gain value (H) | | | - , |
| ি [28] | | | | |
| to | System area | - | - | - |
| s [35] | | | | |

*1 : Setting is unnecessary. If setting is configured, offset/gain setting value is not read properly.

(5) Q68AD-G^{*1}

| Device | Item | Setting data | Setting range | Setting side |
|---------------|---|---|---------------|--------------|
| ⑤ [0] | System area | - | - | - |
| | | The instruction completion status is stored. | | |
| ি [1] | Completion status | 0 :Normal completion | - | System |
| | | Other than 0 :Error completion (error code) Specify the voltage/current of the offset/gain values to | | |
| | | be read. | | |
| | Deep data place if action actions | 0: Voltage specified | 0000н to | lloor |
| s[2] | Pass data classification setting | 1: Current specified | 00FFн | User |
| | | b15 b8 b7 b6 b5 b4 b3 b2 b1 b0 0 to 0 CH8CH7CH6CH5CH4CH3CH2CH1 | | |
| \$[3] | System area | _ | - | - |
| ⑤[4] | CH1 Industrial shipment settings offset value | _ | - | System |
| \$[5] | CH1 Industrial shipment settings gain value | _ | - | System |
| ⑤[6] | CH2 Industrial shipment settings offset value | _ | - | System |
| \$[7] | CH2 Industrial shipment settings gain value | - | - | System |
| ⑤[8] | CH3 Industrial shipment settings offset value | _ | - | System |
| ⑤[9] | CH3 Industrial shipment settings gain value | _ | - | System |
| s [10] | CH4 Industrial shipment settings offset value | _ | - | System |
| <u></u> জ[11] | CH4 Industrial shipment settings gain value | _ | - | System |
| s [12] | CH5 Industrial shipment settings offset value | _ | - | System |
| s [13] | CH5 Industrial shipment settings gain value | _ | _ | System |
| s [14] | CH6 Industrial shipment settings offset value | _ | - | System |
| s [15] | CH6 Industrial shipment settings gain value | _ | - | System |
| s [16] | CH7 Industrial shipment settings offset value | _ | - | System |
| s[17] | CH7 Industrial shipment settings gain value | _ | - | System |
| s [18] | CH8 Industrial shipment settings offset value | _ | - | System |
| s [19] | CH8 Industrial shipment settings gain value | _ | - | System |
| s [20] | CH1 User range settings offset value | _ | - | System |
| s[21] | CH1 User range settings gain value | _ | - | System |
| s [22] | CH2 User range settings offset value | _ | - | System |
| s [23] | CH2 User range settings gain value | _ | - | System |
| s [24] | CH3 User range settings offset value | _ | - | System |
| s [25] | CH3 User range settings gain value | - | - | System |
| s [26] | CH4 User range settings offset value | _ | - | System |
| s [27] | CH4 User range settings gain value | _ | - | System |
| \$ [28] | CH5 User range settings offset value | - | - | System |
| s [29] | CH5 User range settings gain value | _ | - | System |
| s [30] | CH6 User range settings offset value | _ | - | System |
| <u> </u> (31] | CH6 User range settings gain value | _ | - | System |
| s [32] | CH7 User range settings offset value | _ | - | System |
| s [33] | CH7 User range settings gain value | _ | - | System |
| s [34] | CH8 User range settings offset value | _ | - | System |
| s [35] | CH8 User range settings gain value | - | - | System |

*1 : Set the data only to the Pass data classification setting \circledast [2].

(6) Q66AD-DG^{*1}

| Device | Item | Setting data | Setting range | Setting side |
|----------------|---|--|---------------|--------------|
| s [0] | System area | _ | _ | _ |
| \$[1] | Completion status | The instruction completion status is stored. 0 :Normal completion | _ | System |
| s [2] | System area | Other than 0 :Error completion (error code) | _ | |
| § [3] | | | | |
| ⑤ [4] | CH1 Industrial shipment settings offset value | - | - | System |
| s [5] | CH1 Industrial shipment settings gain value | - | - | System |
| s [6] | CH2 Industrial shipment settings offset value | - | - | System |
| § [7] | CH2 Industrial shipment settings gain value | _ | - | System |
| ⑤[8] | CH3 Industrial shipment settings offset value | - | - | System |
| s [9] | CH3 Industrial shipment settings gain value | - | - | System |
| S[10] | CH4 Industrial shipment settings offset value | - | - | System |
| \$[11] | CH4 Industrial shipment settings gain value | - | - | System |
| ⑤[12] | CH5 Industrial shipment settings offset value | _ | - | System |
| <u></u> জ[13] | CH5 Industrial shipment settings gain value | _ | - | System |
| <u></u> | CH6 Industrial shipment settings offset value | _ | - | System |
| s [15] | CH6 Industrial shipment settings gain value | _ | - | System |
| s[16] | | | | |
| to | System area | _ | - | - |
| s [19] | | | | |
| s [20] | CH1 User range settings offset value | _ | - | System |
| ⑤ [21] | CH1 User range settings gain value | _ | - | System |
| \$[22] | CH2 User range settings offset value | _ | - | System |
| s [23] | CH2 User range settings gain value | _ | - | System |
| s[24] | CH3 User range settings offset value | _ | - | System |
| s [25] | CH3 User range settings gain value | _ | - | System |
| s[26] | CH4 User range settings offset value | _ | _ | System |
| s[27] | CH4 User range settings gain value | | - | System |
| <u> </u> | CH5 User range settings offset value | | _ | System |
| © [29] | CH5 User range settings gain value | _ | _ | System |
| ⑤[20] | CH6 User range settings offset value | | _ | System |
| ⑤[31] | CH6 User range settings gain value | _ | _ | System |
| ⑤[31] ⑤[32] | | | | |
| to | System area | _ | _ | _ |
| s [35] | | | | |
| 0[00] | | | | |

*1 : Setting is unnecessary. If setting is configured, offset/gain setting value is not read properly.

(7) Q62DAN/Q62DA *1

| Device | Item | Setting data | Setting range | Setting side |
|---------------|---|---|---------------|--------------|
| s [0] | System area | - | - | - |
| | | The instruction completion status is stored. | | |
| ি [1] | Completion status | 0 : Normal completion | - | System |
| | | Other than 0 : Error completion (error code) | | |
| | | Specify the voltage/current of the offset/gain values to be read. | | |
| | | 0: Voltage specified | 0000н to | |
| ঙ[2] | Pass data classification setting | 1: Current specified | 0003н | User |
| | | b15 b2 b1 b0 0 to 0 CH2 CH1 | | |
| s [3] | System area | _ | - | - |
| ⑤ [4] | CH1 Industrial shipment settings offset value | _ | - | System |
| ঙ [5] | CH1 Industrial shipment settings gain value | _ | - | System |
| \$[6] | CH2 Industrial shipment settings offset value | _ | - | System |
| ⑤ [7] | CH2 Industrial shipment settings gain value | _ | - | System |
| ⑤[8] | CH1 User range settings offset value | _ | - | System |
| s [9] | CH1 User range settings gain value | _ | - | System |
| s [10] | CH2 User range settings offset value | - | - | System |
| <u>জ</u> [11] | CH2 User range settings gain value | - | - | System |

*1 : Set the data only to the Pass data classification setting s [2].

(8) Q62DA-FG^{*1}

| Device | Item | Setting data | Setting range | Setting side |
|---------------|---|--|---------------|--------------|
| s [0] | System area | _ | _ | _ |
| <u>ি</u> [1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| ি[2] | Pass data classification setting | Specify the user range setting to read the offset/gain values. Он: User range setting 1 specified 1н: User range setting 2 specified 2н: User range setting 3 specified $\underbrace{\begin{smallmatrix} b15 to b12 & b11 to b8 & b7 to b4 & b3 to b0}_{OH & OH & CH2 & CH1}$ | - | User |
| ⑤ [3] | System area | - | - | - |
| s [4] | CH1 Industrial shipment settings offset value (used for D/A) | - | - | System |
| ঙ [5] | CH1 Industrial shipment setting gain value (used for D/A) | - | - | System |
| S [6] | CH2 Industrial shipment settings offset value (used for D/A) | - | - | System |
| s[7] | CH2 Industrial shipment setting gain value (used for D/A) | - | - | System |
| ি[8] | CH1 Industrial shipment settings offset value (used for monitor output) | - | - | System |
| ঙ [9] | CH1 Industrial shipment settings gain value (used for monitor output) | _ | - | System |
| s [10] | CH2 Industrial shipment settings offset value (used for monitor output) | - | - | System |
| s [11] | CH2 Industrial shipment settings gain value (used for monitor output) | - | - | System |
| s [12] | CH1 User range settings offset value (used for D/A) | _ | - | System |
| <u>ঙ</u> [13] | CH1 User range settings gain value (used for D/A) | - | - | System |
| \$[14] | CH2 User range settings offset value (used for D/A) | - | - | System |
| ঙ [15] | CH2 User range settings gain value (used for D/A) | _ | - | System |
| জ [16] | CH1 User range settings offset value (used for monitor output) | - | - | System |
| ঙ[17] | CH1 User range settings gain value (used for monitor output) | _ | - | System |
| s [18] | CH2 User range settings offset value (used for monitor output) | _ | - | System |
| ঙ [19] | CH2 User range settings gain value (used for monitor output) | _ | - | System |

*1 : Set the data only to the Pass data classification setting s [2].

(9) Q66DA-G^{*1}

| Device | Item | Setting data | Setting range | Setting side |
|----------------|---|--|---------------|--------------|
| s [0] | System area | - | - | - |
| | | The instruction completion status is stored. | | |
| ⑤[1] | Completion status | 0 :Normal completion Other than 0 :Error completion (error code) | - | System |
| | | Specify the user range setting to read the offset/gain | | |
| | | values. | | |
| | | 0H: User range setting 1 specified | | |
| ାର | Pass data classification setting | 1H: User range setting 2 specified 2H: User range setting 3 specified | 0000н to | User |
| ঙ [2] | T ass data classification setting | b15 b12 b11 b10 b9 b8 b7 b6 b5 b4 b3 b2 b1 b0 | 0АААн | 0301 |
| | | | | |
| | | 0 : Fixed CH6 CH5 CH4 CH3 CH2 CH1 | | |
| <u>\$[3]</u> | System area | _ | - | _ |
| ⑤ [4] | CH1 Industrial shipment settings offset value | - | _ | System |
| <u>\$</u> [5] | CH1 Industrial shipment settings gain value | _ | _ | System |
| <u>\$[6]</u> | CH2 Industrial shipment settings offset value | _ | - | System |
| ⑤ [7] | CH2 Industrial shipment settings gain value | _ | - | System |
| ⑤[8] | CH3 Industrial shipment settings offset value | _ | _ | System |
| ⑤ [9] | CH3 Industrial shipment settings gain value | _ | - | System |
| s[10] | CH4 Industrial shipment settings offset value | _ | - | System |
| s[11] | CH4 Industrial shipment settings gain value | _ | - | System |
| s [12] | CH5 Industrial shipment settings offset value | _ | - | System |
| s [13] | CH5 Industrial shipment settings gain value | _ | _ | System |
| s [14] | CH6 Industrial shipment settings offset value | _ | - | System |
| <u></u> জ[15] | CH6 Industrial shipment settings gain value | - | - | System |
| s[16] | CH1 User range settings offset value | - | - | System |
| <u></u> §[17] | CH1 User range settings gain value | _ | - | System |
| s [18] | CH2 User range settings offset value | _ | - | System |
| s [19] | CH2 User range settings gain value | _ | - | System |
| \$ [20] | CH3 User range settings offset value | _ | _ | System |
| <u></u> \$[21] | CH3 User range settings gain value | - | - | System |
| \$[22] | CH4 User range settings offset value | - | _ | System |
| \$[23] | CH4 User range settings gain value | - | _ | System |
| \$[24] | CH5 User range settings offset value | - | _ | System |
| s [25] | CH5 User range settings gain value | - | _ | System |
| s [26] | CH6 User range settings offset value | _ | - | System |
| s [27] | CH6 User range settings gain value | _ | - | System |
| s [28] | | | | |
| to | System area | - | - | System |
| s [35] | | | | |

*1 : Set the data only to the Pass data classification setting s [2].

(10) Q64RD/Q64RD-G *1

Control data of Q64RD/Q64RD-G (1/5)

| Device | | Item | Setting data | Setting range | Setting side |
|---------|-----------------------|--|---|------------------|-----------------|
| S [0] | | System area | - | - | - |
| \$ |)[1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| S | [2] | System area | _ | _ | _ |
| S | [3] | | | | |
| | s [4] | 3-wire CH1 Factory default offset value | - | - | System |
| | s [5] | 3-wire CH1 Factory default offset value | - | - | System |
| | ⑤ [6] | 3-wire CH1 Factory default gain value | - | - | System |
| Q64RD | ⑤ [7] | 3-wire CH1 Factory default gain value | - | - | System |
| | § [8] | 3-wire CH1 User range settings offset value | - | - | System |
| | ⑤ [9] | 3-wire CH1 User range settings offset value | - | - | System |
| | <u></u> জ [10] | 3-wire CH1 User range settings gain value | - | - | System |
| | <u></u> \$[11] | 3-wire CH1 User range settings gain value | - | - | System |
| | ⑤ [4] | 3-wire CH1 Factory default offset value (L) | | | System |
| | ⑤ [5] | 3-wire CH1 Factory default offset value (H) | | | Oystern |
| | § [6] | 3-wire CH1 Factory default gain value (L) | _ | _ | System |
| Q64RD | ⑤ [7] | 3-wire CH1 Factory default gain value (H) | | | Oystern |
| -G | s [8] | 3-wire CH1 User range settings offset value (L) | _ | _ | System |
| | s [9] | 3-wire CH1 User range settings offset value (H) | | | oyotem |
| | <u></u> জ [10] | 3-wire CH1 User range settings gain value (L) | _ | _ | System |
| | <u></u> জ[11] | 3-wire CH1 User range settings gain value (H) | | | Oystern |
| S | [12] | 3-wire CH1 User range settings resistance offset value (L) | _ | _ | System |
| S | [13] | 3-wire CH1 User range settings resistance offset value (H) | | | Oystern |
| S | [14] | 3-wire CH1 User range settings resistance gain value (L) | _ | _ | System |
| \$ [15] | | 3-wire CH1 User range settings resistance gain value (H) | | | Oystern |
| | ঙ [16] | 4-wire CH1 Factory default offset value | - | - | System |
| | <u>ঃ</u> [17] | 4-wire CH1 Factory default offset value | - | - | System |
| Q64RD | <u></u> § [18] | 4-wire CH1 Factory default gain value | - | - | System |
| | s [19] | 4-wire CH1 Factory default gain value | - | - | System |
| QUILD | \$ [20] | 4-wire CH1 User range settings offset value | - | - | System |
| | <u></u> । (হ) [21] | 4-wire CH1 User range settings offset value | - | _ | System |
| | ি [22] | 4-wire CH1 User range settings gain value | - | _ | System |
| | s [23] | 4-wire CH1 User range settings gain value | - | - | System |

| Device | | Item | Setting data | Setting | Setting |
|-------------|------------------|--|--------------|---------|---------|
| | 0[16] | 4-wire CH1 Factory default offset value (L) | | range | side |
| | জ [16] জ [17] | 4-wire CH1 Factory default offset value (L) | - | - | System |
| | ⑤[17] ⑤[18] | 4-wire CH1 Factory default gain value (L) | | | |
| Q64RD | ⑤[10] ⑤[19] | 4-wire CH1 Factory default gain value (H) | - | - | System |
| -G | ⑤[10] ⑤[20] | 4-wire CH1 User range settings offset value (L) | | | |
| | ⑤[20] ⑤[21] | 4-wire CH1 User range settings offset value (H) | _ | - | System |
| | \$[22] | 4-wire CH1 User range settings gain value (L) | | | |
| | \$[23] | 4-wire CH1 User range settings gain value (H) | - | - | System |
| (5) | [24] | 4-wire CH1 User range settings resistance offset value (L) | | | |
| | [24] | 4-wire CH1 User range settings resistance offset value (H) | - | - | System |
| | [26] | 4-wire CH1 User range settings resistance gain value (L) | | | |
| | [20] | 4-wire CH1 User range settings resistance gain value (H) | - | - | System |
| | [27] ⑤[28] | 3-wire CH2 Factory default offset value | _ | _ | System |
| | (20) (20) | 3-wire CH2 Factory default offset value | _ | _ | System |
| | ⑤ [20] | 3-wire CH2 Factory default gain value | _ | _ | System |
| | ⑤[30] ⑤[31] | 3-wire CH2 Factory default gain value | _ | _ | System |
| Q64RD | ⑤[31] ⑤[32] | 3-wire CH2 User range settings offset value | _ | _ | System |
| | _ | 3-wire CH2 User range settings offset value | _ | _ | System |
| | <u> </u> | 3-wire CH2 User range settings gain value | _ | _ | System |
| | ্ | 3-wire CH2 User range settings gain value | _ | _ | System |
| | ⑤[33] ⑤[28] | 3-wire CH2 Factory default offset value (L) | | | Gystern |
| | (20) (20) | 3-wire CH2 Factory default offset value (H) | - | - | System |
| | (3)[29] (3)[30] | 3-wire CH2 Factory default gain value (L) | | | |
| 00400 | | 3-wire CH2 Factory default gain value (H) | _ | - | System |
| Q64RD -G | ত[31] ত[32] | 3-wire CH2 User range settings offset value (L) | | | |
| | ⑤[32] ⑤[33] | 3-wire CH2 User range settings offset value (H) | - | - | System |
| | ⑤[33] ⑤[34] | 3-wire CH2 User range settings gain value (L) | | | |
| | (34) (35) | 3-wire CH2 User range settings gain value (H) | - | - | System |
| | [36] | 3-wire CH2 User range settings resistance offset value (L) | | | |
| | [30] | 3-wire CH2 User range settings resistance offset value (H) | - | - | System |
| | | 3-wire CH2 User range settings resistance gain value (L) | | | |
| <u> </u> | | 3-wire CH2 User range settings resistance gain value (H) | _ | - | System |
| | <u>(</u> (40) | 4-wire CH2 Factory default offset value | _ | _ | System |
| | ⑤[40] ⑤[41] | 4-wire CH2 Factory default offset value | _ | _ | System |
| | S [42] S | 4-wire CH2 Factory default gain value | _ | _ | System |
| | ⑤[42] ⑤[43] | 4-wire CH2 Factory default gain value | _ | _ | System |
| Q64RD | ⑤[43] ⑤[44] | 4-wire CH2 User range settings offset value | _ | _ | System |
| | S [45] | 4-wire CH2 User range settings offset value | _ | _ | System |
| | ⑤[45] ⑤[46] | 4-wire CH2 User range settings gain value | _ | _ | System |
| | S [47] S | 4-wire CH2 User range settings gain value | _ | _ | System |
| | U [4/] | | | | oyotom |

Control data of Q64RD/Q64RD-G (2/5)

| Dev | vice | Item | Setting data | Setting | Setting |
|---------------------------|------------------|--|--------------|---------|---------|
| | | | | range | side |
| | <u>\$</u> [40] | 4-wire CH2 Factory default offset value (L) | - | _ | System |
| | <u></u> জ [41] | 4-wire CH2 Factory default offset value (H) | | | |
| | s [42] | 4-wire CH2 Factory default gain value (L) | - | - | System |
| Q64RD | s [43] | 4-wire CH2 Factory default gain value (H) | | | |
| -G | <u></u> [44] | 4-wire CH2 User range settings offset value (L) | _ | _ | System |
| | ঙ [45] | 4-wire CH2 User range settings offset value (H) | | | |
| | <u></u> জ [46] | 4-wire CH2 User range settings gain value (L) | _ | _ | System |
| | s [47] | 4-wire CH2 User range settings gain value (H) | | | -, |
| S | [48] | 4-wire CH2 User range settings resistance offset value (L) | _ | _ | System |
| S | [49] | 4-wire CH2 User range settings resistance offset value (H) | | | |
| S | [50] | 4-wire CH2 User range settings resistance gain value (L) | _ | _ | System |
| S | [51] | 4-wire CH2 User range settings resistance gain value (H) | | | Oystern |
| | \$ [52] | 3-wire CH3 Factory default offset value | _ | - | System |
| | \$ [53] | 3-wire CH3 Factory default offset value | - | - | System |
| | s [54] | 3-wire CH3 Factory default gain value | - | - | System |
| 00455 | \$ [55] | 3-wire CH3 Factory default gain value | - | - | System |
| Q64RD | § [56] | 3-wire CH3 User range settings offset value | - | - | System |
| | \$[57] | 3-wire CH3 User range settings offset value | - | - | System |
| | § [58] | 3-wire CH3 User range settings gain value | - | - | System |
| | s [59] | 3-wire CH3 User range settings gain value | _ | - | System |
| | s [52] | 3-wire CH3 Factory default offset value (L) | | | |
| | s [53] | 3-wire CH3 Factory default offset value (H) | - | - | System |
| | <u>\$</u> [54] | 3-wire CH3 Factory default gain value (L) | | | |
| Q64RD | <u>\$</u> [55] | 3-wire CH3 Factory default gain value (H) | - | - | System |
| -G | <u>\$</u> [56] | 3-wire CH3 User range settings offset value (L) | | | |
| | <u>\$</u> [57] | 3-wire CH3 User range settings offset value (H) | - | - | System |
| | <u> </u> | 3-wire CH3 User range settings gain value (L) | | | |
| | <u> </u> | 3-wire CH3 User range settings gain value (H) | - | - | System |
| (s) | [60] | 3-wire CH3 User range settings resistance offset value (L) | | | |
| | [61] | 3-wire CH3 User range settings resistance offset value (H) | - | - | System |
| | [62] | 3-wire CH3 User range settings resistance gain value (L) | | | |
| <u> [63]</u> [63] | | 3-wire CH3 User range settings resistance gain value (H) | - | - | System |
| | <u>\$[64]</u> | 4-wire CH3 Factory default offset value | _ | _ | System |
| | ⑤[65] | 4-wire CH3 Factory default offset value | _ | _ | System |
| | § [66] | 4-wire CH3 Factory default gain value | _ | _ | System |
| | ⑤ [67] | 4-wire CH3 Factory default gain value | | | System |
| Q64RD | ⑤[68] | 4-wire CH3 User range settings offset value | | _ | System |
| | ③ [69] ⑤ [69] | 4-wire CH3 User range settings offset value | _ | _ | System |
| | ⑤ [09] ⑤ [70] | 4-wire CH3 User range settings gain value | _ | _ | System |
| | | 4-wire CH3 User range settings gain value | | | System |
| | \$[71] | - with on to osci range settings gain value | - | _ | System |

Control data of Q64RD/Q64RD-G (3/5)

| Device | | Item | Setting data | Setting range | Setting side |
|--------|---------------|--|--------------|------------------|--------------|
| | s[64] | 4-wire CH3 Factory default offset value (L) | | | |
| | s [65] | 4-wire CH3 Factory default offset value (H) | | - | System |
| | s [66] | 4-wire CH3 Factory default gain value (L) | | | |
| Q64RD | §[67] | 4-wire CH3 Factory default gain value (H) | | - | System |
| -G | §[68] | 4-wire CH3 User range settings offset value (L) | | | Quatana |
| | [69] | 4-wire CH3 User range settings offset value (H) | | - | System |
| | s [70] | 4-wire CH3 User range settings gain value (L) | | | Custom |
| | \$[71] | 4-wire CH3 User range settings gain value (H) | | - | System |
| (5) | [72] | 4-wire CH3 User range settings resistance offset value (L) | | | Custom |
| (\$ | [73] | 4-wire CH3 User range settings resistance offset value (H) | | - | System |
| (\$ | [74] | 4-wire CH3 User range settings resistance gain value (L) | | | Custom |
| (\$ | [75] | 4-wire CH3 User range settings resistance gain value (H) | | - | System |
| | s [76] | 3-wire CH4 Factory default offset value | - | - | System |
| | s[77] | 3-wire CH4 Factory default offset value | - | - | System |
| | s [78] | 3-wire CH4 Factory default gain value | - | - | System |
| Q64RD | s [79] | 3-wire CH4 Factory default gain value | _ | - | System |
| Q64RD | s [80] | 3-wire CH4 User range settings offset value | - | - | System |
| | <u> </u> | 3-wire CH4 User range settings offset value | - | - | System |
| | § [82] | 3-wire CH4 User range settings gain value | - | - | System |
| | ⑤ [83] | 3-wire CH4 User range settings gain value | - | - | System |
| | ⑤[76] | 3-wire CH4 Factory default offset value (L) | | | Custom |
| | \$[77] | 3-wire CH4 Factory default offset value (H) | _ | - | System |
| | ⑤ [78] | 3-wire CH4 Factory default gain value (L) | | | Quatana |
| Q64RD | s [79] | 3-wire CH4 Factory default gain value (H) | | - | System |
| -G | ⑤ [80] | 3-wire CH4 User range settings offset value (L) | | | Quatant |
| | s [81] | 3-wire CH4 User range settings offset value (H) | - | - | System |
| | \$[82] | 3-wire CH4 User range settings gain value (L) | | | Custom |
| | \$[83] | 3-wire CH4 User range settings gain value (H) | | - | System |
| (\$ | [84] | 3-wire CH4 User range settings resistance offset value (L) | | | Sustem |
| (\$ | [85] | 3-wire CH4 User range settings resistance offset value (H) | - | - | System |
| s | [86] | 3-wire CH4 User range settings resistance gain value (L) | | | Sustam |
| © [87] | | 3-wire CH4 User range settings resistance gain value (H) | - | - | System |
| | s [88] | 4-wire CH4 Factory default offset value | - | - | System |
| | s [89] | 4-wire CH4 Factory default offset value | - | - | System |
| | s [90] | 4-wire CH4 Factory default gain value | _ | - | System |
| Q64RD | s [91] | 4-wire CH4 Factory default gain value | - | - | System |
| QU4KD | s [92] | 4-wire CH4 User range settings offset value | _ | - | System |
| | s [93] | 4-wire CH4 User range settings offset value | _ | - | System |
| | s [94] | 4-wire CH4 User range settings gain value | _ | - | System |
| | s [95] | 4-wire CH4 User range settings gain value | _ | - | System |

Control data of Q64RD/Q64RD-G (4/5)

| Device | | Item | Setting data | Setting range | Setting side |
|---------------|---------|--|--------------|------------------|-----------------|
| | \$[88] | 4-wire CH4 Factory default offset value (L) | | | System |
| | \$ [89] | 4-wire CH4 Factory default offset value (H) | _ | _ | System |
| | s [90] | 4-wire CH4 Factory default gain value (L) | | - | System |
| Q64RD | s [91] | 4-wire CH4 Factory default gain value (H) | _ | | Gystem |
| -G | s [92] | 4-wire CH4 User range settings offset value (L) | | | System |
| | s [93] | 4-wire CH4 User range settings offset value (H) | | | System |
| | s [94] | 4-wire CH4 User range settings gain value (L) | | | System |
| | ঙ [95] | 4-wire CH4 User range settings gain value (H) | _ | _ | System |
| S | [96] | 4-wire CH4 User range settings resistance offset value (L) | | | System |
| © [97] | | 4-wire CH4 User range settings resistance offset value (H) | _ | | System |
| ⑤ [98] | | 4-wire CH4 User range settings resistance gain value (L) | | | System |
| S | [99] | 4-wire CH4 User range settings resistance gain value (H) | _ | _ | Gystelli |

Control data of Q64RD/Q64RD-G (5/5)

*1 : Setting is unnecessary. If setting is configured, offset/gain setting value is not read properly.

(11) Q64TD/Q64TDV-GH *1

| Device | Item | Setting data | Setting range | Setting side |
|----------------|--|---|---------------|--------------|
| s [0] | System area | _ | - | - |
| ঙ[1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| ⑤ [2] | Queters and | | | |
| ⑤ [3] | System area | _ | - | _ |
| ⑤ [4] | CH1 Factory default offset value | _ | - | System |
| s [5] | CH1 Factory default gain value | _ | - | System |
| ⑤ [6] | CH1 User range settings offset value | _ | - | System |
| ⑤ [7] | CH1 User range settings gain value | _ | - | System |
| § [8] | CH1 User range settings thermal EMF offset value (L) | | | Sustam |
| ⑤ [9] | CH1 User range settings thermal EMF offset value (H) | _ | - | System |
| s [10] | CH1 User range settings thermal EMF gain value (L) | | | Sustam |
| s[11] | CH1 User range settings thermal EMF gain value (H) | _ | - | System |
| s[12] | CH2 Factory default offset value | _ | - | System |
| <u></u> \$[13] | CH2 Factory default gain value | _ | - | System |
| \$[14] | CH2 User range settings offset value | _ | - | System |
| <u> </u> [15] | CH2 User range settings gain value | _ | - | System |
| <u></u> \$[16] | CH2 User range settings thermal EMF offset value (L) | | | Sustam |
| <u></u> §[17] | CH2 User range settings thermal EMF offset value (H) | _ | - | System |
| s [18] | CH2 User range settings thermal EMF gain value (L) | | | System |
| s [19] | CH2 User range settings thermal EMF gain value (H) | _ | _ | Oystern |
| s [20] | CH3 Factory default offset value | _ | - | System |
| ি [21] | CH3 Factory default gain value | _ | - | System |
| \$ [22] | CH3 User range settings offset value | - | - | System |
| s [23] | CH3 User range settings gain value | - | - | System |
| s [24] | CH3 User range settings thermal EMF offset value (L) | _ | _ | System |
| s [25] | CH3 User range settings thermal EMF offset value (H) | | | Gystern |
| s [26] | CH3 User range settings thermal EMF gain value (L) | _ | _ | System |
| S [27] | CH3 User range settings thermal EMF gain value (H) | | | Gystern |
| S [28] | CH4 Factory default offset value | _ | - | System |
| s [29] | CH4 Factory default gain value | _ | - | System |
| s [30] | CH4 User range settings offset value | - | - | System |
| s [31] | CH4 User range settings gain value | - | - | System |
| s [32] | CH4 User range settings thermal EMF offset value (L) | _ | _ | System |
| s [33] | CH4 User range settings thermal EMF offset value (H) | | | Cyotom |
| s [34] | CH4 User range settings thermal EMF gain value (L) | | | System |
| \$ [35] | CH4 User range settings thermal EMF gain value (H) | | | Gyotom |

*1 : Setting is unnecessary. If setting is configured, offset/gain setting value is not read properly.

(12) Q68TD-G-H02(H01)^{*1}

Control data of Q68TD-G-H02(H01) (1/2)

| Device | Item | Setting data | Setting range | Setting side |
|----------------|--|--|---------------|--------------|
| s [0] | System area | - | - | _ |
| ঙ[1] | Completion status | The instruction completion status is stored. 0 :Normal completion Other than 0 :Error completion (error code) | _ | System |
| জ [2] জ [3] | System area | - | - | _ |
| § [4] | CH1 Factory default offset value | _ | - | System |
| ⑤ [5] | CH1 Factory default gain value | _ | - | System |
| ⑤ [6] | CH1 User range settings offset value | _ | - | System |
| ⑤ [7] | CH1 User range settings gain value | _ | - | System |
| ⑤ [8] | CH1 User range settings thermal EMF offset value (L) | | | Quarteria |
| ⑤ [9] | CH1 User range settings thermal EMF offset value (H) | - | - | System |
| ⑤ [10] | CH1 User range settings thermal EMF gain value (L) | | | Quatara |
| ⑤ [11] | CH1 User range settings thermal EMF gain value (H) | - | - | System |
| <u></u> ।[12] | CH2 Factory default offset value | _ | - | System |
| s[13] | CH2 Factory default gain value | - | - | System |
| s[14] | CH2 User range settings offset value | _ | - | System |
| <u></u> ال | CH2 User range settings gain value | _ | - | System |
| s[16] | CH2 User range settings thermal EMF offset value (L) | | | Sustam |
| ⑤ [17] | CH2 User range settings thermal EMF offset value (H) | _ | - | System |
| s[18] | CH2 User range settings thermal EMF gain value (L) | | | Sustam |
| s[19] | CH2 User range settings thermal EMF gain value (H) | _ | - | System |
| s [20] | CH3 Factory default offset value | _ | - | System |
| s[21] | CH3 Factory default gain value | _ | - | System |
| s [22] | CH3 User range settings offset value | _ | - | System |
| s [23] | CH3 User range settings gain value | _ | - | System |
| s[24] | CH3 User range settings thermal EMF offset value (L) | | | System |
| s [25] | CH3 User range settings thermal EMF offset value (H) | _ | - | System |
| s [26] | CH3 User range settings thermal EMF gain value (L) | | | System |
| s [27] | CH3 User range settings thermal EMF gain value (H) | _ | - | System |
| s [28] | CH4 Factory default offset value | - | - | System |
| s [29] | CH4 Factory default gain value | - | - | System |
| s [30] | CH4 User range settings offset value | - | - | System |
| ⑤ [31] | CH4 User range settings gain value | - | - | System |
| s [32] | CH4 User range settings thermal EMF offset value (L) | | | System |
| ⑤ [33] | CH4 User range settings thermal EMF offset value (H) | | _ | Cystem |
| S[34] | CH4 User range settings thermal EMF gain value (L) | _ | _ | System |
| <u></u> | CH4 User range settings thermal EMF gain value (H) | | | Cystom |
| ⑤ [36] | CH5 Factory default offset value | - | - | System |
| ⑤ [37] | CH5 Factory default gain value | - | - | System |
| S [38] | CH5 User range settings offset value | - | _ | System |
| s [39] | CH5 User range settings gain value | - | - | System |
| ⑤ [40] | CH5 User range settings thermal EMF offset value (L) | _ | _ | System |
| <u></u> জ[41] | CH5 User range settings thermal EMF offset value (H) | | _ | System |

| Device | Item | Setting data | Setting range | Setting side |
|----------------|--|--------------|---------------|--------------|
| <u></u> آ [42] | CH5 User range settings thermal EMF gain value (L) | _ | _ | System |
| <u></u> آ [43] | CH5 User range settings thermal EMF gain value (H) | | | Oystelli |
| s [44] | CH6 Factory default offset value | - | - | System |
| <u></u> آ [45] | CH6 Factory default gain value | - | - | System |
| s [46] | CH6 User range settings offset value | _ | - | System |
| <u></u> ال | CH6 User range settings gain value | _ | - | System |
| s [48] | CH6 User range settings thermal EMF offset value (L) | | | System |
| s [49] | CH6 User range settings thermal EMF offset value (H) | _ | - | System |
| s [50] | CH6 User range settings thermal EMF gain value (L) | | | System |
| <u></u> \$[51] | CH6 User range settings thermal EMF gain value (H) | _ | _ | System |
| s [52] | CH7 Factory default offset value | - | - | System |
| s [53] | CH7 Factory default gain value | - | - | System |
| s [54] | CH7 User range settings offset value | - | - | System |
| s [55] | CH7 User range settings gain value | _ | - | System |
| \$ [56] | CH7 User range settings thermal EMF offset value (L) | | | System |
| s [57] | CH7 User range settings thermal EMF offset value (H) | _ | _ | System |
| s [58] | CH7 User range settings thermal EMF gain value (L) | | | System |
| s [59] | CH7 User range settings thermal EMF gain value (H) | - | - | System |
| ঙ [60] | CH8 Factory default offset value | - | - | System |
| s[61] | CH8 Factory default gain value | _ | - | System |
| s [62] | CH8 User range settings offset value | - | - | System |
| s [63] | CH8 User range settings gain value | _ | - | System |
| ঙ [64] | CH8 User range settings thermal EMF offset value (L) | | | System |
| ঙ [65] | CH8 User range settings thermal EMF offset value (H) | - | _ | System |
| S [66] | CH8 User range settings thermal EMF gain value (L) | | | Sustam |
| s[67] | CH8 User range settings thermal EMF gain value (H) | - | _ | System |

Control data of Q68TD-G-H02(H01) (2/2)

*1 : Setting is unnecessary. If setting is configured, offset/gain setting value is not read properly.

G_OGLOAD

(13) Q68RD3-G^{*1}

Control data of Q68RD3-G (1/2)

| Device | Item | Setting data | Setting range | Setting side |
|---------------|---|--|---------------|--------------|
| s[0] | System area | - | - | - |
| | | The instruction completion status is stored. | | |
| ⑤ [1] | Completion status | 0 :Normal completion Other than 0 :Error completion | - | System |
| | | (error code) | | |
| ⑤ [2] | | | | |
| ⑤[3] | System area | - | - | - |
| ⑤ [4] | CH1 Factory default offset value | _ | - | System |
| <u>\$</u> [5] | CH1 Factory default gain value | _ | - | System |
| ⑤ [6] | CH1 User range settings offset value | - | - | System |
| ⑤ [7] | CH1 User range settings gain value | - | - | System |
| ⑤ [8] | CH1 User range settings resistance offset value (L) | | | Sustam |
| s [9] | CH1 User range settings resistance offset value (H) | _ | - | System |
| <u>©[10]</u> | CH1 User range settings resistance gain value (L) | | | Sustam |
| s[11] | CH1 User range settings resistance gain value (H) | _ | - | System |
| s[12] | CH2 Factory default offset value | _ | - | System |
| s[13] | CH2 Factory default gain value | - | - | System |
| s[14] | CH2 User range settings offset value | - | - | System |
| <u> </u> [15] | CH2 User range settings gain value | - | - | System |
| s[16] | CH2 User range settings resistance offset value (L) | | | System |
| S[17] | CH2 User range settings resistance offset value (H) | _ | - | System |
| s [18] | CH2 User range settings resistance gain value (L) | | | System |
| <u>ঙ</u> [19] | CH2 User range settings resistance gain value (H) | _ | _ | System |
| s[20] | CH3 Factory default offset value | - | - | System |
| s[21] | CH3 Factory default gain value | - | - | System |
| \$[22] | CH3 User range settings offset value | - | - | System |
| s [23] | CH3 User range settings gain value | - | - | System |
| ⑤ [24] | CH3 User range settings resistance offset value (L) | _ | _ | System |
| s [25] | CH3 User range settings resistance offset value (H) | _ | _ | Oystern |
| s [26] | CH3 User range settings resistance gain value (L) | _ | | System |
| s [27] | CH3 User range settings resistance gain value (H) | | | oyotom |
| s [28] | CH4 Factory default offset value | _ | - | System |
| s [29] | CH4 Factory default gain value | _ | - | System |
| s [30] | CH4 User range settings offset value | _ | - | System |
| \$[31] | CH4 User range settings gain value | _ | - | System |
| s [32] | CH4 User range settings resistance offset value (L) | _ | _ | System |
| s [33] | CH4 User range settings resistance offset value (H) | | | ejetem |
| \$[34] | CH4 User range settings resistance gain value (L) | _ | _ | System |
| \$[35] | CH4 User range settings resistance gain value (H) | | | - , 0.0111 |
| \$[36] | CH5 Factory default offset value | _ | - | System |
| \$[37] | CH5 Factory default gain value | _ | - | System |
| s [38] | CH5 User range settings offset value | - | - | System |
| s [39] | CH5 User range settings gain value | - | - | System |
| s [40] | CH5 User range settings resistance offset value (L) | | _ | System |
| ঙ [41] | CH5 User range settings resistance offset value (H) | | | Cyotom |

| Control data of Q68RD3-G (2 | /2) |
|-----------------------------|-----|
|-----------------------------|-----|

| Device | Item | Setting data | Setting range | Setting side |
|----------------|---|--------------|---------------|--------------|
| s [42] | CH5 User range settings resistance gain value (L) | | | System |
| s [43] | CH5 User range settings resistance gain value (H) | _ | - | System |
| s [44] | CH6 Factory default offset value | _ | - | System |
| \$ [45] | CH6 Factory default gain value | _ | - | System |
| s [46] | CH6 User range settings offset value | _ | - | System |
| ঙ [47] | CH6 User range settings gain value | _ | - | System |
| s [48] | CH6 User range settings resistance offset value (L) | | | System |
| s [49] | CH6 User range settings resistance offset value (H) | _ | _ | System |
| \$ [50] | CH6 User range settings resistance gain value (L) | | | System |
| ঙ [51] | CH6 User range settings resistance gain value (H) | _ | _ | System |
| ঙ [52] | CH7 Factory default offset value | _ | - | System |
| \$ [53] | CH7 Factory default gain value | _ | - | System |
| s [54] | CH7 User range settings offset value | _ | - | System |
| s [55] | CH7 User range settings gain value | _ | - | System |
| s [56] | CH7 User range settings resistance offset value (L) | | | System |
| s [57] | CH7 User range settings resistance offset value (H) | _ | _ | System |
| s [58] | CH7 User range settings resistance gain value (L) | | | System |
| s [59] | CH7 User range settings resistance gain value (H) | _ | - | System |
| S [60] | CH8 Factory default offset value | _ | - | System |
| s [61] | CH8 Factory default gain value | _ | - | System |
| s [62] | CH8 User range settings offset value | _ | - | System |
| s [63] | CH8 User range settings gain value | _ | - | System |
| s[64] | CH8 User range settings resistance offset value (L) | | | System |
| s [65] | CH8 User range settings resistance offset value (H) | _ | _ | System |
| s [66] | CH8 User range settings resistance gain value (L) | | | Sustam |
| §[67] | CH8 User range settings resistance gain value (H) | - | _ | System |

*1 : Setting is unnecessary. If setting is configured, offset/gain setting value is not read properly.

G_OGLOAD

(14) Q61LD^{*1}

Control data of Q61LD (1/2)

| Device | Item | Setting data | Setting range | Setting side |
|----------------|--|--|---------------|--------------|
| s [0] | System area | _ | - | System |
| | | The instruction completion status is stored. | | |
| ⑤ [1] | Completion status | 0 :Normal completion | - | System |
| 0.1 | | Other than 0 :Error completion (error code) | | - |
| s[2] | | | | |
| ⑤[2] ⑤[3] | System area | - | - | System |
| <u> </u> | Load cell rated capacity (L) | | _ | System |
| s [5] | Load cell rated capacity (H) | - | - | System |
| ⑤ [6] | Load cell rated output | _ | - | System |
| <u>\$</u> [7] | Number of load cells in connection | - | - | System |
| <u>\$[8]</u> | Zero offset | _ | - | System |
| <u>s</u> [9] | System area | _ | - | System |
| <u>\$</u> [10] | Maximum weighing capacity setting (L) | _ | - | System |
| <u></u> | Maximum weighing capacity setting (H) | _ | - | System |
| <u>\$[12]</u> | Minimum division | _ | - | System |
| <u>\$</u> [13] | Decimal point position | _ | _ | System |
| <u>\$</u> [14] | Unit | _ | - | System |
| ⑤[15] | System area | _ | - | System |
| <u>\$</u> [16] | Standard weight setting (L) | _ | - | System |
| <u>\$</u> [17] | Standard weight setting (H) | _ | _ | System |
| <u>\$</u> [18] | Installation site gravitational acceleration (L) | _ | - | System |
| <u>s</u> [19] | Installation site gravitational acceleration (H) | _ | _ | System |
| ⑤[20] | Calibration site gravitational acceleration (L) | - | - | System |
| <u>\$[21]</u> | Calibration site gravitational acceleration (H) | _ | - | System |
| \$[22] | Digital output zero correction value (L) | _ | - | System |
| \$[23] | Digital output zero correction value (H) | _ | - | System |
| ⑤ [24] | Digital output span correction value (L) | _ | - | System |
| ⑤ [25] | Digital output span correction value (H) | _ | - | System |
| s [26] | | | | |
| to | System area | - | - | System |
| ⑤ [33] | | | | |
| s[34] | Instrumentation amplifier gain setting | - | - | System |
| <u></u> [35] | A/D converter gain setting | - | - | System |
| s [36] | Zero offset output value (L) | - | - | System |
| ⑤ [37] | Zero offset output value (H) | - | - | System |
| \$ [38] | Two-point zero calibration value (L) | - | - | System |
| s [39] | Two-point zero calibration value (H) | - | - | System |
| s [40] | Two-point span calibration value (L) | - | - | System |
| s[41] | Two-point span calibration value (H) | - | - | System |
| s [42] | | | | |
| to | System area | - | - | System |
| <u></u> ال | | | | |
| s [54] | 1.0mV/V zero calibration value (L) | _ | - | System |
| s [55] | 1.0mV/V zero calibration value (H) | - | - | System |

Control data of Q61LD (2/2)

| Device | Item | Setting data | Setting range | Setting side |
|----------------|------------------------------------|--------------|---------------|--------------|
| s [56] | 1.0mV/V span calibration value (L) | - | - | System |
| \$ [57] | 1.0mV/V span calibration value (H) | - | - | System |
| s [58] | 2.0mV/V zero calibration value (L) | - | - | System |
| s [59] | 2.0mV/V zero calibration value (H) | _ | - | System |
| §[60] | 2.0mV/V span calibration value (L) | - | - | System |
| s[61] | 2.0mV/V span calibration value (H) | _ | - | System |
| s [62] | 3.0mV/V zero calibration value (L) | _ | - | System |
| s [63] | 3.0mV/V zero calibration value (H) | _ | - | System |
| <u>\$[64]</u> | 3.0mV/V span calibration value (L) | _ | - | System |
| s [65] | 3.0mV/V span calibration value (H) | _ | - | System |
| s [66] | | | | |
| to | System area | _ | - | System |
| s [85] | | | | |

*1 : Setting is unnecessary. If setting is configured, offset/gain setting value is not read properly.

Program Example

[Structured ladder/FBD]

The following program reads out the offset/gain value of the A/D converter module mounted on the I/O numbers from X/Y00 to X/Y0F when the flag turns ON.

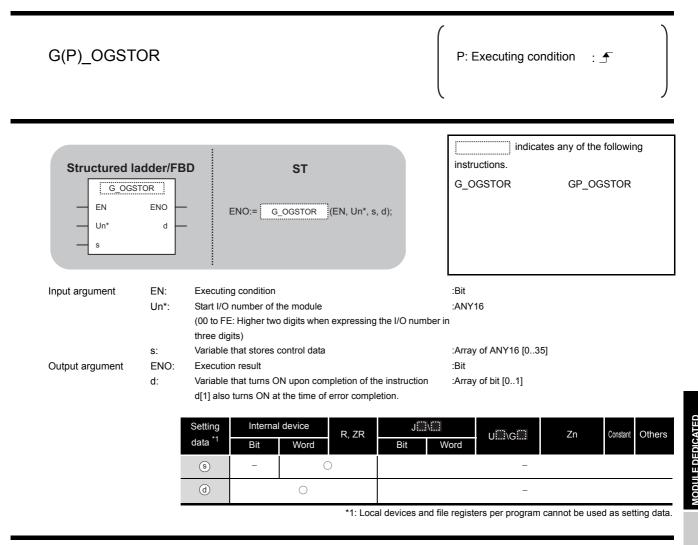
·Var_Flaq_Inst Specifies voltage MOV ΕN FNC -[[↑]] when instruction 0. ar ControlData[2] s flag turns ON Turns execution flag ON SET ENO ΕN Var_Flag_Exe d /ar Flag Exe Performs readout G_OGLOAD **FN** ENO · · · H00 Un* Var Result d Var ControlData s Var_Result[0] + Var_Result[1] Normal completion -|| · ||--1/1-ΕN ENO Turns execution flag OFF d /ar_Flaq_Exe Var_Result[0] Var_Result[1] Error completion Process on error completion ŀ ŀ

[ST]

(* Instruction flag ON *) IF(Var_Flag_Inst=TRUE)THEN -----. MOV(TRUE,0,Var_ControlData[2]); (* Specifies voltage *) (* Turns execution flag ON *) SET(TRUE, Var_Flag_Exe); END_IF; IF(Var_Flag_Exe=TRUE)THEN (* Execution flag ON *) G_OGLOAD(TRUE, H00, Var_ControlData, Var_Result); (* Performs readout *) IF(Var Result[0]=TRUE)THEN (* Execution finished *) IF(Var_Result[1]=FALSE)THEN (* Normal completion *) RST(TRUE, Var_Flag_Exe); (* Turns execution flag OFF *) ELSE (* Error completion *) ----Process on error completion *) END IF; END_IF; END_IF;

5.1.3 OGSTOR instruction

G_OGSTOR



☆ Function

This instruction restores the user range settings offset/gain values stored in the programmable controller CPU to the analog modules.

Control Data

(1) Q64AD/Q64DAN/Q64DA

| Device | Item | Setting data | Setting range | Setting side |
|----------------|---|---|-------------------|--------------|
| ⑤[0] | System area | - | - | - |
| ঙ [1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| ঙ [2] | Pass data classification setting | The value set for pass data classification setting (\$)[2] by the OGLOAD instruction is stored. 0: Voltage specified 1: Current specified $\frac{b15 \qquad b4 \qquad b3 \qquad b2 \qquad b1 \qquad b0}{\boxed{0 \qquad t0 \qquad 0 \qquad CH4 \ CH3 \ CH2 \ CH1}}$ | 0000н to 000Fн | System |
| s [3] | System area | _ | - | - |
| <u></u> \$[4] | CH1 Industrial shipment settings offset value | _ | - | System |
| s [5] | CH1 Industrial shipment settings gain value | _ | - | System |
| § [6] | CH2 Industrial shipment settings offset value | _ | - | System |
| ⑤ [7] | CH2 Industrial shipment settings gain value | _ | - | System |
| §[8] | CH3 Industrial shipment settings offset value | _ | - | System |
| s [9] | CH3 Industrial shipment settings gain value | _ | - | System |
| s[10] | CH4 Industrial shipment settings offset value | _ | - | System |
| \$[11] | CH4 Industrial shipment settings gain value | _ | - | System |
| <u></u> \$[12] | CH1 User range settings offset value | _ | - | System |
| s [13] | CH1 User range settings gain value | _ | - | System |
| s [14] | CH2 User range settings offset value | _ | - | System |
| s [15] | CH2 User range settings gain value | _ | - | System |
| s [16] | CH3 User range settings offset value | _ | - | System |
| <u></u> জ[17] | CH3 User range settings gain value | _ | - | System |
| s [18] | CH4 User range settings offset value | _ | - | System |
| ⑤ [19] | CH4 User range settings gain value | _ | - | System |

(2) Q68ADV/Q68ADI/Q68DAVN/Q68DAV/Q68DAIN/Q68DAI

| Device | Item | Setting data | Setting range | Setting side |
|--|---|---|---------------|--------------|
| ⑤ [0] | System area | _ | - | _ |
| | | The instruction completion status is stored. | | |
| ৩[1] | Completion status | 0 : Normal completion Other than 0 : Error completion (error code) | - | System |
| ()[2] | | | | |
| \$[2] \$[3] | System area | - | - | - |
| <u> </u> | CH1 Industrial shipment settings offset value | | | System |
| <u> </u> () () () () () () () () () (| CH1 Industrial shipment settings gain value | _ | | System |
| ⑤[6] | CH2 Industrial shipment settings offset value | _ | | System |
| ⑤[7] | CH2 Industrial shipment settings gain value | _ | | System |
| ⑤[8] | CH3 Industrial shipment settings offset value | _ | _ | System |
| ⑤[9] | CH3 Industrial shipment settings gain value | _ | _ | System |
| ⑤[10] | CH4 Industrial shipment settings offset value | _ | - | System |
| ⑤[11] | CH4 Industrial shipment settings gain value | - | _ | System |
| ⑤[12] | CH5 Industrial shipment settings offset value | _ | - | System |
| ⑤[13] | CH5 Industrial shipment settings gain value | | - | System |
| <u></u> [14] | CH6 Industrial shipment settings offset value | _ | - | System |
| <u></u> | CH6 Industrial shipment settings gain value | _ | _ | System |
| s [16] | CH7 Industrial shipment settings offset value | _ | - | System |
| s[17] | CH7 Industrial shipment settings gain value | - | - | System |
| s [18] | CH8 Industrial shipment settings offset value | _ | - | System |
| s [19] | CH8 Industrial shipment settings gain value | _ | - | System |
| <u>s</u> [20] | CH1 User range settings offset value | _ | - | System |
| <u>ি</u> [21] | CH1 User range settings gain value | _ | - | System |
| s [22] | CH2 User range settings offset value | _ | - | System |
| s [23] | CH2 User range settings gain value | _ | - | System |
| s [24] | CH3 User range settings offset value | - | - | System |
| s [25] | CH3 User range settings gain value | - | - | System |
| s [26] | CH4 User range settings offset value | - | - | System |
| s [27] | CH4 User range settings gain value | - | - | System |
| \$ [28] | CH5 User range settings offset value | - | - | System |
| \$ [29] | CH5 User range settings gain value | - | - | System |
| s [30] | CH6 User range settings offset value | - | - | System |
| s [31] | CH6 User range settings gain value | - | - | System |
| \$ [32] | CH7 User range settings offset value | - | - | System |
| \$ [33] | CH7 User range settings gain value | - | - | System |
| <u></u> ال | CH8 User range settings offset value | - | - | System |
| s [35] | CH8 User range settings gain value | - | _ | System |

(3) Q64AD-GH

| Device | Item | Setting data | Setting range | Setting side |
|------------------------------|---|--|-------------------|--------------|
| s [0] | System area | - | _ | _ |
| s [1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | - | System |
| ঙ [2] | Pass data classification setting | The value set for pass data classification setting (s) [2] by the OGLOAD instruction is stored. 0: Voltage specified 1: Current specified $\frac{b15 \qquad b4 \qquad b3 \qquad b2 \qquad b1 \qquad b0}{0 \qquad to \ 0 \qquad CH4 \ CH3 \ CH2 \ CH1}$ | 0000н to 000Fн | System |
| ⑤[3] | System area | _ | - | - |
| § [4] | CH1 Industrial shipment settings offset value (L) | | | |
| s [5] | CH1 Industrial shipment settings offset value (H) | - | - | System |
| \$[6] | CH1 Industrial shipment settings gain value (L) | | | 0 I |
| ⑤ [7] | CH1 Industrial shipment settings gain value (H) | - | - | System |
| § [8] | CH2 Industrial shipment settings offset value (L) | | | Sustan |
| s [9] | CH2 Industrial shipment settings offset value (H) | 1 - | - | System |
| s[10] | CH2 Industrial shipment settings gain value (L) | | | Sustam |
| <u></u> জ[11] | CH2 Industrial shipment settings gain value (H) | | - | System |
| s[12] | CH3 Industrial shipment settings offset value (L) | | | Sustam |
| s [13] | CH3 Industrial shipment settings offset value (H) | | - | System |
| ঙ [14] | CH3 Industrial shipment settings gain value (L) | | | System |
| <u> </u> । । । । | CH3 Industrial shipment settings gain value (H) | | _ | Oystern |
| <u></u> \$[16] | CH4 Industrial shipment settings offset value (L) | _ | _ | System |
| <u></u> \$[17] | CH4 Industrial shipment settings offset value (H) | | _ | Oystern |
| s[18] | CH4 Industrial shipment settings gain value (L) | _ | _ | System |
| s [19] | CH4 Industrial shipment settings gain value (H) | | | Gyotom |
| s[20] | CH1 User range settings offset value (L) | _ | _ | System |
| s[21] | CH1 User range settings offset value (H) | | | eyetein |
| \$[22] | CH1 User range settings gain value (L) | _ | _ | System |
| s [23] | CH1 User range settings gain value (H) | | | |
| ঙ [24] | CH2 User range settings offset value (L) | _ | _ | System |
| ঙ [25] | CH2 User range settings offset value (H) | | | |
| \$[26] | CH2 User range settings gain value (L) | _ | _ | System |
| ঙ [27] | CH2 User range settings gain value (H) | | | , |
| ি [28] | CH3 User range settings offset value (L) | _ | _ | System |
| ি [29] | CH3 User range settings offset value (H) | | | , |
| ি [30] | CH3 User range settings gain value (L) | _ | _ | System |
| ি [31] | CH3 User range settings gain value (H) | | | , |
| ি [32] | CH4 User range settings offset value (L) | _ | _ | System |
| \$[33] | CH4 User range settings offset value (H) | | | - , |
| জ [34] | CH4 User range settings gain value (L) | _ | _ | System |
| \$ [35] | CH4 User range settings gain value (H) | | | , |

(4) Q62AD-DGH

| Device | Item | Setting data | Setting range | Setting side |
|------------------------|---|---|---------------|--------------|
| ⑤[0] | System area | - | - | _ |
| <u></u> জ[1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| s [2] | System area | _ | _ | _ |
| ⑤[3] | | | | |
| ⑤ [4] | CH1 Industrial shipment settings offset value (L) | _ | _ | System |
| ⑤ [5] | CH1 Industrial shipment settings offset value (H) | | | Gystein |
| ⑤[6] | CH1 Industrial shipment settings gain value (L) | | | System |
| ঙ [7] | CH1 Industrial shipment settings gain value (H) | | _ | System |
| ⑤[8] | CH2 Industrial shipment settings offset value (L) | | | System |
| ⑤[9] | CH2 Industrial shipment settings offset value (H) | _ | - | System |
| s [10] | CH2 Industrial shipment settings gain value (L) | | | System |
| s[11] | CH2 Industrial shipment settings gain value (H) | | _ | System |
| \$[12] to \$[19] | System area | _ | _ | - |
| s [20] | CH1 User range settings offset value (L) | | | System |
| s [21] | CH1 User range settings offset value (H) | _ | _ | System |
| s [22] | CH1 User range settings gain value (L) | | | System |
| s [23] | CH1 User range settings gain value (H) | | - | System |
| s [24] | CH2 User range settings offset value (L) | | | System |
| s [25] | CH2 User range settings offset value (H) | | _ | System |
| s [26] | CH2 User range settings gain value (L) | | | System |
| s [27] | CH2 User range settings gain value (H) | | _ | System |
| \$[28] to \$[35] | System area | - | _ | System |

G_OGSTOR

(5) Q68AD-G

| Device | Item | Setting data | Setting range | Setting side |
|---------------|---|---|---------------|--------------|
| ⑤ [0] | System area | _ | _ | _ |
| | | The instruction completion status is stored. | | |
| s[1] | Completion status | 0 :Normal completion | - | System |
| | | Other than 0 :Error completion (error code) Specify the voltage/current of the offset/gain values to | | |
| | | be read. | | |
| © [2] | Pass data classification setting | 0: Voltage specified | 0000н to | User |
| \$[2] | | 1: Current specified | 0000Fн | 0361 |
| | | b15 b8 b7 b6 b5 b4 b3 b2 b1 b0 0 to 0 CH8 CH7 CH6 CH5 CH4 CH3 CH2 CH1 | | |
| s [3] | System area | _ | _ | _ |
| <u> </u> | CH1 Industrial shipment settings offset value | _ | _ | System |
| ⑤[5] | CH1 Industrial shipment settings gain value | | _ | System |
| ⑤[6] | CH2 Industrial shipment settings offset value | _ | _ | System |
| ⑤[7] | CH2 Industrial shipment settings gain value | _ | _ | System |
| ⑤[8] | CH3 Industrial shipment settings offset value | _ | - | System |
| <u> </u> | CH3 Industrial shipment settings gain value | | _ | System |
| <u>\$[10]</u> | CH4 Industrial shipment settings offset value | | _ | System |
| ⑤[11] | CH4 Industrial shipment settings gain value | _ | _ | System |
| <u>\$[12]</u> | CH5 Industrial shipment settings offset value | _ | _ | System |
| <u> </u> | CH5 Industrial shipment settings gain value | _ | _ | System |
| <u> </u> | CH6 Industrial shipment settings offset value | _ | _ | System |
| <u>s</u> [15] | CH6 Industrial shipment settings gain value | _ | _ | System |
| s[16] | CH7 Industrial shipment settings offset value | _ | - | System |
| s[17] | CH7 Industrial shipment settings gain value | _ | - | System |
| s [18] | CH8 Industrial shipment settings offset value | _ | - | System |
| s [19] | CH8 Industrial shipment settings gain value | _ | - | System |
| s[20] | CH1 User range settings offset value | _ | - | System |
| <u></u> জ[21] | CH1 User range settings gain value | _ | - | System |
| \$[22] | CH2 User range settings offset value | _ | - | System |
| s [23] | CH2 User range settings gain value | _ | _ | System |
| s [24] | CH3 User range settings offset value | _ | _ | System |
| s [25] | CH3 User range settings gain value | _ | _ | System |
| s [26] | CH4 User range settings offset value | _ | - | System |
| s[27] | CH4 User range settings gain value | _ | _ | System |
| s [28] | CH5 User range settings offset value | _ | - | System |
| s [29] | CH5 User range settings gain value | _ | - | System |
| s [30] | CH6 User range settings offset value | _ | - | System |
| <u></u> জ[31] | CH6 User range settings gain value | _ | _ | System |
| s [32] | CH7 User range settings offset value | _ | - | System |
| s [33] | CH7 User range settings gain value | _ | - | System |
| s[34] | CH8 User range settings offset value | _ | - | System |
| \$ [35] | CH8 User range settings gain value | _ | _ | System |

(6) Q66AD-DG^{*1}

| Device | Item | Setting data | Setting range | Setting side |
|--------------------------|---|---|---------------|--------------|
| s [0] | System area | _ | _ | _ |
| ©[1] | Completion status | The instruction completion status is stored. 0 :Normal completion Other than 0 :Error completion (error code) | _ | System |
| <u> </u> | System area | - | - | - |
| ⑤ [4] | CH1 Industrial shipment settings offset value | _ | - | System |
| ⑤ [5] | CH1 Industrial shipment settings gain value | _ | - | System |
| ⑤[6] | CH2 Industrial shipment settings offset value | _ | - | System |
| ⑤ [7] | CH2 Industrial shipment settings gain value | _ | - | System |
| §[8] | CH3 Industrial shipment settings offset value | _ | - | System |
| s [9] | CH3 Industrial shipment settings gain value | _ | - | System |
| s [10] | CH4 Industrial shipment settings offset value | _ | - | System |
| \$[11] | CH4 Industrial shipment settings gain value | _ | - | System |
| s [12] | CH5 Industrial shipment settings offset value | _ | - | System |
| <u></u> জ[13] | CH5 Industrial shipment settings gain value | _ | - | System |
| <u></u> \$[14] | CH6 Industrial shipment settings offset value | _ | - | System |
| <u></u> জ[15] | CH6 Industrial shipment settings gain value | _ | - | System |
| \$ [16] to \$ [19] | System area | _ | _ | System |
| s [20] | CH1 User range settings offset value | _ | - | System |
| s [21] | CH1 User range settings gain value | | - | System |
| s [22] | CH2 User range settings offset value | _ | - | System |
| s [23] | CH2 User range settings gain value | _ | - | System |
| s [24] | CH3 User range settings offset value | _ | - | System |
| s [25] | CH3 User range settings gain value | _ | - | System |
| s [26] | CH4 User range settings offset value | _ | - | System |
| \$[27] | CH4 User range settings gain value | _ | - | System |
| \$ [28] | CH5 User range settings offset value | | - | System |
| s [29] | CH5 User range settings gain value | _ | - | System |
| ⑤ [30] | CH6 User range settings offset value | - | - | System |
| s [31] | CH6 User range settings gain value | _ | - | System |
| \$ [32] to \$ [35] | System area | _ | - | System |

*1 : Setting is unnecessary. If setting is configured, offset/gain setting value is not read properly.

(7) Q62DAN/Q62DA

| Device | Item | Setting data | Setting range | Setting side |
|---------------|---|--|-------------------|--------------|
| ⑤[0] | System area | _ | - | - |
| s[1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | - | System |
| ®[2] | Pass data classification setting | The value set for pass data classification setting (s) [2] by the OGLOAD instruction is stored. 0: Voltage specified 1: Current specified b15 	b2 	b1 	b0 0 to 0 CH2 CH1 | 0000н to 0003н | System |
| s [3] | System area | _ | - | - |
| ⑤ [4] | CH1 Industrial shipment settings offset value | _ | - | System |
| \$ [5] | CH1 Industrial shipment settings gain value | _ | - | System |
| ⑤[6] | CH2 Industrial shipment settings offset value | _ | - | System |
| ⑤ [7] | CH2 Industrial shipment settings gain value | _ | - | System |
| §[8] | CH1 User range settings offset value | _ | - | System |
| s [9] | CH1 User range settings gain value | _ | - | System |
| s[10] | CH2 User range settings offset value | _ | - | System |
| <u></u> জ[11] | CH2 User range settings gain value | _ | - | System |

(8) Q62DA-FG

| Device | Item | Setting data | Setting range | Setting side |
|---------------|---|--|---------------|--------------|
| ⑤ [0] | System area | _ | _ | _ |
| ঙ[1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| ®[2] | Pass data classification setting | The value set for pass data classification setting (\$) [2] by the OGLOAD instruction is stored. 0H: User range setting 1 specified 1H: User range setting 2 specified 2H: User range setting 3 specified 2H: User range setting 3 specified 0H 0H CH2 CH1 | - | System |
| \$[3] | System area | - | - | - |
| s[4] | CH1 Industrial shipment settings offset value (used for D/A) | _ | - | System |
| ঙ [5] | CH1 Industrial shipment settings gain value (used for D/A) | _ | - | System |
| s [6] | CH2 Industrial shipment settings offset value (used for D/A) | _ | - | System |
| s[7] | CH2 Industrial shipment settings gain value (used for D/A) | - | - | System |
| ⑤ [8] | CH1 Industrial shipment settings offset value (used for monitor output) | - | - | System |
| ঙ [9] | CH1 Industrial shipment settings gain value (used for monitor output) | - | - | System |
| <u>জ</u> [10] | CH2 Industrial shipment settings offset value (used for monitor output) | _ | - | System |
| \$[11] | CH2 Industrial shipment settings gain value (used for monitor output) | _ | - | System |
| s [12] | CH1 User range settings offset value (used for D/A) | _ | - | System |
| ি [13] | CH1 User range settings gain value (used for D/A) | - | - | System |
| <u>ি</u> [14] | CH2 User range settings offset value (used for D/A) | - | - | System |
| s [15] | CH2 User range settings gain value (used for D/A) | _ | - | System |
| s [16] | CH1 User range settings offset value (used for monitor output) | _ | _ | System |
| s [17] | CH1 User range settings gain value (used for monitor output) | _ | _ | System |
| <u>ি</u> [18] | CH2 User range settings offset value (used for monitor output) | - | _ | System |
| s [19] | CH2 User range settings gain value (used for monitor output) | - | - | System |

(9) Q66DA-G^{*1}

| Device | Item | Setting data | Setting range | Setting side |
|---------------|---|--|---------------|--------------|
| s [0] | System area | _ | _ | - |
| \$[1] | Completion status | The instruction completion status is stored. 0 :Normal completion Other than 0 :Error completion (error code) | _ | System |
| s [2] | Pass data classification setting | The value set for pass data classification setting (s) [2] by the OGLOAD instruction is stored. OH: User range setting 1 specified 1H: User range setting 2 specified 2H: User range setting 3 specified b15 b12 b11 b10 b9 b8 b7 b6 b5 b4 b3 b2 b1 b0 15 to $10 cm$ 10 | - | User |
| \$[3] | System area | _ | - | - |
| s [4] | CH1 Industrial shipment settings offset value | - | - | System |
| <u></u> \$[5] | CH1 Industrial shipment settings gain value | - | - | System |
| s [6] | CH2 Industrial shipment settings offset value | - | - | System |
| \$[7] | CH2 Industrial shipment settings gain value | - | - | System |
| (8) | CH3 Industrial shipment settings offset value | _ | - | System |
| s [9] | CH3 Industrial shipment settings gain value | _ | - | System |
| s [10] | CH4 Industrial shipment settings offset value | _ | - | System |
| s[11] | CH4 Industrial shipment settings gain value | _ | - | System |
| <u></u> জ[12] | CH5 Industrial shipment settings offset value | _ | - | System |
| <u></u> জ[13] | CH5 Industrial shipment settings gain value | _ | - | System |
| <u></u> জ[14] | CH6 Industrial shipment settings offset value | _ | - | System |
| s [15] | CH6 Industrial shipment settings gain value | _ | - | System |
| ⑤ [16] | CH1 User range settings offset value | _ | - | System |
| §[17] | CH1 User range settings gain value | _ | - | System |
| ⑤ [18] | CH2 User range settings offset value | _ | _ | System |
| ⑤ [19] | CH2 User range settings gain value | _ | - | System |
| ⑤ [20] | CH3 User range settings offset value | _ | - | System |
| ⑤ [21] | CH3 User range settings gain value | _ | - | System |
| \$[22] | CH4 User range settings offset value | _ | - | System |
| s [23] | CH4 User range settings gain value | _ | - | System |
| s[24] | CH5 User range settings offset value | _ | - | System |
| \$[25] | CH5 User range settings gain value | _ | - | System |
| \$[26] | CH6 User range settings offset value | _ | - | System |
| \$[27] | CH6 User range settings gain value | _ | - | System |
| <u>s</u> [28] | | | | |
| to | System area | _ | - | - |
| s [35] | | | | |

*1 : Set the data only to the Pass data classification setting \$ [2].

When the data is written to the area to be set by system, offset/gain setting value is not read properly.

Control data of Q64RD/Q64RD-G (1/5)

| Dev | vice | Item | Setting data | Setting range | Setting side |
|-------|----------------|--|---|------------------|-----------------|
| S | [0] | System area | - | _ | - |
| | [1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| S | [2] | System area | | | |
| S | [3] | System area | _ | _ | _ |
| | ⑤ [4] | 3-wire CH1 Factory default offset value | - | - | System |
| | ⑤ [5] | 3-wire CH1 Factory default offset value | - | - | System |
| | ⑤ [6] | 3-wire CH1 Factory default gain value | - | - | System |
| Q64RD | (\$ [7] | 3-wire CH1 Factory default gain value | - | - | System |
| QUAND | (8) | 3-wire CH1 User range settings offset value | - | - | System |
| | s [9] | 3-wire CH1 User range settings offset value | - | - | System |
| | s[10] | 3-wire CH1 User range settings gain value | - | - | System |
| | <u> </u> (। | 3-wire CH1 User range settings gain value | - | - | System |
| | s [4] | 3-wire CH1 Factory default offset value (L) | | | Sustam |
| | s [5] | 3-wire CH1 Factory default offset value (H) | | _ | System |
| | s [6] | 3-wire CH1 Factory default gain value (L) | | | Sustam |
| Q64RD | s[7] | 3-wire CH1 Factory default gain value (H) | | _ | System |
| -G | (8) | 3-wire CH1 User range settings offset value (L) | | | Sustam |
| | s [9] | 3-wire CH1 User range settings offset value (H) | | _ | System |
| | <u>জ</u> [10] | 3-wire CH1 User range settings gain value (L) | | | Custom |
| | \$[11] | 3-wire CH1 User range settings gain value (H) | | _ | System |
| S | [12] | 3-wire CH1 User range settings resistance offset value (L) | | | Queters |
| S | [13] | 3-wire CH1 User range settings resistance offset value (H) | | _ | System |
| S | [14] | 3-wire CH1 User range settings resistance gain value (L) | | | Custom |
| S | [15] | 3-wire CH1 User range settings resistance gain value (H) | | _ | System |
| | s [16] | 4-wire CH1 Factory default offset value | - | - | System |
| | s[17] | 4-wire CH1 Factory default offset value | - | - | System |
| | s[18] | 4-wire CH1 Factory default gain value | - | - | System |
| 06400 | s [19] | 4-wire CH1 Factory default gain value | - | - | System |
| Q64RD | s [20] | 4-wire CH1 User range settings offset value | - | - | System |
| | <u>জ</u> [21] | 4-wire CH1 User range settings offset value | - | - | System |
| | ⑤ [22] | 4-wire CH1 User range settings gain value | - | - | System |
| | s [23] | 4-wire CH1 User range settings gain value | - | _ | System |

| Other Avire CH1 Factory default offset value (1) - ratiog store 064RD 0 [19] 4-wire CH1 Factory default gain value (1) - - System 0 0 [19] 4-wire CH1 Factory default gain value (1) - - System 0 0 [19] 4-wire CH1 User range settings offset value (1) - - System 0 2[2] 4-wire CH1 User range settings offset value (1) - - System 0 [2]3 4-wire CH1 User range settings offset value (1) - - System 0 [2]4 4-wire CH1 User range settings resistance offset value (1) - - System 0 [2]3 4-wire CH1 User range settings resistance gain value (1) - - System 0 [2]3 4-wire CH1 User range settings resistance gain value (1) - - System 0 [2]3 3-wire CH2 Eactory default offset value - - System 0 [3]3 3-wire CH2 Eactory default offset value - - System < | Dev | vice | Item | Setting data | Setting | Setting |
|---|----------|----------------|--|--------------|---------|-----------|
| 0:010 4-wire CH1 Factory default diset value (h) - - System 0:181 4-wire CH1 Factory default gain value (L) - - System 0:191 4-wire CH1 Factory default gain value (L) - - System 0:191 4-wire CH1 User range settings offset value (h) - - System 0:121 4-wire CH1 User range settings and value (h) - - System 0:121 4-wire CH1 User range settings resistance offset value (h) - - System 0:121 4-wire CH1 User range settings resistance offset value (h) - - System 0:123 4-wire CH1 User range settings resistance gain value (h) - - System 0:123 4-wire CH1 User range settings resistance gain value (h) - - System 0:123 3-wire CH2 Factory default dister value - - System 0:123 3-wire CH2 Factory default dister value - - System 0:123 3-wire CH2 Factory default dister value - - System <td< td=""><td colspan="2"></td><td>4 wire CH1 Eactory default offset value (L)</td><td></td><td>range</td><td>side</td></td<> | | | 4 wire CH1 Eactory default offset value (L) | | range | side |
| Other Other Other System 0110 4-wire CHT Factory default gain value (1) - - System 0120 4-wire CHT Factory default gain value (1) - - System 0121 4-wire CHT User range settings offset value (1) - - System 0121 4-wire CHT User range settings and value (1) - - System 0122 4-wire CHT User range settings resistance offset value (1) - - System 0122 4-wire CHT User range settings resistance offset value (1) - - System 0128 4-wire CHT User range settings resistance and value (1) - - System 0128 4-wire CHT User range settings resistance and value (1) - - System 0128 3-wire CH2 Factory default offset value - - System 0131 3-wire CH2 Factory default gain value - - System 0133 3-wire CH2 Eactory default gain value - - System 0133 3-wire CH2 Eactory default gain value | | | | - | - | System |
| OBJAND O(10) 4-wire CH1 Factory default gain value (H) - - System G (2) 4-wire CH1 User range settings offset value (L) - - System G (2) 4-wire CH1 User range settings gain value (L) - - System G (2) 4-wire CH1 User range settings gain value (L) - - System G (2) 4-wire CH1 User range settings resistance gain value (L) - - System G (2) 4-wire CH1 User range settings resistance gain value (L) - - System G (2) 4-wire CH1 User range settings resistance gain value (L) - - System G (2) 4-wire CH1 User range settings resistance gain value (L) - - System G (2) 3-wire CH2 Factory default offset value - - System G (3) 3-wire CH2 Eactory default offset value - - System G (3) 3-wire CH2 User range settings offset value - - System< | | | | | | |
| G C[2] 4-wire CH1 User range settings offset value (1) - - System C[2] 4-wire CH1 User range settings an value (1) - - System C[2] 4-wire CH1 User range settings existance offset value (1) - - System C[2] 4-wire CH1 User range settings resistance offset value (1) - - System C[2] 4-wire CH1 User range settings resistance offset value (1) - - System C[2] 4-wire CH1 User range settings resistance gain value (1) - - System C[2] 4-wire CH1 User range settings resistance gain value (1) - - System C[2] 4-wire CH1 User range settings resistance gain value (1) - - System C[2] 4-wire CH1 User range settings gain value - - System C[3] 3-wire CH2 Factory default offset value - - System C[3] 3-wire CH2 User range settings gain value - - System C[3] 3-wire CH2 User range settings gain value (1) - - <t< td=""><td>00400</td><td></td><td></td><td>-</td><td>-</td><td>System</td></t<> | 00400 | | | - | - | System |
| O [20] Initial Section Processing Constraints - - System O [21] 4-wire CH1 User range settings dist value (1) - - System O [23] 4-wire CH1 User range settings gain value (1) - - System O [24] 4-wire CH1 User range settings resistance offset value (1) - - System O [25] 4-wire CH1 User range settings resistance dist value (1) - - System O [26] 4-wire CH1 User range settings resistance dist value (1) - - System O [26] 4-wire CH1 User range settings resistance dist value (1) - - System O [27] 4-wire CH1 User range settings resistance dist value (1) - - System O [28] 3-wire CH2 Factory default dist value - - System O [28] 3-wire CH2 Eactory default dist value - - System O [28] 3-wire CH2 User range settings dist value - - System O [28] 3-wire CH2 User range settings dist value (1) - - System <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> | | | | | | |
| Ote Ote System O[22] 4-wire CH1 User range settings gain value (L) - - System O[23] 4-wire CH1 User range settings resistance offset value (L) - - System O[26] 4-wire CH1 User range settings resistance offset value (L) - - System O[26] 4-wire CH1 User range settings resistance gain value (L) - - System O[28] 4-wire CH1 User range settings resistance gain value (H) - - System O[28] 3-wire CH2 Factory default offset value - - System O[29] 3-wire CH2 Factory default gain value - - System O[30] 3-wire CH2 Eactory default gain value - - System O[31] 3-wire CH2 User range settings offset value - - System O[33] 3-wire CH2 User range settings offset value - - System O[34] 3-wire CH2 User range settings offset value (L) - - System O[34] 3-wire CH2 Eactory default offset value (L) | | | | - | - | System |
| Organ 4-wire CH1 User range settings resistance offset value (L) - - System O(24) 4-wire CH1 User range settings resistance offset value (L) - - System O(25) 4-wire CH1 User range settings resistance offset value (L) - - System O(27) 4-wire CH1 User range settings resistance gain value (L) - - System O(28) 4-wire CH2 Factory default offset value - - System O(29) 3-wire CH2 Factory default offset value - - System O(30) 3-wire CH2 Factory default offset value - - System O(31) 3-wire CH2 User range settings offset value - - System O(32) 3-wire CH2 User range settings offset value - - System O(32) 3-wire CH2 User range settings offset value - - System O(33) 3-wire CH2 User range settings offset value - - System O(33) 3-wire CH2 Factory default offset value (L) - - System < | | | | | | |
| Old 4-wire CH1 User range settings resistance offset value (L) - - System O(26) 4-wire CH1 User range settings resistance gain value (L) - - System O(27) 4-wire CH1 User range settings resistance gain value (L) - - System O(28) 4-wire CH1 User range settings resistance gain value (L) - - System O(28) 3-wire CH2 Factory default offset value - - System O(30) 3-wire CH2 Factory default gain value - - System O(31) 3-wire CH2 Factory default gain value - - System O(32) 3-wire CH2 Factory default offset value - - System O(33) 3-wire CH2 Factory default offset value - - System O(33) 3-wire CH2 Factory default offset value - - System O(34) 3-wire CH2 Factory default offset value (L) - - System O(212) 3-wire CH2 Factory default gen value (H) - - System O(33) <t< td=""><td></td><td></td><td></td><td>-</td><td>-</td><td>System</td></t<> | | | | - | - | System |
| O[25] 4-wire CH1 User range settings resistance offset value (H) - - System O[26] 4-wire CH1 User range settings resistance gain value (L) - - System O[27] 4-wire CH1 User range settings resistance gain value (H) - - System O[28] 3-wire CH2 Factory default offset value - - System O[130] 3-wire CH2 Factory default gain value - - System O[131] 3-wire CH2 Factory default gain value - - System O[133] 3-wire CH2 Factory default gain value - - System O[133] 3-wire CH2 Factory default gain value - - System O[134] 3-wire CH2 User range settings gain value - - System O[135] 3-wire CH2 User range settings gain value (L) - - System O[136] 3-wire CH2 Factory default offset value (L) - - System O[131] 3-wire CH2 factory default offset value (L) - - System O[131] <t< td=""><td></td><td></td><td></td><td></td><td></td><td></td></t<> | | | | | | |
| Op/261 4-wire CH1 User range settings resistance gain value (1) - - System 0[27] 4-wire CH1 User range settings resistance gain value (H) - - System 0[28] 3-wire CH2 Factory default offset value - - System 0[30] 3-wire CH2 Factory default gain value - - System 0[31] 3-wire CH2 Factory default gain value - - System 0[32] 3-wire CH2 User range settings offset value - - System 0[33] 3-wire CH2 User range settings offset value - - System 0[33] 3-wire CH2 User range settings offset value - - System 0[34] 3-wire CH2 User range settings gain value - - System 0[30] 3-wire CH2 Factory default offset value (L) - - System 0[31] 3-wire CH2 Eactory default gain value (L) - - System 0[32] 3-wire CH2 User range settings offset value (L) - - System 0[33] 3-wire CH | | | | - | - | System |
| OLM - System Operating 0[27] 4-wire CH1 User range settings resistance gain value (H) - - System 0[28] 3-wire CH2 Factory default offset value - - System 0[30] 3-wire CH2 Factory default offset value - - System 0[31] 3-wire CH2 Factory default gain value - - System 0[32] 3-wire CH2 User range settings offset value - - System 0[33] 3-wire CH2 User range settings offset value - - System 0[34] 3-wire CH2 User range settings gain value - - System 0[35] 3-wire CH2 User range settings gain value - - System 0[35] 3-wire CH2 User range settings gain value - - System 0[36] 3-wire CH2 Factory default offset value (L) - - System 0[37] 3-wire CH2 User range settings gain value (L) - - System 0[38] 3-wire CH2 User range settings gain value (L) - <t< td=""><td></td><td></td><td></td><td></td><td></td><td></td></t<> | | | | | | |
| O[28] 3-wire CH2 Factory default offset value - - System O[64RD 0[30] 3-wire CH2 Factory default gain value - - System 0[30] 3-wire CH2 Factory default gain value - - System 0[30] 3-wire CH2 Factory default gain value - - System 0[32] 3-wire CH2 Factory default gain value - - System 0[33] 3-wire CH2 User range settings offset value - - System 0[34] 3-wire CH2 User range settings gain value - - System 0[35] 3-wire CH2 Factory default offset value (L) - - System 0[36] 3-wire CH2 Factory default gain value (L) - - System 0[31] 3-wire CH2 Factory default gain value (L) - - System 0[32] 3-wire CH2 User range settings offset value (L) - - System 0[33] 3-wire CH2 User range settings resistance offset value (L) - - System 0[34] 3-wire CH2 | | | | - | - | System |
| OLD System QG4RD 3-wire CH2 Factory default offset value - - System 0[30] 3-wire CH2 Factory default gain value - - System 0[31] 3-wire CH2 Factory default gain value - - System 0[32] 3-wire CH2 User range settings offset value - - System 0[33] 3-wire CH2 User range settings gain value - - System 0[34] 3-wire CH2 User range settings gain value - - System 0[35] 3-wire CH2 User range settings gain value - - System 0[36] 3-wire CH2 Factory default offset value (L) - - System 0[30] 3-wire CH2 Factory default gain value (L) - - System 0[31] 3-wire CH2 User range settings offset value (L) - - System 0[32] 3-wire CH2 User range settings offset value (L) - - System 0[33] 3-wire CH2 User range settings resistance offset value (L) - - System | s | _ | | | | |
| Ocket Right | | | | - | - | |
| Octa System O(31) 3-wire CH2 Factory default gain value - - System O(32) 3-wire CH2 User range settings offset value - - System O(33) 3-wire CH2 User range settings gain value - - System O(34) 3-wire CH2 User range settings gain value - - System O(35) 3-wire CH2 Eactory default offset value (L) - - System O(30) 3-wire CH2 Eactory default offset value (L) - - System O(31) 3-wire CH2 Factory default offset value (L) - - System O(31) 3-wire CH2 Factory default gain value (L) - - System O(32) 3-wire CH2 User range settings offset value (L) - - System O(33) 3-wire CH2 User range settings gain value (L) - - System O(34) 3-wire CH2 User range settings gain value (L) - - System O(36) 3-wire CH2 User range settings resistance offset value (L) - - System <td><u>s</u> [29]</td> <td></td> <td>-</td> <td>-</td> <td></td> | | <u>s</u> [29] | | - | - | |
| Od4RD O O O O © [32] 3-wire CH2 User range settings offset value - - System © [34] 3-wire CH2 User range settings gain value - - System © [35] 3-wire CH2 User range settings gain value - - System © [36] 3-wire CH2 Factory default offset value (L) - - System © [29] 3-wire CH2 Factory default gain value (L) - - System © [30] 3-wire CH2 Factory default gain value (L) - - System © [31] 3-wire CH2 Factory default gain value (L) - - System © [32] 3-wire CH2 User range settings offset value (L) - - System © [33] 3-wire CH2 User range settings offset value (L) - - System © [34] 3-wire CH2 User range settings gain value (L) - - System © [36] 3-wire CH2 User range settings resistance offset value (L) - - System © [37] 3-wire CH2 User range settings resistanc | | <u>\$</u> [30] | 3-wire CH2 Factory default gain value | - | - | |
| Other System © [33] 3-wire CH2 User range settings offset value - - System © [34] 3-wire CH2 User range settings gain value - - System © [35] 3-wire CH2 User range settings gain value - - System © [28] 3-wire CH2 Factory default offset value (L) - - System © [30] 3-wire CH2 Factory default gain value (L) - - System © [30] 3-wire CH2 Factory default gain value (L) - - System © [31] 3-wire CH2 Factory default gain value (H) - - System © [33] 3-wire CH2 User range settings offset value (L) - - System © [34] 3-wire CH2 User range settings offset value (L) - - System © [35] 3-wire CH2 User range settings gain value (L) - - System © [36] 3-wire CH2 User range settings effset value (L) - - System © [36] 3-wire CH2 User range settings effset value (H) - - | Q64RD | s [31] | 3-wire CH2 Factory default gain value | - | - | System |
| Other System © [34] 3-wire CH2 User range settings gain value - - System © [35] 3-wire CH2 Factory default offset value (L) - - System © [29] 3-wire CH2 Factory default offset value (L) - - System © [30] 3-wire CH2 Factory default gain value (L) - - System © [30] 3-wire CH2 Factory default gain value (L) - - System © [31] 3-wire CH2 Factory default gain value (H) - - System © [33] 3-wire CH2 User range settings offset value (L) - - System © [33] 3-wire CH2 User range settings offset value (L) - - System © [34] 3-wire CH2 User range settings gain value (H) - - System © [36] 3-wire CH2 User range settings resistance offset value (L) - - System © [36] 3-wire CH2 User range settings resistance offset value (H) - - - System © [37] 3-wire CH2 User range settings resistance offset value (| | s [32] | 3-wire CH2 User range settings offset value | - | - | System |
| Image: Classical stress of the classical stress | | s [33] | 3-wire CH2 User range settings offset value | - | - | System |
| Image: Constraint of the second sec | | <u>ি</u> [34] | 3-wire CH2 User range settings gain value | - | - | System |
| Otean | | s [35] | 3-wire CH2 User range settings gain value | - | - | System |
| (a) [29] 3-wire CH2 Factory default offset value (H) - - System (a) [30] 3-wire CH2 Factory default gain value (L) - - System (a) [31] 3-wire CH2 Factory default gain value (H) - - System (a) [32] 3-wire CH2 User range settings offset value (L) - - System (a) [33] 3-wire CH2 User range settings gain value (L) - - System (a) [35] 3-wire CH2 User range settings gain value (L) - - System (a) [36] 3-wire CH2 User range settings gain value (H) - - System (a) [36] 3-wire CH2 User range settings gain value (H) - - System (a) [37] 3-wire CH2 User range settings resistance offset value (H) - - System (b) [38] 3-wire CH2 User range settings resistance gain value (H) - - System (b) [39] 3-wire CH2 User range settings resistance gain value (H) - - System (b) [41] 4-wire CH2 Factory default offset value - - System (a) [42] 4-wire CH2 Factory default gain value < | | s [28] | 3-wire CH2 Factory default offset value (L) | _ | | System |
| O64RD • [31] 3-wire CH2 Factory default gain value (H) - - System • • [31] 3-wire CH2 User range settings offset value (L) - - System • • [32] 3-wire CH2 User range settings offset value (H) - - System • • [33] 3-wire CH2 User range settings gain value (L) - - System • • [36] 3-wire CH2 User range settings gain value (L) - - System • • [36] 3-wire CH2 User range settings resistance offset value (L) - - System • • [36] 3-wire CH2 User range settings resistance offset value (L) - - System • • [37] 3-wire CH2 User range settings resistance offset value (H) - - System • • [38] 3-wire CH2 User range settings resistance gain value (L) - - System • • [39] 3-wire CH2 User range settings resistance gain value (H) - - System • • • • • • • • • • • • • • • • • • • | | s [29] | 3-wire CH2 Factory default offset value (H) | | | Cycloni |
| O64RD © [31] 3-wire CH2 Factory default gain value (H) - - System •G [32] 3-wire CH2 User range settings offset value (L) - - System •G [33] 3-wire CH2 User range settings gain value (L) - - System •G [34] 3-wire CH2 User range settings gain value (L) - - System •G [35] 3-wire CH2 User range settings gain value (H) - - System •G [35] 3-wire CH2 User range settings resistance offset value (L) - - System •G [36] 3-wire CH2 User range settings resistance offset value (L) - - System •G [37] 3-wire CH2 User range settings resistance gain value (L) - - System •G [38] 3-wire CH2 User range settings resistance gain value (L) - - System •G [39] 3-wire CH2 Eactory default offset value - - System •G [40] 4-wire CH2 Factory default offset value - - System •G [41] 4-wire CH2 Eactory defau | | \$ [30] | 3-wire CH2 Factory default gain value (L) | | | System |
| Image: Signal of the one | Q64RD | <u></u> জ[31] | 3-wire CH2 Factory default gain value (H) | _ | _ | Oystem |
| S [33] 3-wire CH2 User range settings gain value (H) - - System S [34] 3-wire CH2 User range settings gain value (L) - - System S [35] 3-wire CH2 User range settings gain value (H) - - System S [36] 3-wire CH2 User range settings resistance offset value (L) - - System S [37] 3-wire CH2 User range settings resistance offset value (H) - - System S [38] 3-wire CH2 User range settings resistance gain value (L) - - System S [39] 3-wire CH2 User range settings resistance gain value (L) - - System S [40] 4-wire CH2 User range settings resistance gain value (H) - - System S [41] 4-wire CH2 Factory default offset value - - System S [42] 4-wire CH2 Factory default gain value - - System S [43] 4-wire CH2 Factory default gain value - - System S [41] 4-wire CH2 Factory default gain value - - System S [43] 4-wire CH2 User range settings offset value - | -G | \$ [32] | 3-wire CH2 User range settings offset value (L) | | | Svetom |
| S [35] 3-wire CH2 User range settings gain value (H) - - System S [36] 3-wire CH2 User range settings resistance offset value (L) - - System S [37] 3-wire CH2 User range settings resistance offset value (H) - - System S [38] 3-wire CH2 User range settings resistance gain value (L) - - System S [39] 3-wire CH2 User range settings resistance gain value (H) - - System S [40] 4-wire CH2 Factory default offset value - - System S [41] 4-wire CH2 Factory default offset value - - System S [42] 4-wire CH2 Factory default gain value - - System S [43] 4-wire CH2 Factory default gain value - - System S [43] 4-wire CH2 Factory default gain value - - System S [44] 4-wire CH2 User range settings offset value - - System S [45] 4-wire CH2 User range settings offset value - - System S [45] 4-wire CH2 User range settings offset value - - | | s [33] | 3-wire CH2 User range settings offset value (H) | | - | System |
| (§ [35] 3-wire CH2 User range settings gain value (H) - - System (§ [36] 3-wire CH2 User range settings resistance offset value (L) - - System (§ [37] 3-wire CH2 User range settings resistance offset value (H) - - System (§ [38] 3-wire CH2 User range settings resistance gain value (L) - - System (§ [39] 3-wire CH2 User range settings resistance gain value (H) - - System (§ [40] 4-wire CH2 Factory default offset value - - System (§ [41] 4-wire CH2 Factory default offset value - - System (§ [41] 4-wire CH2 Factory default offset value - - System (§ [42] 4-wire CH2 Factory default gain value - - System (§ [43] 4-wire CH2 Factory default gain value - - System (§ [44] 4-wire CH2 User range settings offset value - - System (§ [45] 4-wire CH2 User range settings offset value - - System (§ [46] 4-wire CH2 User range settings gain value - | | s [34] | 3-wire CH2 User range settings gain value (L) | | | Custom |
| Sigs 3-wire CH2 User range settings resistance offset value (H) - - System Sigs 3-wire CH2 User range settings resistance gain value (L) - - System Sigs 3-wire CH2 User range settings resistance gain value (H) - - System Sigs 3-wire CH2 User range settings resistance gain value (H) - - System Sigs 3-wire CH2 User range settings resistance gain value (H) - - System Sigs 3-wire CH2 Factory default offset value - - System Sigs Sigs 4-wire CH2 Factory default offset value - - System Sigs Sigs 4-wire CH2 Factory default gain value - - System Sigs Sigs 4-wire CH2 Factory default gain value - - System Sigs Sigs 4-wire CH2 Factory default gain value - - System Sigs Sigs 4-wire CH2 User range settings offset value - - System Sigs Sigs 4-wire CH2 User range settings offset value - - System Sigs </td <td></td> <td>\$ [35]</td> <td>3-wire CH2 User range settings gain value (H)</td> <td>-</td> <td>_</td> <td>System</td> | | \$ [35] | 3-wire CH2 User range settings gain value (H) | - | _ | System |
| (§ [37] 3-wire CH2 User range settings resistance offset value (H) - - System (§ [38] 3-wire CH2 User range settings resistance gain value (L) - - System (§ [39] 3-wire CH2 User range settings resistance gain value (H) - - System (§ [39] 3-wire CH2 User range settings resistance gain value (H) - - System (§ [40] 4-wire CH2 Factory default offset value - - System (§ [41] 4-wire CH2 Factory default offset value - - System (§ [42] 4-wire CH2 Factory default gain value - - System (§ [42] 4-wire CH2 Factory default gain value - - System (§ [43] 4-wire CH2 Factory default gain value - - System (§ [44] 4-wire CH2 User range settings offset value - - System (§ [45] 4-wire CH2 User range settings offset value - - System (§ [46] 4-wire CH2 User range settings gain value - - System | S | [36] | 3-wire CH2 User range settings resistance offset value (L) | | | Quarterin |
| Sign 3-wire CH2 User range settings resistance gain value (H) - System Sign 3-wire CH2 User range settings resistance gain value (H) - - System Sign 4-wire CH2 Factory default offset value - - System Sign 4-wire CH2 Factory default offset value - - System Sign 4-wire CH2 Factory default offset value - - System Sign 4-wire CH2 Factory default gain value - - System Sign 4-wire CH2 Factory default gain value - - System Sign 4-wire CH2 Factory default gain value - - System Sign 4-wire CH2 User range settings offset value - - System Sign 4-wire CH2 User range settings offset value - - System Sign 4-wire CH2 User range settings offset value - - System Sign 4-wire CH2 User range settings gain value - - System Sign 4-wire CH2 User range settings gain value - - System | S | [37] | 3-wire CH2 User range settings resistance offset value (H) | - | _ | System |
| (S)[39] 3-wire CH2 User range settings resistance gain value (H) (S)[40] 4-wire CH2 Factory default offset value - - System (S)[41] 4-wire CH2 Factory default offset value - - System (S)[41] 4-wire CH2 Factory default offset value - - System (S)[42] 4-wire CH2 Factory default gain value - - System (S)[42] 4-wire CH2 Factory default gain value - - System (S)[43] 4-wire CH2 Factory default gain value - - System (S)[44] 4-wire CH2 User range settings offset value - - System (S)[44] 4-wire CH2 User range settings offset value - - System (S)[45] 4-wire CH2 User range settings offset value - - System (S)[46] 4-wire CH2 User range settings gain value - - System | S | [38] | 3-wire CH2 User range settings resistance gain value (L) | | | Quarterin |
| Q64RD | <u> </u> | | 3-wire CH2 User range settings resistance gain value (H) | - | - | System |
| Q64RD | | <u>\$</u> [40] | 4-wire CH2 Factory default offset value | _ | - | System |
| Image: Second | | <u>\$</u> [41] | 4-wire CH2 Factory default offset value | _ | _ | System |
| Image: Second | | | 4-wire CH2 Factory default gain value | _ | - | System |
| (a) [44] 4-wire CH2 User range settings offset value - - System (b) [45] 4-wire CH2 User range settings offset value - - System (b) [46] 4-wire CH2 User range settings gain value - - System | | | 4-wire CH2 Factory default gain value | _ | - | System |
| (\$ [45] 4-wire CH2 User range settings offset value - - System (\$ [46] 4-wire CH2 User range settings gain value - - System | Q64RD | | 4-wire CH2 User range settings offset value | _ | _ | System |
| (§ [46]] 4-wire CH2 User range settings gain value – – System | | | 4-wire CH2 User range settings offset value | _ | _ | System |
| | | | 4-wire CH2 User range settings gain value | _ | _ | System |
| | | ⑤[47] | 4-wire CH2 User range settings gain value | _ | _ | System |

Control data of Q64RD/Q64RD-G (2/5)

| Device | | Item | Setting data | Setting range | Setting side |
|--------|----------------|--|--------------|------------------|-----------------|
| | s [40] | 4-wire CH2 Factory default offset value (L) | | Juni ge | |
| | <u></u> | 4-wire CH2 Factory default offset value (H) | - | - | System |
| | <u> </u> | 4-wire CH2 Factory default gain value (L) | | | |
| Q64RD | <u>\$[43]</u> | 4-wire CH2 Factory default gain value (H) | - | _ | System |
| -G | <u>\$</u> [44] | 4-wire CH2 User range settings offset value (L) | | | |
| | <u>\$</u> [45] | 4-wire CH2 User range settings offset value (H) | - | - | System |
| | <u>\$[46]</u> | 4-wire CH2 User range settings gain value (L) | | | |
| | s[47] | 4-wire CH2 User range settings gain value (H) | - | - | System |
| (\$) | [48] | 4-wire CH2 User range settings resistance offset value (L) | | | Quetera |
| \$ | [49] | 4-wire CH2 User range settings resistance offset value (H) | - | _ | System |
| \$ | [50] | 4-wire CH2 User range settings resistance gain value (L) | | | Queters |
| \$ | [51] | 4-wire CH2 User range settings resistance gain value (H) | - | _ | System |
| | \$ [52] | 3-wire CH3 Factory default offset value | - | _ | System |
| | \$ [53] | 3-wire CH3 Factory default offset value | - | _ | System |
| | s [54] | 3-wire CH3 Factory default gain value | - | - | System |
| Q64RD | s [55] | 3-wire CH3 Factory default gain value | _ | - | System |
| Q04KD | s [56] | 3-wire CH3 User range settings offset value | _ | - | System |
| | s[57] | 3-wire CH3 User range settings offset value | - | - | System |
| | s [58] | 3-wire CH3 User range settings gain value | - | - | System |
| | s [59] | 3-wire CH3 User range settings gain value | - | - | System |
| | s [52] | 3-wire CH3 Factory default offset value (L) | | | Sustam |
| | s [53] | 3-wire CH3 Factory default offset value (H) | _ | - | System |
| | \$[54] | 3-wire CH3 Factory default gain value (L) | | | System |
| Q64RD | \$ [55] | 3-wire CH3 Factory default gain value (H) | _ | _ | System |
| -G | <u></u> জ [56] | 3-wire CH3 User range settings offset value (L) | | | System |
| | <u></u> \$[57] | 3-wire CH3 User range settings offset value (H) | - | _ | Oystem |
| | \$ [58] | 3-wire CH3 User range settings gain value (L) | | | System |
| | \$ [59] | 3-wire CH3 User range settings gain value (H) | - | _ | System |
| s | [60] | 3-wire CH3 User range settings resistance offset value (L) | _ | _ | System |
| S | [61] | 3-wire CH3 User range settings resistance offset value (H) | | | oystem |
| S | [62] | 3-wire CH3 User range settings resistance gain value (L) | _ | _ | System |
| S | [63] | 3-wire CH3 User range settings resistance gain value (H) | _ | | Oystern |
| | \$[64] | 4-wire CH3 Factory default offset value | - | - | System |
| | \$ [65] | 4-wire CH3 Factory default offset value | - | - | System |
| | \$ [66] | 4-wire CH3 Factory default gain value | - | - | System |
| Q64RD | s [67] | 4-wire CH3 Factory default gain value | _ | _ | System |
| | \$ [68] | 4-wire CH3 User range settings offset value | - | _ | System |
| | \$ [69] | 4-wire CH3 User range settings offset value | _ | | System |
| | s [70] | 4-wire CH3 User range settings gain value | - | _ | System |
| | <u></u> (জ[71] | 4-wire CH3 User range settings gain value | - | - | System |

Control data of Q64RD/Q64RD-G (3/5)

| (| s [64] s [65] | 4-wire CH3 Factory default offset value (L) | | range | side |
|----------|------------------|--|---|-------|----------|
| (| | | | | |
| | | 4-wire CH3 Factory default offset value (H) | - | - | System |
| C | | 4-wire CH3 Factory default gain value (L) | | | |
| 00400 | § [66] | 4-wire CH3 Factory default gain value (L) 4-wire CH3 Factory default gain value (H) | - | - | System |
| | § [67] | 4-wire CH3 User range settings offset value (L) | | | |
| | § [68] | 4-wire CH3 User range settings offset value (L) 4-wire CH3 User range settings offset value (H) | - | - | System |
| | s [69] | 4-wire CH3 User range settings gain value (L) | | | |
| | \$ [70] | 4-wire CH3 User range settings gain value (L) 4-wire CH3 User range settings gain value (H) | - | - | System |
| | ତ [71] | 4-wire CH3 User range settings resistance offset value (L) | | | |
| <u> </u> | | 4-wire CH3 User range settings resistance offset value (L) 4-wire CH3 User range settings resistance offset value (H) | - | - | System |
| © [73] | | 4-wire CH3 User range settings resistance diset value (H) 4-wire CH3 User range settings resistance gain value (L) | | | |
| © [74] | | | - | - | System |
| © [75] | - | 4-wire CH3 User range settings resistance gain value (H) | | | Custom |
| | s [76] | 3-wire CH4 Factory default offset value | - | - | System |
| | \$[77] | 3-wire CH4 Factory default offset value | - | - | System |
| | \$ [78] | 3-wire CH4 Factory default gain value | - | - | System |
| Q64RD | s [79] | 3-wire CH4 Factory default gain value | - | - | System |
| | \$ [80] | 3-wire CH4 User range settings offset value | - | - | System |
| | s [81] | 3-wire CH4 User range settings offset value | - | - | System |
| (| s [82] | 3-wire CH4 User range settings gain value | - | - | System |
| | \$ [83] | 3-wire CH4 User range settings gain value | - | - | System |
| | \$[76] | 3-wire CH4 Factory default offset value (L) | _ | _ | System |
| | s [77] | 3-wire CH4 Factory default offset value (H) | | | |
| | \$ [78] | 3-wire CH4 Factory default gain value (L) | _ | _ | System |
| | s [79] | 3-wire CH4 Factory default gain value (H) | | | - , |
| -G | \$[80] | 3-wire CH4 User range settings offset value (L) | _ | _ | System |
| | \$[81] | 3-wire CH4 User range settings offset value (H) | | | e yete |
| | \$[82] | 3-wire CH4 User range settings gain value (L) | _ | _ | System |
| | \$ [83] | 3-wire CH4 User range settings gain value (H) | | | Gyötölli |
| s [84] |] | 3-wire CH4 User range settings resistance offset value (L) | _ | _ | System |
| s [85] |] | 3-wire CH4 User range settings resistance offset value (H) | | | Gyötölli |
| S[86] |] | 3-wire CH4 User range settings resistance gain value (L) | | | System |
| S[87] |] | 3-wire CH4 User range settings resistance gain value (H) | _ | | Oystelli |
| (| \$[88] | 4-wire CH4 Factory default offset value | - | - | System |
| (| s [89] | 4-wire CH4 Factory default offset value | - | - | System |
| | s [90] | 4-wire CH4 Factory default gain value | - | - | System |
| | s [91] | 4-wire CH4 Factory default gain value | - | - | System |
| Q64RD | s [92] | 4-wire CH4 User range settings offset value | - | - | System |
| (| s [93] | 4-wire CH4 User range settings offset value | - | - | System |
| (| s [94] | 4-wire CH4 User range settings gain value | - | - | System |
| (| \$ [95] | 4-wire CH4 User range settings gain value | _ | - | System |

Control data of Q64RD/Q64RD-G (4/5)

| Device | | Item | Setting data | Setting range | Setting side | |
|--------|---------------|--|--------------|---------------|-----------------|--|
| | S[88] | 4-wire CH4 Factory default offset value (L) | _ | _ | System | |
| | s [89] | 4-wire CH4 Factory default offset value (H) | | | System | |
| | s [90] | 4-wire CH4 Factory default gain value (L) | _ | _ | System | |
| Q64RD | <u>ি</u> [91] | 4-wire CH4 Factory default gain value (H) | | | System | |
| -G | s [92] | 4-wire CH4 User range settings offset value (L) | _ | _ | System | |
| | s [93] | 4-wire CH4 User range settings offset value (H) | | | Cystem | |
| | s [94] | 4-wire CH4 User range settings gain value (L) | _ | - | System | |
| | s [95] | 4-wire CH4 User range settings gain value (H) | | | Cystem | |
| S | [96] | 4-wire CH4 User range settings resistance offset value (L) | _ | _ | System | |
| \$ | [97] | 4-wire CH4 User range settings resistance offset value (H) | | | e y e tom | |
| S | [98] | 4-wire CH4 User range settings resistance gain value (L) | _ | _ | System | |
| S | [99] | 4-wire CH4 User range settings resistance gain value (H) | | | Gystern | |

Control data of Q64RD/Q64RD-G (5/5)

*1 : Setting is unnecessary. If setting is configured, offset/gain setting value is not read properly.

(11) Q64TD/Q64TDV-GH

| Device | Item | Setting data | Setting range | Setting side |
|---------------|--|--|---------------|--------------|
| § [0] | System area | _ | - | _ |
| 6[1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| §[2] | System area | | | |
| ঙ [3] | | _ | _ | - |
| ⑤ [4] | CH1 Factory default offset value | - | - | System |
| ⑤ [5] | CH1 Factory default gain value | - | - | System |
| ⑤[6] | CH1 User range settings offset value | - | - | System |
| ⑤ [7] | CH1 User range settings gain value | - | - | System |
| ⑤ [8] | CH1 User range settings thermal EMF offset value (L) | _ | _ | System |
| s [9] | CH1 User range settings thermal EMF offset value (H) | | | Gyöteini |
| s [10] | CH1 User range settings thermal EMF gain value (L) | _ | _ | System |
| s[11] | CH1 User range settings thermal EMF gain value (H) | | | eyetein |
| s[12] | CH2 Factory default offset value | - | - | System |
| ি [13] | CH2 Factory default gain value | - | - | System |
| <u>ি</u> [14] | CH2 User range settings offset value | _ | - | System |
| <u>জ</u> [15] | CH2 User range settings gain value | _ | - | System |
| <u></u> জ[16] | CH2 User range settings thermal EMF offset value (L) | _ | _ | System |
| <u></u> জ[17] | CH2 User range settings thermal EMF offset value (H) | | | |
| <u></u> জ[18] | CH2 User range settings thermal EMF gain value (L) | _ | _ | System |
| <u></u> §[19] | CH2 User range settings thermal EMF gain value (H) | | | ., |
| s [20] | CH3 Factory default offset value | _ | - | System |
| S[21] | CH3 Factory default gain value | _ | - | System |
| S [22] | CH3 User range settings offset value | _ | - | System |
| s [23] | CH3 User range settings gain value | _ | - | System |
| s [24] | CH3 User range settings thermal EMF offset value (L) | - | - | System |
| s [25] | CH3 User range settings thermal EMF offset value (H) | | | - |
| \$[26] | CH3 User range settings thermal EMF gain value (L) | _ | - | System |
| <u></u> জ[27] | CH3 User range settings thermal EMF gain value (H) | | | |
| \$ [28] | CH4 Factory default offset value | _ | - | System |
| \$ [29] | CH4 Factory default gain value | _ | - | System |
| s [30] | CH4 User range settings offset value | - | - | System |
| s [31] | CH4 User range settings gain value | - | - | System |
| \$[32] | CH4 User range settings thermal EMF offset value (L) | _ | - | System |
| \$ [33] | CH4 User range settings thermal EMF offset value (H) | | | |
| s [34] | CH4 User range settings thermal EMF gain value (L) | _ | - | System |
| s [35] | CH4 User range settings thermal EMF gain value (H) | | | - |

(12) Q68TD-G-H02(H01)

Control data of Q68TD-G-H02(H01) (1/2)

| | Item | Setting data | Setting range | Setting side |
|--------------------------|--|---|---------------|--------------|
| ⑤ [0] | System area | - | - | - |
| ঙ[1] | Completion status | The instruction completion status is stored. 0 :Normal completion Other than 0 :Error completion (error code) | _ | System |
| s[2] | System area | _ | _ | _ |
| \$[3] | oystem area | _ | _ | _ |
| ঙ[4] | CH1 Factory default offset value | - | - | System |
| s [5] | CH1 Factory default gain value | - | - | System |
| ⑤[6] | CH1 User range settings offset value | - | - | System |
| ঙ[7] | CH1 User range settings gain value | _ | - | System |
| \$[8] | CH1 User range settings thermal EMF offset value (L) | | | System |
| \$[9] | CH1 User range settings thermal EMF offset value (H) | _ | - | System |
| s [10] | CH1 User range settings thermal EMF gain value (L) | | | System |
| s[11] | CH1 User range settings thermal EMF gain value (H) | _ | _ | System |
| s [12] | CH2 Factory default offset value | _ | - | System |
| <u> </u> ⁽¹³⁾ | CH2 Factory default gain value | _ | - | System |
| <u></u> জ[14] | CH2 User range settings offset value | _ | _ | System |
| <u></u> জ [15] | CH2 User range settings gain value | _ | _ | System |
| s [16] | CH2 User range settings thermal EMF offset value (L) | | | |
| s [17] | CH2 User range settings thermal EMF offset value (H) | - | - | System |
| s [18] | CH2 User range settings thermal EMF gain value (L) | | | |
| \$[19] | CH2 User range settings thermal EMF gain value (H) | _ | - | System |
| \$[20] | CH3 Factory default offset value | _ | - | System |
| \$[21] | CH3 Factory default gain value | _ | - | System |
| \$[22] | CH3 User range settings offset value | _ | - | System |
| s [23] | CH3 User range settings gain value | _ | - | System |
| \$[24] | CH3 User range settings thermal EMF offset value (L) | | | _ |
| s [25] | CH3 User range settings thermal EMF offset value (H) | _ | - | System |
| \$ [26] | CH3 User range settings thermal EMF gain value (L) | | | _ |
| | CH3 User range settings thermal EMF gain value (H) | _ | - | System |
| \$ [28] | CH4 Factory default offset value | _ | - | System |
| \$ [29] | CH4 Factory default gain value | _ | - | System |
| s [30] | CH4 User range settings offset value | _ | - | System |
| s[31] | CH4 User range settings gain value | _ | _ | System |
| \$[32] | CH4 User range settings thermal EMF offset value (L) | | | |
| ⑤ [33] | CH4 User range settings thermal EMF offset value (H) | - | - | System |
| § [34] | CH4 User range settings thermal EMF gain value (L) | | | |
| ⑤ [35] | CH4 User range settings thermal EMF gain value (H) | - | - | System |
| ⑤[36] | CH5 Factory default offset value | _ | - | System |
| ⑤[37] | CH5 Factory default gain value | _ | _ | System |
| 010.1 | CH5 User range settings offset value | | _ | System |
| \$ [38] | CHO User range settings onset value | | | |

| Device | Item | Setting data | Setting range | Setting side |
|----------------|--|--------------|---------------|--------------|
| <u>s</u> [40] | CH5 User range settings thermal EMF offset value (L) | | | Svotom |
| s [41] | CH5 User range settings thermal EMF offset value (H) | _ | - | System |
| s [42] | CH5 User range settings thermal EMF gain value (L) | | | System |
| \$ [43] | CH5 User range settings thermal EMF gain value (H) | - | _ | System |
| \$[44] | CH6 Factory default offset value | _ | - | System |
| \$ [45] | CH6 Factory default gain value | _ | - | System |
| \$[46] | CH6 User range settings offset value | _ | - | System |
| ঙ [47] | CH6 User range settings gain value | _ | - | System |
| \$[48] | CH6 User range settings thermal EMF offset value (L) | | | System |
| ঙ [49] | CH6 User range settings thermal EMF offset value (H) | _ | _ | System |
| ঙ [50] | CH6 User range settings thermal EMF gain value (L) | | | System |
| ঙ [51] | CH6 User range settings thermal EMF gain value (H) | _ | _ | Oystern |
| ঙ [52] | CH7 Factory default offset value | _ | - | System |
| s [53] | CH7 Factory default gain value | _ | - | System |
| \$[54] | CH7 User range settings offset value | _ | - | System |
| ঙ [55] | CH7 User range settings gain value | _ | - | System |
| s [56] | CH7 User range settings thermal EMF offset value (L) | _ | _ | System |
| <u></u> \$[57] | CH7 User range settings thermal EMF offset value (H) | | | Gystern |
| \$ [58] | CH7 User range settings thermal EMF gain value (L) | _ | _ | System |
| \$ [59] | CH7 User range settings thermal EMF gain value (H) | _ | | Oystern |
| \$[60] | CH8 Factory default offset value | - | - | System |
| s[61] | CH8 Factory default gain value | - | - | System |
| s [62] | CH8 User range settings offset value | - | - | System |
| § [63] | CH8 User range settings gain value | - | - | System |
| § [64] | CH8 User range settings thermal EMF offset value (L) | | | System |
| s [65] | CH8 User range settings thermal EMF offset value (H) | | | Oystern |
| \$[66] | CH8 User range settings thermal EMF gain value (L) | _ | | System |
| s [67] | CH8 User range settings thermal EMF gain value (H) | | _ | Gystern |

Control data of Q68TD-G-H02(H01) (2/2)

(13) Q68RD3-G

Control data of Q68RD3-G (1/2)

| Device | Item | Setting data | Setting range | Setting side |
|---------------|---|--|---------------|--------------|
| ⑤ [0] | System area | _ | - | - |
| ঙ[1] | Completion status | The instruction completion status is stored. 0 :Normal completion Other than 0 :Error completion (error code) | _ | System |
| ঙ[2] ঙ[3] | System area | - | - | - |
| ⑤ [4] | CH1 Factory default offset value | _ | - | System |
| ⑤ [5] | CH1 Factory default gain value | _ | - | System |
| ⑤[6] | CH1 User range settings offset value | _ | - | System |
| ⑤ [7] | CH1 User range settings gain value | _ | - | System |
| ⑤[8] | CH1 User range settings resistance offset value (L) | | | Questions |
| s [9] | CH1 User range settings resistance offset value (H) | _ | _ | System |
| s[10] | CH1 User range settings resistance gain value (L) | | | Custom |
| <u></u> জ[11] | CH1 User range settings resistance gain value (H) | _ | _ | System |
| s [12] | CH2 Factory default offset value | _ | - | System |
| s [13] | CH2 Factory default gain value | _ | - | System |
| s[14] | CH2 User range settings offset value | _ | - | System |
| <u> </u> (15] | CH2 User range settings gain value | _ | - | System |
| s[16] | CH2 User range settings resistance offset value (L) | | | System |
| s [17] | CH2 User range settings resistance offset value (H) | _ | _ | System |
| s [18] | CH2 User range settings resistance gain value (L) | | | Sustam |
| s [19] | CH2 User range settings resistance gain value (H) | | _ | System |
| s [20] | CH3 Factory default offset value | - | - | System |
| ি [21] | CH3 Factory default gain value | - | - | System |
| \$[22] | CH3 User range settings offset value | _ | - | System |
| \$[23] | CH3 User range settings gain value | - | - | System |
| জ [24] | CH3 User range settings resistance offset value (L) | | | System |
| \$[25] | CH3 User range settings resistance offset value (H) | | _ | Oystern |
| \$[26] | CH3 User range settings resistance gain value (L) | _ | _ | System |
| \$[27] | CH3 User range settings resistance gain value (H) | | | Gyötenn |
| s [28] | CH4 Factory default offset value | - | - | System |
| s [29] | CH4 Factory default gain value | - | - | System |
| <u> </u> | CH4 User range settings offset value | - | - | System |
| \$[31] | CH4 User range settings gain value | - | - | System |
| \$[32] | CH4 User range settings resistance offset value (L) | | | System |
| \$ [33] | CH4 User range settings resistance offset value (H) | | | Gystelli |
| s [34] | CH4 User range settings resistance gain value (L) | | _ | System |
| <u> </u> | CH4 User range settings resistance gain value (H) | | | Gystelli |
| \$[36] | CH5 Factory default offset value | - | - | System |
| \$[37] | CH5 Factory default gain value | - | _ | System |
| s [38] | CH5 User range settings offset value | _ | - | System |

Control data of Q68RD3-G (2/2)

| Device | Item | Setting data | Setting range | Setting side |
|---------------|---|--------------|---------------|--------------|
| s [39] | CH5 User range settings gain value | - | _ | System |
| s [40] | CH5 User range settings resistance offset value (L) | | | System |
| s[41] | CH5 User range settings resistance offset value (H) | | | Gystein |
| s [42] | CH5 User range settings resistance gain value (L) | _ | _ | System |
| s [43] | CH5 User range settings resistance gain value (H) | _ | | oystem |
| ⑤ [44] | CH6 Factory default offset value | - | - | System |
| s [45] | CH6 Factory default gain value | - | - | System |
| ⑤ [46] | CH6 User range settings offset value | - | - | System |
| ⑤ [47] | CH6 User range settings gain value | - | - | System |
| ⑤ [48] | CH6 User range settings resistance offset value (L) | _ | _ | System |
| ⑤ [49] | CH6 User range settings resistance offset value (H) | | | Oystelli |
| s [50] | CH6 User range settings resistance gain value (L) | | | System |
| s [51] | CH6 User range settings resistance gain value (H) | _ | - | System |
| s [52] | CH7 Factory default offset value | - | - | System |
| s [53] | CH7 Factory default gain value | _ | - | System |
| s [54] | CH7 User range settings offset value | _ | - | System |
| S[55] | CH7 User range settings gain value | _ | - | System |
| \$[56] | CH7 User range settings resistance offset value (L) | | | System |
| \$[57] | CH7 User range settings resistance offset value (H) | _ | - | System |
| s [58] | CH7 User range settings resistance gain value (L) | | | System |
| \$[59] | CH7 User range settings resistance gain value (H) | _ | - | System |
| s [60] | CH8 Factory default offset value | _ | - | System |
| <u></u> §[61] | CH8 Factory default gain value | _ | - | System |
| s [62] | CH8 User range settings offset value | - | - | System |
| s [63] | CH8 User range settings gain value | _ | - | System |
| s[64] | CH8 User range settings resistance offset value (L) | | | Sustam |
| s [65] | CH8 User range settings resistance offset value (H) | | _ | System |
| s [66] | CH8 User range settings resistance gain value (L) | | | Sustam |
| s[67] | CH8 User range settings resistance gain value (H) | - | - | System |

(14) Q61LD^{*1}

Control data of Q61LD (1/2)

| Device | Item | Setting data | Setting range | Setting side |
|--------------------------|--|--|---------------|--------------|
| s [0] | System area | - | - | System |
| s [1] | Completion status | The instruction completion status is stored. 0 :Normal completion Other than 0 :Error completion (error code) | _ | System |
| \$[2] \$[3] | - System area | - | _ | System |
| ⑤ [4] | Load cell rated capacity (L) | - | - | System |
| s [5] | Load cell rated capacity (H) | - | - | System |
| ⑤[6] | Load cell rated output | - | - | System |
| ⑤ [7] | Number of load cells in connection | - | - | System |
| ⑤[8] | Zero offset | - | - | System |
| s [9] | System area | - | - | System |
| s [10] | Maximum weighing capacity setting (L) | - | - | System |
| <u>ঙ</u> [11] | Maximum weighing capacity setting (H) | - | - | System |
| s [12] | Minimum division | - | - | System |
| <u> </u> [13] | Decimal point position | - | - | System |
| s[14] | Unit | - | - | System |
| s [15] | System area | - | - | System |
| s [16] | Standard weight setting (L) | - | - | System |
| s[17] | Standard weight setting (H) | - | - | System |
| s[18] | Installation site gravitational acceleration (L) | - | - | System |
| s [19] | Installation site gravitational acceleration (H) | - | _ | System |
| s [20] | Calibration site gravitational acceleration (L) | - | - | System |
| s [21] | Calibration site gravitational acceleration (H) | - | - | System |
| s [22] | Digital output zero correction value (L) | - | - | System |
| s [23] | Digital output zero correction value (H) | - | - | System |
| s [24] | Digital output span correction value (L) | - | - | System |
| s [25] | Digital output span correction value (H) | - | - | System |
| \$ [26] to \$ [33] | System area | - | - | System |
| ⑤ [34] | Instrumentation amplifier gain setting | _ | - | System |
| s [35] | A/D converter gain setting | _ | _ | System |
| \$[36] | Zero offset output value (L) | _ | - | System |
| ⑤ [37] | Zero offset output value (H) | _ | - | System |
| \$ [38] | Two-point zero calibration value (L) | _ | - | System |
| ⑤ [39] | Two-point zero calibration value (H) | _ | - | System |
| <u>\$</u> [40] | Two-point span calibration value (L) | _ | - | System |
| <u> </u> | Two-point span calibration value (H) | _ | - | System |
| S [42] to S [53] | System area | - | - | System |
| ⑤[54] | 1.0mV/V zero calibration value (L) | _ | _ | System |
| [51] [55] | 1.0mV/V zero calibration value (H) | _ | _ | System |

Control data of Q61LD (2/2)

| Device | Item | Setting data | Setting range | Setting side |
|----------------|------------------------------------|--------------|---------------|--------------|
| s [56] | 1.0mV/V span calibration value (L) | - | - | System |
| \$ [57] | 1.0mV/V span calibration value (H) | _ | - | System |
| s [58] | 2.0mV/V zero calibration value (L) | _ | - | System |
| \$[59] | 2.0mV/V zero calibration value (H) | - | - | System |
| \$[60] | 2.0mV/V span calibration value (L) | _ | - | System |
| s [61] | 2.0mV/V span calibration value (H) | _ | - | System |
| \$[62] | 3.0mV/V zero calibration value (L) | - | - | System |
| \$[63] | 3.0mV/V zero calibration value (H) | _ | - | System |
| \$[64] | 3.0mV/V span calibration value (L) | _ | - | System |
| s[65] | 3.0mV/V span calibration value (H) | _ | - | System |
| s [66] | | | | |
| to | System area | _ | - | System |
| s [85] | | | | |

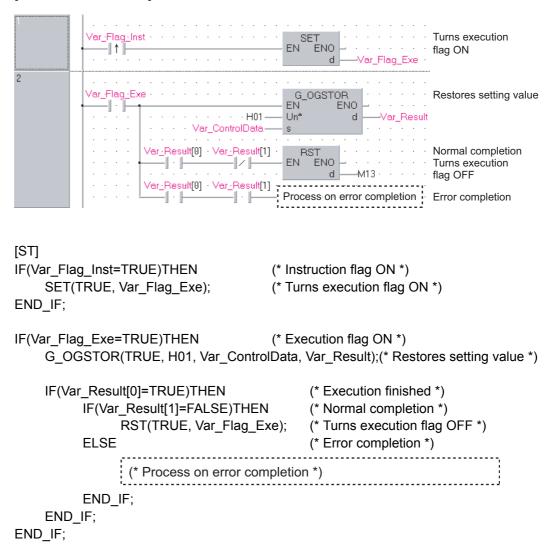
*1 : Setting is unnecessary. If setting is configured, offset/gain setting value is not read properly.

(15) L60AD4/L60DA4

| Device | Item | Setting data | Setting range | Setting side |
|---------------|---|---|-------------------|--------------|
| ⑤ [0] | System area | - | - | - |
| <u></u> জ[1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | - | System |
| ঙ[2] | Stored data type setting | The value set for pass data classification setting (s) [2] by the OGLOAD instruction is stored. 0: Voltage specified 1: Current specified b15 b4 b3 b2 b1 b0 0 to 0 CH4CH3CH2CH1 | 0000н to 000Fн | System |
| \$[3] | System area | - | - | - |
| \$[4] | CH1 Industrial shipment settings offset value | - | - | System |
| \$[5] | CH1 Industrial shipment settings gain value | _ | - | System |
| ⑤[6] | CH2 Industrial shipment settings offset value | _ | - | System |
| ⑤ [7] | CH2 Industrial shipment settings gain value | _ | - | System |
| (8) | CH3 Industrial shipment settings offset value | _ | - | System |
| s[9] | CH3 Industrial shipment settings gain value | - | - | System |
| s[10] | CH4 Industrial shipment settings offset value | - | - | System |
| s[11] | CH4 Industrial shipment settings gain value | - | - | System |
| s[12] | CH1 User range settings offset value | - | - | System |
| <u></u> জ[13] | CH1 User range settings gain value | _ | - | System |
| s[14] | CH2 User range settings offset value | _ | - | System |
| s [15] | CH2 User range settings gain value | _ | - | System |
| s[16] | CH3 User range settings offset value | _ | - | System |
| s[17] | CH3 User range settings gain value | _ | - | System |
| s [18] | CH4 User range settings offset value | _ | - | System |
| s [19] | CH4 User range settings gain value | _ | - | System |

Program Example

The following program restores the offset/gain setting value to the A/D converter module mounted on the I/O numbers from X/Y10 to X/Y1F when the flag turns ON.

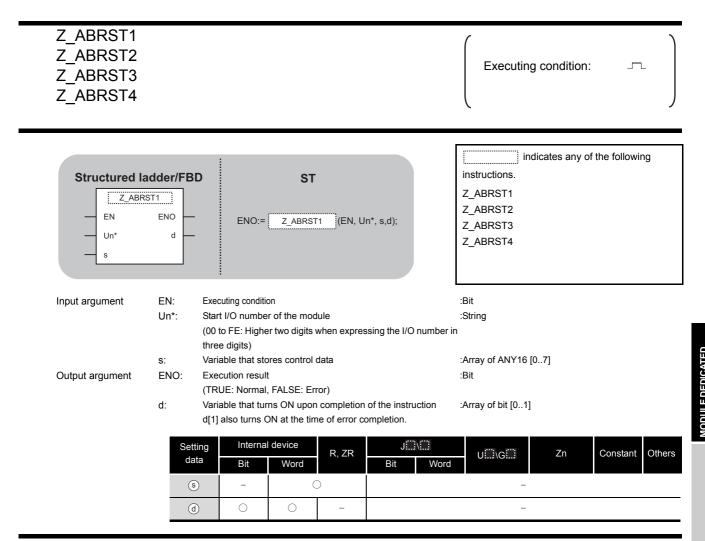


[Structured ladder/FBD]

5.2 Positioning Instruction

5.2.1 ABRST instruction

Z_ABRST1



Grant Function

This instruction restores the absolute position of the specified axis. (Refer to the following)

- Z_ABRST1: Axis 1
- Z_ABRST2: Axis 2
- Z_ABRST3: Axis 3
- Z_ABRST4: Axis 4

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|----------------------|-------------------------------------|---|-------------------------------|-----------------|
| s [0] | System area | - | - | - |
| <u></u> ۱] | Completion status | The instruction completion status is stored.0 : Normal completionOther than 0 : Error completion (error code) | _ | System |
| ঙ [2] | Receive signal from servo amplifier | Write the following signal status read from the servo amplifier to the input module. b0: ABS data bit0 b1: ABS data bit1 b2: Send data READY flag | b0: 0/1 b1: 0/1 b2: 0/1 | User |
| ঙ [3] | Send signal to servo amplifier | The ON/OFF status of the following data, that are calculated by the dedicated instructions on the basis of "receive signal from servo amplifier" and output to the amplifier, are stored. • b0: Servo ON • b1: ABS transfer mode • b2: ABS request flag | _ | System |
| ঙ [4] | Status | Communication status with the servo amplifier • 0 : Communication completed (Set by the user at the start of communication) • Other than 0: During communication (Stored by the system.) | 0 | User/ System |
| \$[5] to \$[7] | System area | - | _ | - |

Program Example

The following program restores the absolute position of the axis 1.

The devices from X47 to X49 and from Y50 to Y52 are used for the communication with the servo amplifier.

- X47: ABS data bit0
- X48: ABS data bit1
- X49: Send data READY flag
- Y50: Servo ON
- Y51: ABS transfer mode
- Y52: ABS request flag

[Structured ladder/FBD]

| 1 | SM400 · · · · · · · · · · · · · · · · · · | Absolute position restoration pulse |
|---|--|--|
| 2 | Ver_Flag_Inst X0 EN EN EN EN EN EN EN EN G Var_Flag_Mem G Var_ControlData[4] | Turns absolute position restoration memory ON Clears completion status |
| 3 | Var_Result[0] · Var_Result[1] · Var_ControlData[3] · S · d · K1Y50 · · · · · · · · · · · · · · · · · · · | Turns the servo ON with the data to be sent to the servo amplifier |
| | · Var_Result[1] · · · · · · · · · · · · · · · · · · · | Sets completion status to error code |
| | AND= EN EN | Turns absolute position restoration memory OFF |
| 4 | Var_Flag_Mem··X47·····BSET BET BEN ENO n d Var_ControlData[2] | Sets ABS data Sets ABS data in data b0 received from the servo |
| | | Sets ABS data in data b1 |
| | X49 X49 EN ENO N ControlData[2] | Sets send data ready flag in data b2 |
| | Z_ABRST1 EN ENO Un d Var_ControlData—s | Restores absolute position |

Z_ABRST1

[ST] PLS(SM400, Var_Flag_Inst); (* Absolute position restoration pulse *) IF((Var_Flag_Inst=TRUE) & (X0=FALSE))THEN SET(TRUE, Var Flag Mem); (* Turns absolute position restoration memory ON *) (* Clears completion status *) MOV(TRUE, 0, Var_ControlData[4]); END_IF; IF(Var_Result[0]=TRUE)THEN (* Execution finished *) IF(Var_Result[1]=FALSE)THEN (* Normal completion *) MOV(TRUE, Var ControlData[3], K1Y50); (* Turns the servo ON with the data to be sent to the servo amplifier *) ELSE (* Error completion *) MOV(TRUE, Var_ControlData[4], Var_ErrorCode); (* Sets completion status to error code *) END_IF; IF(Var_ControlData[4]=0)THEN RST(TRUE, Var_Flag_Mem); (* Turns absolute position restoration memory OFF *) END IF; END IF; IF(Var_Flag_Mem=TRUE)THEN (* absolute position restoration memory ON *) (* Sets ABS data *) BSET(X47, 0, Var_ControlData[2]); (* Sets ABS data in data b0 received from the servo *) BSET(X48, 1, Var ControlData[2]); (* Sets ABS data in data b1 received from the servo *) BSET(X49, 2, Var_ControlData[2]); (* Sets send data ready flag in data b2 received from the servo *) Z_ABRST1(TRUE, "00", Var_ControlData, Var_Result); (* Restores absolute position *)

END_IF;

5.2.2 PSTRT instruction

ZP_PSTRT1

| ZP_PSTRT1 ZP_PSTRT2 ZP_PSTRT3 ZP_PSTRT4 | | | Executing condition : |
|--|--|--|---|
| Structured I ZP_PS EN Un* s | RT1 | ST ZP_PSTRT1(EN, Un*, s,d | instructions. ZP_PSTRT1 ZP_PSTRT2 ZP_PSTRT3 ZP_PSTRT4 |
| Input argument Output argument | three digits) s: Variable that stores ENO: Execution result (TRUE: Normal, FA | the module vo digits when expressing th control data | :Array of ANY16 [02] :Bit |
| | d[1] also turns ON a | at the time of error complet | on. J⊡\⊡ U⊡\G⊡ Zn Constant Others |

Grand Function

This instruction starts positioning of the specified axis. (Refer to the following.)

- ZP_PSTRT1: Axis 1
- ZP_PSTRT2: Axis 2
- ZP_PSTRT3: Axis 3
- ZP_PSTRT4: Axis 4

5

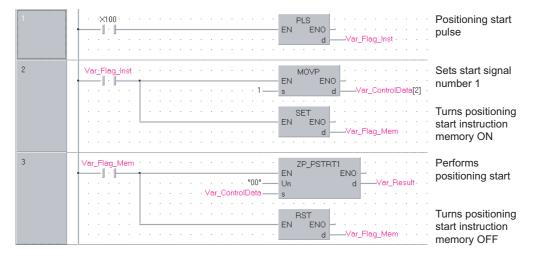
E DEDICATED

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|--------|-------------------|--|--|--------------|
| s [0] | System area | - | - | - |
| s[1] | Completion status | The instruction completion status is stored. • 0 : Normal completion • Other than 0 : Error completion (error code) | _ | System |
| (٤ [2] | Start No. | Specify the following data number to be started by the PSTRT [] instruction. 1 to 600 : Positioning data number 7000 to 7004: Block start 9001 : Machine OPR 9002 : Fast OPR 9003 : Current value change 9004 : Multiple axes concurrent start | 1 to 600, 7000 to 7004, 9001 to 9004 | User |

Program Example

The following program executes the positioning start of the positioning data number 1 when X100 turns ON.



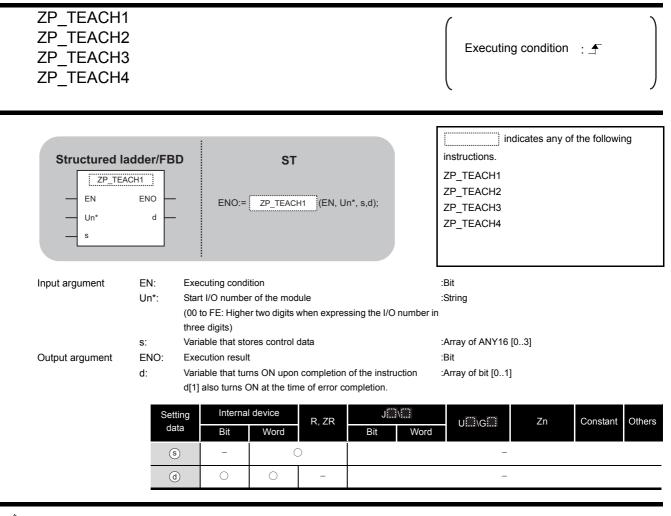
[Structured ladder/FBD]

[ST]

PLS(X100, Var_Flag_Inst); (* Positioning start pulse *) IF(Var_Flag_Inst=TRUE)THEN MOVP(TRUE, 1, Var_ControlData[2]);(* Sets start signal number 1 *) SET(TRUE, Var_Flag_Mem); (* Turns positioning start instruction memory ON *) END_IF; IF(Var_Flag_Mem=TRUE)THEN (* Positioning start instruction memory ON *) ZP_PSTRT1(TRUE, "00", Var_ControlData, Var_Result); (* Performs positioning start *) RST(TRUE, Var_Flag_Mem); (* Turns positioning start instruction memory OFF *) END_IF;

5.2.3 TEACH instruction

ZP_TEACH1



Grant Function

This instruction performs teaching for the specified axis. (Refer to the following)

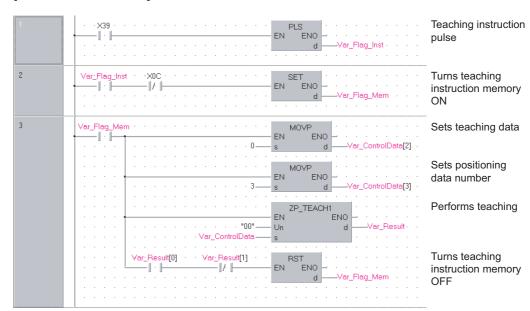
- ZP_TEACH1: Axis 1
- ZP_TEACH2: Axis 2
- ZP_TEACH3: Axis 3
- ZP_TEACH4: Axis 4

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|--------------|-------------------------|---|---------------|--------------|
| ⑤ [0] | System area | _ | _ | - |
| ঙ[1] | Completion status | The instruction completion status is stored. • 0 : Normal completion • Other than 0 : Error completion (error code) | - | System |
| \$[2] | Teaching data selection | Set the address (positioning address/circular address) to which the current feed value is written. 0: Write the current feed value to the positioning address 1: Write the current feed value to the circular address | 0,1 | User |
| s [3] | Positioning data No. | Set the positioning data number for which teaching is performed. | 1 to 600 | User |

Program Example

The following program performs teaching for the positioning data number 3 of the axis 1 when X39 turns ON.



[Structured ladder/FBD]

[ST]

PLS(X39, Var_Flag_Inst);

(* Teaching instruction pulse *)

```
IF((Var_Flag_Inst=TRUE)&(X0C=FALSE))THEN
```

SET(TRUE, Var_Flag_Mem); (* Turns teaching instruction memory ON *) END_IF;

IF(Var_Flag_Mem=TRUE)THEN (* Teaching instruction memory ON *) MOVP(TRUE, H0, Var_ControlData[2]); (* Sets teaching data *) MOVP(TRUE, K3, Var_ControlData[3]); (* Sets positioning data number *)

ZP_TEACH1(TRUE, "00", Var_ControlData, Var_Result);

(* Performs teaching *)

IF((Var_Result[0]=TRUE)&(Var_Result[1]=FALSE))THEN RST(TRUE, Var_Flag_Mem); (* Turns teaching instruction memory OFF *) END_IF; END_IF;

5.2.4 PFWRT instruction

ZP_PFWRT

| ZP_PFWRT | | Executing condition : |
|-----------------------------------|--|--|
| Structured I | |); |
| Input argument Output argument | EN: Executing condition Un*: Start I/O number of the module (00 to FE: Higher two digits when expressing the three digits) s: Variable that stores control data ENO: Execution result d: Variable that turns ON upon completion of the d[1] also turns ON at the time of error completed | :Array of ANY16 [01] :Bit instruction :Array of bit [01] |
| | Setting dataInternal device BitR, ZRSiImage: Image: | J∭\() it Word U∭\G∭ Zn Constant Others - |

Grant Function

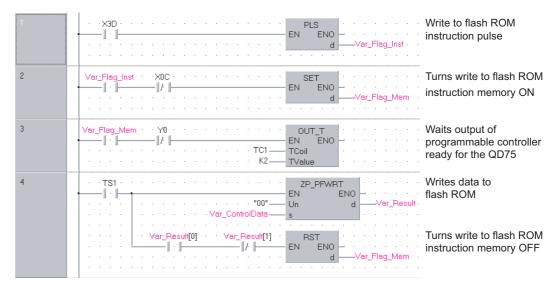
This instruction writes the QD75 parameters, positioning data, and block start data to the flash ROM.

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|--------------|-------------------|--|---------------|--------------|
| ⑤[0] | System area | _ | _ | - |
| | | The instruction completion status is stored. | | |
| ⑤ [1] | Completion status | O : Normal completion | - | System |
| | | Other than 0 : Error completion (error code) | | |

Program Example

The following program writes the parameters, positioning data, and block start data stored in buffer memory to the flash ROM when X3D turns ON.



[Structured ladder/FBD]

[ST]

PLS(X3D, Var_Flag_Inst); (* Write to flash ROM instruction pulse *) IF((Var_Flag_Inst=TRUE)&(X0C=FALSE))THEN

SET(TRUE, Var_Flag_Mem); (* Turns write to flash ROM instruction memory ON *) END_IF;

IF((Var_Flag_Mem=TRUE)&(Y0=FALSE))THEN OUT T(TRUE, TC1, 2);

(* Waits output of programmable controller ready for the QD75 *)

END_IF;

```
IF(TS1=TRUE)THEN (* Write to flash ROM instruction memory ON *)
ZP_PFWRT(TRUE, "00", Var_ControlData, Var_Result);
(* Writes data to flash ROM *)
IF((Var_Result[0]=TRUE)&(Var_Result[1]=FALSE))THEN
```

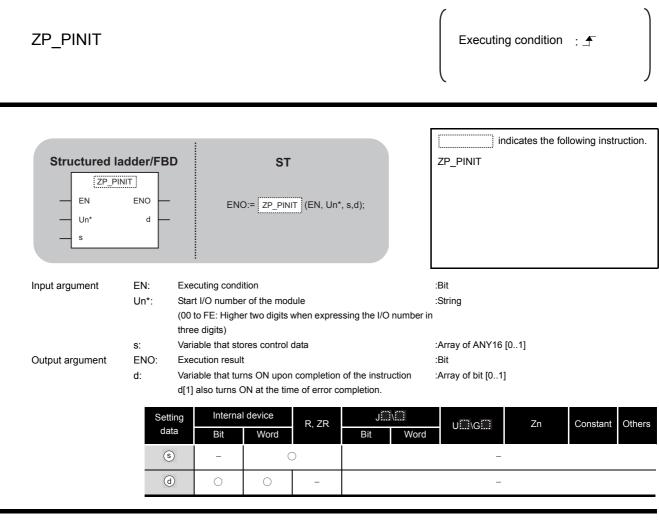
RST(TRUE, Var_Flag_Mem);

(* Turns write to flash ROM instruction memory OFF *)

END_IF; END_IF; 5

5.2.5 PINIT instruction

ZP_PINIT



Grant Function

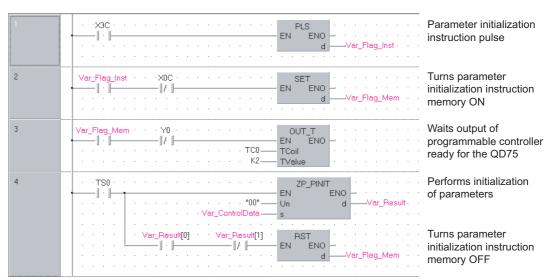
This instruction initializes the QD75 setting data.

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|--------------|-------------------|--|---------------|--------------|
| ⑤[0] | System area | - | - | - |
| | | The instruction completion status is stored. | | |
| ⑤ [1] | Completion status | O : Normal completion | - | System |
| | | Other than 0 : Error completion (error code) | | |

Program Example

The following program initializes the parameters of buffer memory and those of flash ROM when X3C turns ON.



[Structured ladder/FBD]

[ST]

PLS(X3C, Var_Flag_Inst); (* Parameter initialization instruction pulse *)

```
IF((Var_Flag_Inst=TRUE)&(X0C=FALSE))THEN
    SET(TRUE, Var_Flag_Mem);
                      (* Turns parameter initialization instruction memory ON *)
END_IF;
IF((Var_Flag_Mem=TRUE)&(Y0=FALSE))THEN
    OUT T(TRUE, TC0, 2);
                      (* Waits output of programmable controller ready for the QD75 *)
END_IF;
IF(TS0=TRUE)THEN
                      (* Parameter initialization instruction memory ON *)
    ZP_PINIT(TRUE, "00", Var_ControlData, Var_Result);
                      (* Performs initialization of parameters *)
    IF((Var_Result[0]=TRUE)&(Var_Result[1]=FALSE))THEN
          RST(TRUE, Var_Flag_Mem);
                      (* Turns parameter initialization instruction memory OFF *)
    END_IF;
END_IF;
```

5.3 Serial Communication

5.3.1 ONDEMAND instruction

G_ONDEMAND

| | | | | | | Serial |
|---|---------------------------|------------------------------------|---|------------|--|---------------------------------------|
| G(P)_ONDE | MAND | | | | P: Executing conc | lition : 🖵 |
| Structured I G_ONDE EN Un* s1 s2 | | | ST _ONDEMAND (EN, Un*, s1 | , s2, d); | indicates instructions. G_ONDEMAND | s any of the following GP_ONDEMAND |
| Input argument | EN: Un*: s1: s2: | three digits) Variable that ste | | | :Bit :ANY16 in :Array of ANY16 [02] :ANY16 | |
| Output argument | ENO: d: | | t ns ON upon completion o DN at the time of error con | | :Bit :Array of bit [01] | |
| | | Setting Int data *1 Bi | ernal device R, ZR | J\. Bit | Word U G | Zn Constant Others |
| | | 62 - (d) (C) | 0 | | - | |

*1: Local devices and file registers per program cannot be used as setting data.

Grant Function

This instruction sends data using the on-demand function of MC protocol.

| Device | Item | Setting data | Setting range | Setting side |
|--------|----------------------|--|---------------|--------------|
| | | Set the transmission channel. | | |
| s1 [0] | Transmission channel | 1: Channel 1 (CH1 side) | 1, 2 | User |
| | | 2: Channel 2 (CH2 side) | | |
| | | The instruction completion status is stored. | | |
| s1[1] | Transmission result | 0 : Normal completion | - | System |
| | | Other than 0 : Error completion (error code) | | |
| s1 [2] | Number of send data | Set the number of send data. | 1 or more | User |

Program Example

The following program sends data of devices from D10 to D11 using the on-demand function.

(For the Q series C24 whose I/O signals are X/Y00 to X/Y1F)

[Structured ladder/FBD]

| 1 | · ·×53 · · · · · · · · · · · · · · · · · · · | On-demand transmission instruction pulse |
|---|--|--|
| 2 | Var_Frag_Inst | Sets transmission channel to 1 |
| | MOV EN S d Var_ControlData[2] | Sets number of send data to 2 words |
| | H1234 EN ENO s d D10 | Sets send data to D10 to D11 |
| | H5678 BNO D11 | |
| | EN ENOVar_Flag_Normal | Turns normal completion flag OFF |
| | EN ENO dVar_Flag_Error | Turns error completion flag OFF |
| | EN ENO d Var_Flag_Exe | Turns execution flag ON |
| 3 | Var_Flag_Exe GP_ONDEMAND Image: Sector of the sector | Performs on-demand function transmission |
| 4 | Var_Result[0] Var_Result[1] SET I I I I Var_Result[0] I I I | Turns normal completion flag ON |
| | · · · · · · · · · · · · · · · · · · · | Turns error completion flag ON |
| | RST EN ENO d Var_Flag_Exe | Turns execution flag OFF |

| [ST] | |
|-----------------------------------|--|
| PLS(X53, Var_Flag_Inst); | (* On-demand transmission instruction pulse *) |
| IF(Var_Flag_Inst=TRUE)THEN | (* Instruction flag ON *) |
| MOV(TRUE, 1, Var_ControlData[0]); | (* Sets transmission channel to 1 *) |
| MOV(TRUE, 2, Var_ControlData[2]); | (* Sets number of send data to 2 words *) |
| MOV(TRUE, H1234, D10); | (* Sets send data to D10 to D11 *) |
| MOV(TRUE, H5678, D11); | |
| RST(TRUE, Var_Flag_Normal); | (* Turns normal completion flag OFF *) |
| RST(TRUE, Var_Flag_Error); | (* Turns error completion flag OFF *) |
| SET(TRUE, Var_Flag_Exe); | (* Turns execution flag ON *) |
| END_IF; | |
| IF(Var_Flag_Exe=TRUE)THEN | (* Execution flag ON *) |
| GP_ONDEMAND(TRUE, H0, Var_Co | ntrolData, D10, Var_Result); |
| | (* Performs on-demand function transmission *) |
| END_IF; | |
| IF(Var_Result[0]=TRUE)THEN | (* Execution finished *) |
| IF(Var_Result[1]=FALSE)THEN | (* Normal completion *) |
| SET(TRUE, Var_Flag_Normal) | ; (* Turns normal completion flag ON *) |
| ELSE | (* Error completion *) |
| SET(TRUE, Var_Flag_Error); | (* Turns error completion flag ON *) |
| END_IF; | |
| RST(TRUE, Var_Flag_Exe); | (* Turns execution flag OFF *) |
| END_IF; | |
| | |

- 1. The communication status can be checked by the SPBUSY instruction. \square Section 5.3.6
- 2. Specify the capacity of the send data (stored in devices from D10 to D11 in the program example above) and the number of send data within the user-defined buffer memory range assigned for the on-demand function.

DDULE DEDICATED STRUCTION **G**

5.3.2 OUTPUT instruction

G_OUTPUT

Serial

| G(P)_OUTP | UT | | | | | P: Executing cond | dition : 🛧 |
|---|-------------|-------------------------------------|----------------|---|----------------|---------------------------------------|------------------------|
| Structured la G_OUT EN Un* s1 s2 | | | ENO:= G_OU | ST TPUT (EN, Un*, s1, s2, | d); | indicate instructions. G_OUTPUT | s any of the following |
| Input argument | EN: Un*: | Start I/O (00 to FE three dig | its) | digits when expressing | the I/O number | | |
| | s1: | | that stores co | | . 1. | :Array of ANY16 [02] | |
| Output argument | s2: ENO: | Start nun Executio | | evice that stores send d | ata | :ANY16 :Bit | |
| Supul argument | d: | Variable | that turns ON | I upon completion of the time of error comple | | Array of bit [01] | |
| | | Setting data ^{*1} | Internal o | device R, ZR | J∭\∭ Bit \ | U[]]\G[]] Word | Zn Constant Others |
| | Ī | <u>s1</u> | - | 0 | | _ | |
| | 1 | s2 | - | 0 | | _ | |
| | | | | | | | |

Grant Function

This instruction sends data in the message format specified by the user using the nonprocedural protocol.

| Device | Item | Setting data | Setting range | Setting side |
|---------|----------------------|---|---------------|--------------|
| ্রা [0] | Transmission channel | Set the transmission channel. 1: Channel 1 (CH1 side) 2: Channel 2 (CH2 side) | 1, 2 | User |
| গে[1] | Transmission result | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| s1 [2] | Number of send data | Set the number of send data. | 1 or more | User |

Program Example

The following program sends data of devices from D11 to D15 using the nonprocedural protocol. (For the Q series C24 whose I/O signals are X/Y00 to X/Y1F) [Structured ladder/FBD]

| 1 | ·x20 · · · · · · · · · · · · · · · · · · · | Transmission instruction pulse |
|---|---|-------------------------------------|
| 2 | Var_Flag_Inst | Sets send data |
| | H0A0D s d D15 | Sets completion code |
| | MOV EN ENO | Sets transmission channel to 1 |
| | MOV EN ENO | Sets number of send data to 5 words |
| | G_OUTPUT EN ENO Un* d Var_ControlDatas1 s2 | Sends data |
| 3 | Var_Result[0] · Var_Result[1] · · · · · · · · · · · · · · · · · · · | Turns normal completion flag ON |
| | · Var_Result[1] MOV · Var_ControlData[1] EN · Var_ErrorCode | Stores error code |
| | EN ENO d Var_Flag_Error | Turns error completion flag ON |
| 4 | ·×21 · · · · · · · · · · · · · · · · · · · | Turns normal completion flag OFF |
| | RST EN ENO d Var_Flag_Error | Turns error completion flag OFF |

5-69

| [ST] | |
|---|---|
| PLS(X20, Var_Flag_Inst); | (* Transmission instruction pulse*) |
| IF (Var_Flag_Inst=TRUE) THEN MOV(TRUE, H4241, D11); MOV(TRUE, H4443, D12); MOV(TRUE, H4645, D13); MOV(TRUE, H0047, D14); MOV(TRUE, H0AD, D15); MOV(TRUE, 1, Var_ControlData[0]); MOV(TRUE, 5, Var_ControlData[2]); | (* Sets send data *) (* Sets transmission channel to 1 *) (* Sets number of send data to 5 words *) |
| G_OUTPUT(TRUE, H0, Var_ControlData[2]), | |
| | (* Sends data *) |
| END_IF; | |
| IF(Var_Result[0]=TRUE)THEN IF(Var_Result[1]=FALSE)THEN SET(TRUE, Var_Flag_Normal); ELSE MOV(TRUE, Var_ControlData[1], Var SET(TRUE, Var_Flag_Error); END_IF; END_IF; | |
| IF (X21=TRUE) THEN RST(TRUE, Var_Flag_Normal); RST(TRUE, Var_Flag_Error); END_IF; | (* Turns normal completion flag OFF *) (* Turns error completion flag OFF *) |

5.3.3 INPUT instruction

G_INPUT



G_INPUT

| Structured Ia G_INF EN Un* s | | | NO:= G_ | ST | , Un*, s, d1, | d2); | | indicat ructions. NPUT | es any of ti | he following |
|--|------|--------------------|-------------|---------------|----------------|-------------|---------|------------------------------|--------------|--------------|
| Input argument | EN: | Executing | g condition | | | | :Bit | | | |
| | Un*: | | | the module | | | :ANY | ′16 | | |
| | | | | o digits wher | n expressing | the I/O nun | nber in | | | |
| | | three dig | , | | | | | | | |
| | S: | | | control data | | | | y of ANY16 [03] | | |
| Output argument | ENO: | Execution | | | | | :Bit | | | |
| | d1: | | | | stores receiv | | :ANY | | | |
| | d2: | | | | npletion of th | | n :Arra | y of bit [01] | | |
| | - | | | | | | | | | |
| | | Setting | | al device | R, ZR | |]\] | U\G | Zn | Constant Ot |
| | | data ^{*1} | Bit | Word | | Bit | Word | | | |
| | | s | - | (| C | | | - | | |
| | | | | | | | | | | |
| | | d1 | - | (| C | | | - | | |

Grant Function

This instruction receives data in the message format specified by the user using the nonprocedural protocol.

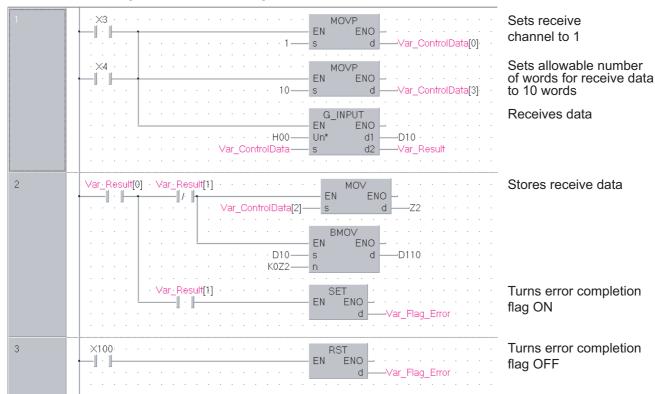
JLE DEDICATED RUCTION **G1**

| Device | Item | Setting data | Setting range | Setting side |
|--------------|--|---|---------------|--------------|
| s [0] | Reception channel | Set the reception channel. 1: Channel 1 (CH1 side) 2: Channel 2 (CH2 side) | 1, 2 | User |
| ঙ[1] | Reception result | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | - | System |
| <u> </u> (হ] | Number of receive data | The number of receive data are stored. | 0 or more | System |
| <u>জ</u> [3] | Allowable number of words for receive data | Set the allowable number of words for receive data to be stored in $\textcircled{0}$. | 1 or more | User |

Program Example

The following program stores data which are received using the nonprocedural protocol in the devices starting from D10.

(For the Q series C24 whose I/O signals are X/Y00 to X/Y1F)



[Structured ladder/FBD]

```
[ST]
IF((X3=TRUE) OR (X4=TRUE))THEN
    MOVP(TRUE, 1, Var_ControlData[0]);
                                              (* Sets receive channel to 1 *)
    MOVP(TRUE, 10, Var_ControlData[3]);
                      (* Sets allowable number of words for receive data to 10 words *)
    G_INPUT(TRUE, H0, Var_ControlData, D10, Var_Result);
                                                    (* Receives data *)
END_IF;
IF(Var_Result[0]=TRUE)THEN
                                                    (* Execution finished *)
    IF(Var Result[1]=FALSE)THEN
                                                    (* Normal completion *)
          MOV(TRUE, Var_ControlData[2], Z2);
          BMOV(TRUE, D10, K0Z2, D110);
                                                    (* Stores receive data *)
    ELSE
                                                    (* Error completion *)
          SET(TRUE, Var_Flag_Error);
                                                    (* Turns error completion flag ON *)
    END_IF;
END IF;
IF(X100=TRUE)THEN
                                              (* Turns error completion flag OFF *)
    RST(TRUE, Var_Flag_Error);
END_IF;
```

5.3.4 BIDOUT instruction

G_BIDOUT

Serial

| G(P)_BIDOU | JT | | | | P: Executing condition : |
|---|-------------|-------------------------------|-----------------|---|---|
| Structured Ia G_BID EN Un* s1 s2 | | | ENO:= G_B | ST IDOUT (EN ,Un* ,s1 ,s2 ,d); | indicates any of the following instructions. G_BIDOUT GP_BIDOUT |
| Input argument | EN: Un*: | Start I/O | • | ne module digits when expressing the I/O r | :Bit :ANY16 number in |
| | s1: s2: | | that stores of | ontrol data levice that stores send data | :Array of ANY16 [02] :ANY16 |
| Output argument | SZ: ENO: | | on result | | :Bit |
| | d: | | | N upon completion of the instruct the time of error completion. | ction :Array of bit [01] |
| | | Setting data ^{*1} | Internal Bit | device R, ZR Word Bit | UIII\GIII Zn Constant Oth |
| | 1 | s1 | - | 0 | _ |
| | 1 | s2 | _ | 0 | - |
| | | | | | |

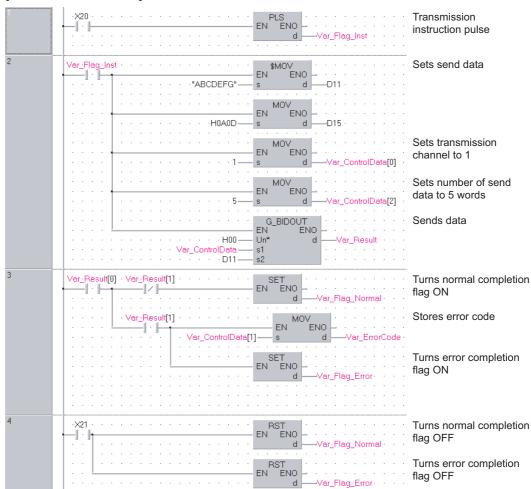
This instruction sends data using the bidirectional protocol.

| Device | Item | Setting data | Setting range | Setting side |
|--------|----------------------|---|---------------|--------------|
| s1 [0] | Transmission channel | Set the transmission channel. 1: Channel 1 (CH1 side) 2: Channel 2 (CH2 side) | 1, 2 | User |
| s1[1] | Transmission result | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| s1[2] | Number of send data | Set the number of send data. | 1 or more | User |

Program Example

The following program sends desired data stored in devices from D11 to D15 using the bidirectional protocol.

(For the Q series C24 whose I/O signals are X/Y00 to X/Y1F)



[Structured ladder/FBD]

G_BIDOUT

[ST] PLS(X20, Var_Flag_Inst); (* Transmission instruction pulse *) IF(Var_Flag_Inst=TRUE)THEN MOV(TRUE, H4241, D11); (* Sets send data *) MOV(TRUE, H4443, D12); MOV(TRUE, H4645, D13); MOV(TRUE, H0047, D14); MOV(TRUE, H0AD, D15); MOV(TRUE, 1, Var_ControlData[0]); (* Sets transmission channel to 1 *) MOV(TRUE, 5, Var ControlData[2]); (* Sets allowable number of words for send data to 5 words *) G_BIDOUT(TRUE, H0, Var_ControlData, D11, Var_Result); (* Sends data *) END_IF; IF(Var Result[0]=TRUE)THEN (* Execution finished *) IF(Var_Result[1]=FALSE)THEN (* Normal completion *) SET(TRUE, Var_Flag_Normal); (* Turns normal completion flag ON *) ELSE (* Error completion *) MOV(TRUE, Var_ControlData[1], Var_ErrorCode);(* Stores error code *) (* Turns error completion flag ON *) SET(TRUE, Var_Flag_Error); END IF; END_IF; IF(X21=TRUE)THEN RST(TRUE, Var_Flag_Normal); (* Turns normal completion flag OFF *) RST(TRUE, Var_Flag_Error); (* Turns error completion flag OFF *) END_IF;

G_BIDIN

Serial

5.3.5 BIDIN instruction

G(P)_BIDIN P: Executing condition : 1 indicates any of the following Structured ladder/FBD instructions. ST G_BIDIN GP_BIDIN G_BIDIN ΕN ENO ENO:= G_BIDIN (EN, Un*, s, d1, d2); Un* d1 d2 s :Bit Input argument EN: Executing condition Un*: :ANY16 Start I/O number of the module (00 to FE: Higher two digits when expressing the I/O number in three digits) Variable that stores control data :Array of ANY16 [0..3] s: Output argument ENO: Execution result :Bit d1: :ANY16 Start number of the device that stores receive data d2: :Array of bit [0..1] Variable that turns ON upon completion of the instruction d2[1] also turns ON at the time of error completion. Setting Internal device J...\ R, ZR Others U...\G Zn Constant data *1 Bit Word Word Bi (s) 0 d1) _ _ d2) _

*1: Local devices and file registers per program cannot be used as setting data.

Grant Function

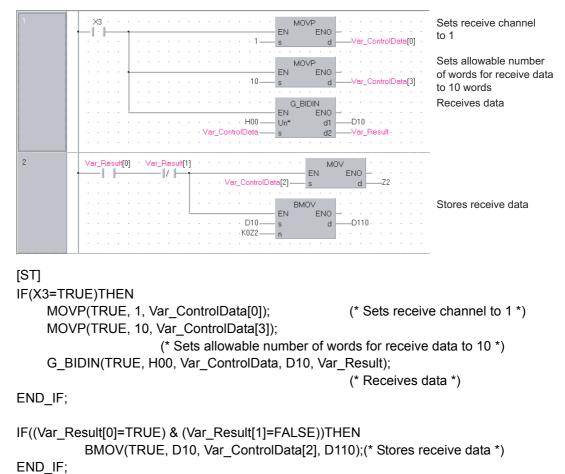
This instruction receives data using the bidirectional protocol.

| Device | Item | Setting data | Setting range | Setting side |
|--------------|--|---|---------------|--------------|
| (0] ھ | Reception channel | Set the reception channel. 1: Channel 1 (CH1 side) 2: Channel 2 (CH2 side) | 1, 2 | User |
| \$[1] | Reception result | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | - | System |
| ⑤ [2] | Number of receive data | The number of received data are stored. | 1 or more | System |
| s [3] | Allowable number of words for receive data | Set the allowable number of words for receive data to be stored in (d). | 1 or more | User |

Program Example

The following program receives data using the bidirectional protocol and stores the data in the devices starting from D10.

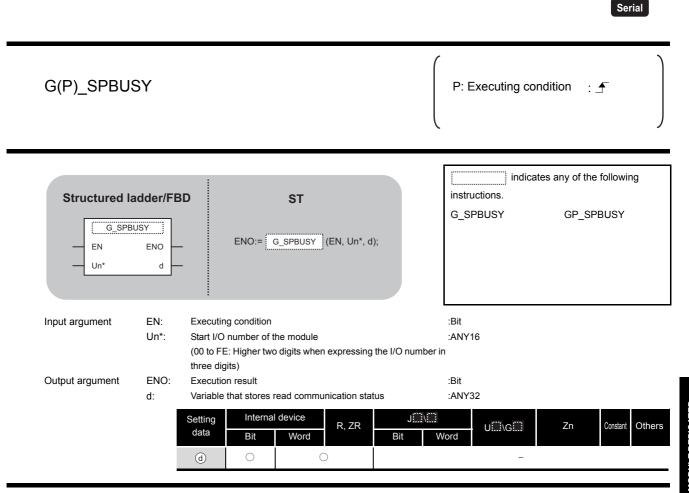
(For the Q series C24 whose I/O signals are X/Y00 to X/Y1F)



[Structured ladder/FBD]

G_SPBUSY

5.3.6 SPBUSY instruction



Grant Function

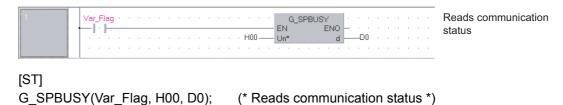
This instruction reads the data transmission/reception status.

Program Example

The following program reads out the communication status of the target module.

(For the Q series C24 whose I/O signals are X/Y00 to X/Y1F)

[Structured ladder/FBD]



G_SPBUSY

5.3.7 CSET instruction (receive data clear)

Serial ZP_CSET Executing condition : _ indicates the following instruction. Structured ladder/FBD ST ZP_CSET ZP_CSET ΕN ENO ENO:= ZP_CSET (EN, Un*, s1, s2, d1, d2); d1 Un' s1 d2 s2 :Bit Input argument EN: Executing condition Un*: :String Start I/O number of the module (00 to FE: Higher two digits when expressing the I/O number in three digits) s1: :ANY16 Channel number that requests receive data clear 1: Channel 1 (CH1 side) 2: Channel 2 (CH2 side) s2: Variable that stores control data :Array of ANY16 [0..111] Execution result :Bit Output argument ENO: d1: Dummy :ANY16 d2: :Array of bit [0..1] Variable that turns ON upon completion of the instruction d2[1] also turns ON at the time of error completion. Setting Internal device J....\... Constant R, ZR U...\G... Zn Others К, Н data Bit Word Bit Word (s1) _ 0 _ \bigcirc _ \bigcirc (s2) _ _ _ 0 d1) _ _ _ _ d2) _ _ _

*1: Local devices and file registers per program cannot be used as setting data.

ZP_CSET

Grant Function

Clears receive data without stopping transmission using the nonprocedural protocol.

| Device | Item | Setting data | Setting range | Setting side |
|--|-------------------|---|---------------|--------------|
| s2 [0] | Execution type | Specify '0'. | 0 | User |
| ⊚[1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | - | System |
| s2 [2] | Request type | Specify the request. 4: Receive data clear request | 4 | User |
| ☑ [3]to☑ [111] | For system | - | _ | System |

Program Example

The following program clears the receive data in the Q series C24 side.

(For the Q series C24 whose I/O signals are X/Y00 to X/Y1F)

| 1 | X20 X20 MOVP Clears execution I I EN ENO S d Var_ControlData[0] type to 0 |
|---|---|
| | EN ENO |
| | ····· ······· ······· Var_ControlData[2] ····· ······ ······ Clears data |
| | EN EN EN EN EN Var_Dummy Un* d1 Var_Dummy Var_Result Var_Result S1 d2 Var_Result Var_Ness |
| 2 | Var_Result[0] Var_Result[1] SET Turns normal completion Image: Set in the second |
| | Var_Result[1] SET Turns error completion Image: Non-State State Sta |
| | MOV MOV Stores error code |
| | · · · · · · · · · · · · · · · · · · · |
| 3 | • X21 • • • • • • • • • • • • • • • • • • • |
| | RST Turns error completion |
| | d <u>Var_Flag_Error</u> |

[Structured ladder/FBD]

ULE DEDICATED RUCTION **G**

ZP_CSET

```
[ST]
IF(X20=TRUE)THEN
    MOVP(TRUE, 0, Var_ControlData[0]);
                                              (* Clears execution type to 0 *)
    MOVP(TRUE, 4, Var_ControlData[2]);
                                               (* Sets request type *)
    ZP_CSET(TRUE, "00", 1, Var_ControlData, Var_Dummy, Var_Result);
                                              (* Clears data *)
END_IF;
IF(Var_Result[0]=TRUE)THEN
                                              (* Execution finished *)
    IF(Var_Result[1]=FALSE)THEN
                                               (* Normal completion *)
          SET(TRUE, Var_Flag_Normal);
                                               (* Turns normal completion flag ON *)
    ELSE
                                               (* Error completion *)
          MOV(TRUE, Var_ControlData[1], Var_ErrorCode);
                                              (* Stores error code *)
                                              (* Turns error completion flag ON *)
          SET(TRUE, Var_Flag_Error);
    END_IF;
END IF;
IF(X21=TRUE)THEN
    RST(TRUE, Var_Flag_Normal);
                                              (* Turns normal completion flag OFF *)
                                              (* Turns error completion flag OFF *)
    RST(TRUE, Var_Flag_Error);
END_IF;
```

5.3.8 BUFRCVS instruction

Z_BUFRCVS



Z_BUFRCVS

| Structured la EN Un* s | | | ST Z_BUFRCVS (EN, Un*, s, d); | Z_BUFRCVS | s the follo | owing instructio |
|---------------------------------|------|--|---|-----------------|-------------|------------------|
| Input argument | EN: | Executing condition | on | :Bit | | |
| | Un*: | Start I/O number of (00 to FE: Higher for the second secon | two digits when expressing the I | :String /O | | |
| | s: | Reception channe 1: Channel 1 (CH 2: Channel 2 (CH2 | 1 side) | :ANY16 | | |
| Output argument | ENO: | Execution result | , | :Bit | | |
| | d: | | e device that stores read data e read from the receive area of b | :ANY16 uffer | | |
| | | Setting Intern data ^{*1} Bit | nal device R, ZR Book R, ZR Book Book Book Book Book Book Book Boo | J\GU\G | Zn | Constant K, H |
| | Ī | <u>s</u> – | 0 | | | 0 - |
| | - | | | | | |

Z_BUFRCVS

5

LE DEDICATED

Grant Function

This instruction receives data with an interrupt program during communication using the nonprocedural protocol or bidirectional protocol.

Receive Data

| Item | Setting data | Setting range | Setting side | |
|---------------------|--|--|---|--|
| Receive data length | The number of data read from the number of receive | 0 or more | System | |
| | data storage area is stored. | | | |
| | | | | |
| Receive data | , and the second s | _ | System | |
| | stored in ascending address order. | | | |
| | Receive data length | Receive data length The number of data read from the number of receive data storage area is stored. Data read from the receive data storage area are | Receive data length The number of data read from the number of receive data storage area is stored. 0 or more Receive data Data read from the receive data storage area are | |

Program Example

The following program receives data with an interrupt program.

[Structured ladder/FBD]

| 1 | DX3 · · · · · · · · · · · · · · · · · · · | Turns normal completion flag ON |
|---|---|------------------------------------|
| 2 | DX4 · · · · · · · · · · · · · · · · · · · | Turns error completion flag ON |
| 3 | SM400 SM400 SM400 SM400 SM400 S S S SM400 S S S | Executes interrupt receive program |

[ST]

(* Set the normal/error confirmation flag for the main program *)
(* The main program resets flags *)
SET(DX3, Var_Flag_Normal);
SET(DX4, Var_Flag_Error);
(* Turns error completion flag ON *)

(* Receives data from CH1 and stores the data in devices starting from D200 *) Z_BUFRCVS(SM400, "00", 1, D200); (* Executes interrupt receive program *)

G_PRR

Sorial

5.3.9 PRR instruction

| | | | | | | Serial |
|---------------------------------------|------------|---|--|-----------------|--|----------------------------------|
| G(P)_PRR | | | | | P: Executing cond | dition : 🗲 |
| Structured I G_P EN Un* s | | | ST PRR (EN, Un*, s, d); | | G_PRR | s any of the following GP_PRR |
| Input argument | Un*: | Executing condition Start I/O number of the (00 to FE: Higher two di three digits) | module gits when expressing the | I/O number i | :Bit :ANY16 n | |
| Output argument | ENO: d: | | trol data upon completion of the in e time of error completion | | :Array of ANY16 [04] :Bit :Array of bit [01] | |
| | | Setting Internal de jata *1 Bit (s) - (d) O | Word R, ZR | J∷∷\ Bit V | Vord UIII\GIII | Zn Constant Other |
| | | | | evices and file | e registers per program c | annot be used as setting dat |



This instruction sends data by user frame according to the specification in user frame specification area for transmission during communication using the nonprocedural protocol.

G_PRR

DULE DEDICATED

| Device | Item | Setting data | Setting range | Setting side |
|--------------|------------------------------|--|---------------|--------------|
| | | Set the transmission channel. | | |
| ⑤[0] | Transmission channel | 1: Channel 1 (CH1 side) | 1, 2 | User |
| | | 2: Channel 2 (CH2 side) | | |
| | | The instruction completion status is stored. | | |
| ⑤ [1] | ⑤[1] Transmission result | 0 : Normal completion | - | System |
| | | Other than 0 : Error completion (error code) | | |
| | | Specify whether to add CR/LF codes to the send data. | | |
| s[2] | CR/LF addition specification | 0: CR/LF not added | 0, 1 | User |
| | | 1: CR/LF added | | |
| | | Specify the position in the user frame specification | | |
| ⑤ [3] | Transmission pointer | area for transmission from where the frame number | 1 to 100 | User |
| | | data are to be sent. | | |
| ⑤ [4] | Number of send data | Set the number of user frames to be sent. | 1 to 100 | User |

Program Example

The following program sends desired data and the user frames from number 1 to number 5 which are registered in the transmission frame setting.

(For the Q series C24 whose I/O signals are X/Y80 to X/Y9F)

Transmission ×50 PLS EN ENO instruction pulse 'ar Flag Inst d 2 Sets number of MO' ΕN ENO send data 4 TransData[0] Sets desired MOV ΕN ENO send data H1234 _TransData[1] MO ENO ΕN H56AB ar_TransData[2] Sets send data TO ENO ΕN to buffer memory Var_TransData[0] n1 n2 n3 H400 МО User frame 0 ENO ΕN H3F2 ar_Frame[0] User frame 1 MO' ΕN ENO H3F3 ar_Frame[1] User frame 2 MOV ΕN ENO H8001 ar_Frame[2] User frame 3 MO' ENO ΕN H8000 ar_Frame[3] User frame 4 MON ENO ΕN H41B ar_Frame[4] User frame 5 MO' ENO ΕN 0 ar_Frame[5] Sets user frames TO ENO ΕN to buffer memory Var_Frame[0] H8 n1 HOBA 6 n3 3 /ar_Flag_Inst MOV Sets transmission ΕN ENO channel to 1 1 ControlData[0] Clears transmission MOV ENO ΕN result 0 ControlData[1] MOV ENO Sets CR/LF to 'CR/LF ΕN not added' HO ControlData[2] Sets transmission MO' V ENO ΕN pointer H1 ControlData[3] Sets number of MOV ΕN ENO send data H5 ControlData[4] GP_PRR EN ENO Un* Performs user frame transmission H08 ar_Result Var. ControlDate SET EN ENO 4 /ar_Result[0] · Var_Result[1] Turns normal completion -171 -1 - 1 flag ON ar_Flaq_Normal Var_Result[1] SET EN ENO Turns error completion -|| · ||flag ON 'ar Flag Error

[Structured ladder/FBD]

[ST] (* Transmission instruction pulse *) PLS(X50, Var_Flag_Inst); IF((Var_Flag_Inst=TRUE) & (X9E=TRUE) & (X9F=FALSE))THEN MOV(TRUE, 4, Var TransData[0]); (* Sets number of send data *) MOV(TRUE, H1234, Var TransData[1]); (* Sets desired send data *) MOV(TRUE, H56AB, Var_TransData[2]); TO(TRUE, Var_TransData[0], H8, H400, 3); (* Sets send data to buffer memory *) MOV(TRUE, H3F2, Var_Frame[0]); (* Sets user frame 0 *) MOV(TRUE, H3F3, Var Frame[1]); (* Sets user frame 1 *) MOV(TRUE, H8001, Var Frame[2]); (* Sets user frame 2 *) MOV(TRUE, H8000, Var_Frame[3]); (* Sets user frame 3 *) MOV(TRUE, H41B, Var Frame[4]); (* Sets user frame 4 *) MOV(TRUE, 0, Var_Frame[5]); (* Sets user frame 5 *) TO(TRUE, Var_Frame[0], H8, H0BA, 6); (* Sets user frames to buffer memory *) END IF; IF(Var_Flag_Inst=TRUE)THEN MOV(TRUE, 1, Var_ControlData[0]); (* Sets transmission channel to 1 *) MOV(TRUE, 0, Var_ControlData[1]); (* Clears transmission result *) MOV(TRUE, H0, Var ControlData[2]); (* Sets CR/LF to 'CR/LF not added' *) MOV(TRUE, H1, Var ControlData[3]); (* Sets transmission pointer *) MOV(TRUE, H5, Var_ControlData[4]); (* Sets number of send data *) GP PRR(TRUE, H08, Var ControlData, Var Result); (* Performs user frame transmission *) END IF; IF(Var_Result[0]=TRUE)THEN (* Execution finished *) IF(Var_Result[1]=FALSE)THEN (* Normal completion *) SET(TRUE, Var_Flag_Normal); (* Turns normal completion flag ON *) ELSE (* Error completion *) SET(TRUE, Var_Flag_Error); (* Turns error completion flag ON *) END IF; END_IF;

ZP_CSET

5.3.10 CSET instruction (initial setting)

| | | | | | | Sei | rial |
|---------------------------------------|--|--|--|------------------|--|------------------|--------|
| ZP_CSET | | | | | Executing condition | : | |
| Structured I EN Un* s1 s2 | | ENO:=ZP_C | ST :SET (EN, Un*, s1, s2, d | 1, d2); | ZP_CSET | owing instr | uctior |
| Input argument | Un*: Start (00 t | - | the module o digits when expressing | the I/O number i | :Bit :String n | | |
| | s1: Rece 1: Cl | digits) ption channel i annel 1 (CH1 s annel 2 (CH2 s | side) | | :ANY16 | | |
| Output argument | s2: Varia ENO: Exec d1: Dum d2: Varia | ble that stores ution result ny ble that turns C | | | :Array of ANY16 [0111] :Bit :ANY16 :Array of bit [01] | | |
| | Settin data | | I device R, ZR Word | J∖ Bit V | U[]]\G[]] Zn | Constant K, H | Othe |
| | (s1) | - | 0 | | _ | 0 | _ |
| | 62 | - | 0 | | _ | - | _ |
| | đ | - | 0 | | - | - | |
| | d2 | 0 | 0 | | - | - | - |

*1: Local devices and file registers per program cannot be used as setting data.

Grant Function

This instruction changes the setting values for sending/receiving data using communication protocols.

5-89

ZP_CSET

5

E DEDICATED

| Device | Item | Setting data | Setting range | Setting side |
|-------------------------|--|--|---|--------------|
| s2 [0] | Execution type | Specify '0'. | 0 | User |
| s2[1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| €2 [2] | Request type | Specify the request. 1: Change of unit (word/byte) and buffer memory assignment | 1 | User |
| 2[3] | Word/byte unit specification | Specify the unit of the number of send/receive data. 0: Current setting value 1: In units of words 2: In units of bits | 0,1,2 | User |
| s2 [4] | Buffer memory start address for on- demand function | Specify the start address of the buffer memory used by the on-demand function 0н: Current setting value is used. 400н to 1AFFн, 2600н to 3FFFн: Start address | 0н, 400н to 1AFFн, 2600н to 3FFFн | User |
| ଛ2[5] | Buffer memory size for on-demand function | Specify the size (the number of words) of the buffer memory to be used by the on-demand function. 0н: Current setting value is used. 1н to 1A00н: Size | 0н, 1н to 1А00н | User |
| @[6] | Send area start address | Specify the start address of the send area used for the nonprocedural/bidirectional protocol. 0н: Current setting value is used. 400н to 1AFFн, 2600н to 3FFFн: Start address | 0н, 400н to 1АFFн, 2600н to 3FFFн | User |
| @[7] | Send area size | Specify the size (the number of words) of the send area used by the nonprocedural/bidirectional protocol. 0H: Current setting value is used. 1H to 1A00H: Size * The start area of the send area (1 word) is used for the number of send data specification area. | 0н, 1н to 1А00н | User |
| ⊚[8] | Receive area start address | Specify the start address of the receive area used for the nonprocedural/bidirectional protocol. 0н: Current setting value is used. 400н to 1AFFн, 2600н to 3FFFн: Start address | 0н, 400н to 1AFFн, 2600н to 3FFFн | User |
| @[9] | Receive area size | Specify the size (the number of words) of the receive area used for the nonprocedural/bidirectional protocol. Он: Current setting value is used. 1н to 1A00н: Size * The start area of the receive area (1 word) is used for the number of receive data storage area. | 0н, 1н to 1А00н | User |
| © [10] to © [111] | For system | _ | _ | System |

Program Example

The following program changes the send buffer area of the CH1 side interface.

(For the Q series C24 whose I/O signals are X/Y00 to X/Y1F)

- Sets send buffer to C00H to FFFH.
- Sets receive buffer to 600H to 7FFH.

[Structured ladder/FBD]

| 1 | Var_Flag_Inst FMOVP Image: Inst Image: Inst Image: Inst Image: Image: Inst Image: Image: Inst Image: Ima | Clears D0 to D111 to 0 |
|---|---|-----------------------------------|
| | Image: Non-state of the state of the st | Sets execution type |
| | MOVP EN ENO s dVar_ControlData[2] | Sets request type |
| | MOVP EN ENO s dVar_ControlData[3] · | Sets word/byte unit to word |
| | MOVP EN ENO S dVar_ControlData[4] | Sets on-demand start address |
| | | Sets on-demand buffer size |
| | MOVP EN ENO s d Var ControlData[6] | Sets send buffer start address |
| | MOVP EN ENO s d ControlData[7] | Sets send buffer size |
| | MOVP EN ENO s d Var ControlData[8] | Sets receive buffer start address |
| | | Sets receive buffer size |
| | H200 s d Var_ControlData[9] ZP_CSET | Performs initialization |
| 2 | Var_Result[0] Var_Result[1] s1 d2 Var_Result[1] | Turns normal |
| | EN ENO d Var_Flag_Normal | completion flag ON |
| | SET SET <td>Turns error completion flag ON</td> | Turns error completion flag ON |

ULE DEDICATED RUCTION **G**

ZP_CSET

| [ST] IF(Var_Flag_Inst=TRUE)THEN FMOVP(TRUE,0,112, Var_ControlData[0]); MOVP(TRUE, 0, Var_ControlData[0]); MOVP(TRUE, 1, Var_ControlData[2]); MOVP(TRUE, 1, Var_ControlData[3]); MOVP(TRUE, H400, Var_ControlData[4]); MOVP(TRUE, 0, Var_ControlData[5]); MOVP(TRUE, H400, Var_ControlData[6]); MOVP(TRUE, H400, Var_ControlData[6]); MOVP(TRUE, H400, Var_ControlData[7]); MOVP(TRUE, H400, Var_ControlData[8]); MOVP(TRUE, H200, Var_ControlData[9]); ZP_CSET(TRUE, "00", 1, Var_ControlData, | (* Sets execution type *) (* Sets request type *) (* Sets word/byte unit to word *) (* Sets on-demand start address *) (* Sets on-demand buffer size *) (* Sets send buffer start address *) (* Sets send buffer size *) (* Sets receive buffer start address *) (* Sets receive buffer size *) Var_Dummy, Var_Result); |
|---|--|
| END_IF; | (* Performs initialization *) |
| IF(Var_Result[0]=TRUE)THEN IF(Var_Result[1]=FALSE)THEN SET(TRUE, Var_Flag_Normal); ELSE SET(TRUE, Var_Flag_Error); END_IF; END_IF; | (* Execution finished *) (* Normal completion *) (* Turns normal completion flag ON *) (* Error completion *) (* Turns error completion flag ON *) |

5.3.11 CSET instruction (programmable controller CPU monitor)

ZP_CSET

| | | | | | | | Serial |
|-----------------|---------------------------|--|---------------------------------|-----------------------------------|---------------|--|-------------------|
| ZP_CSET | | | | | | Executing condition | : 🕈 |
| Structured Ia | | | NO:= ZP_CSE | ST (EN, Un*, s1, s2, d7 | 1, d2); | ZP_CSET | owing instruction |
| Input argument | EN: Un*: | Start I/O r (00 to FE | | module igits when expressing | the I/O numbe | :Bit :String r in | |
| | s1: | 1: Chann | n channel nun el 1 (CH1 side | e) | | :ANY16 | |
| Output argument | s2: ENO: d1: d2: | Variable t Executior Dummy Variable t | hat turns ON | | | :Array of ANY16 [0111] :Bit :ANY16 :Array of bit [01] | |
| | | Setting data ^{*1} | Internal de Bit | evice R, ZR Word | J∰\É. | 〕 Word U∭\G∭ Zn | Constant K, H |
| | Ī | §1 | - | 0 | | _ | 0 |
| | | 62 | - | 0 | | - | _ |
| | | dl | - | 0 | | - | - |

*1: Local devices and file registers per program cannot be used as setting data.

Grant Function

This instruction registers and cancels the programmable controller CPU monitoring.

| (1) | Registering the | programmable controller | CPU monitoring |
|-----|-----------------|-------------------------|----------------|
|-----|-----------------|-------------------------|----------------|

| Device | | Item | Setting data | Setting range | Setting side |
|---------------|---|--------------------------------------|---|---|--------------|
| s2 [0] | Execution type | | Specify '0'. | 0 | User |
| @[1] | Co | mpletion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| s2 [2] | Re | quest type | Specify the request. 2: Registration of programmable controller CPU monitoring | 2 | User |
| s2 [3] | Су | cle time unit | Specify the unit of cycle time. 0: 100ms 1: Second 2: Minute | 0 to 2 | User |
| s2 [4] | Су | cle time | Specify the cycle time. 1H to FFFFH: Cycle time | 1н to FFFFн | User |
| s2 [5] | | ogrammable controller CPU monitoring | Specify the monitoring function. 1: Constant cycle transmission 2: Condition agreement transmission | 1,2 | User |
| s2 [6] | Programmable controller CPU monitoring transmission method | | Specify the transmission method. 0: Data transmission (device data, CPU error information) 1: Notification | 0,1 | User |
| ⊚[7] | | User frame output start pointer | Specify the start pointer of the table to which the user frame number for constant cycle transmission is set. 0 : No specification (at condition agreement transmission and notification) 1 to 100 : Start pointer | 0, 1 to 100 | User |
| €2 [8] | Constant cycle transmission | Number of user frame transmissions | Specify the number of user frame transmissions (outputs) for constant cycle transmission. 0 : No specification (at condition agreement transmission and notification) 1 to 100 : Number of transmissions | 0, 1 to 100 | User |
| @[9] | | Modem connection data No. | Specify the data number for modem function connection when making notification in constant cycle transmission. 0 : No specification (at data transmission and condition agreement transmission) BB8H to BD5H : Connection data number (flash ROM) 8001H to 801FH: Connection data number (buffer memory) | 0, ВВ8н to ВD5н, 8001н to 801Fн | User |
| s2 [10] | Number of registered word blocks | | Specify the number of blocks of the word device to be monitored. | 0 to 10 | User |
| ፼ [11] | Number of registered bit blocks | | Specify the number of blocks of the bit device to be monitored. | 0 to 10 | User |
| ⊚[12] | Programmable controller CPU error monitoring (programmable controller CPU status monitoring) | | Specify whether to also execute programmable controller CPU error monitoring. 0: Not monitored 1: Monitored | 0,1 | User |

| Device | | | Item | Setting data | Setting range | Setting side |
|-------------------------|---|----------------------------------|---------------------------------------|--|---|--------------|
| s2 [13] | | Device code | | Specify the code of the device to be monitored. 0 : No device monitored Other than 0 : Device code | 90н to CCн (Device code) | User |
| s₂ [14] s₂ [15] | | Мо | nitoring start device | Specify the start number of the monitoring device in this block. | 0 or more | User |
| ଛି [16] | | Nu | mber of registered points | Specify the number of registered points (read points) of this block. 0 : No device monitored 1 or more : Number of registered points * For a bit device, specify the number of points in units of words. | 0, 1 or more | User |
| ፼[17] | | | Monitoring condition | Specify the monitoring condition of this block. 0 : No specification (at constant cycle transmission) 1 or more : Monitoring condition | 0 to 65535 | User |
| ଽୖ2[18] | Dragrammable | | Monitoring condition value | Specify the monitoring condition value for this block. 0 or more: Monitoring condition * Specify '0' at constant cycle transmission. | 0 to 000Ан, 0101н to 010Ан | User |
| ⊚[19] | Programmable controller CPU monitoring setting 1st * 1st block | 1 | User frame output start pointer | Specify the start pointer of the table to which the user frame number for condition agreement transmission for this block is set. 0 : No specification (at constant cycle transmission and notification) 1 to 100 : Start pointer | 0, 1 to 100 | User |
| @[20] | | Condition agreement transmission | Number of user frame transmissions | Specify the number of user frame transmissions (outputs) for condition agreement transmission for this block. 0 : No specification (at constant cycle transmission and notification) 1 to 100 : Number of transmissions | 0, 1 to 100 | User |
| ፼[21] | | | Modem connection data No. | Specify the data number for modem function connection when making notification in condition agreement transmission for this block. 0 : No specification (at data transmission and constant cycle transmission) BB8н to BD5н : Connection data number (flash ROM) 8001н to 801Fн: Connection data number (buffer memory) | 0, ВВ8н to ВD5н, 8001н to 801Fн | User |
| © [22] to © [102] | Programmable controller CPU monitoring setting 2nd to 10th * 2nd to 10th blo | ock | 1 | The same item arrangement as the first programmable controller CPU monitoring setting item. | _ | User |

| Device | | | Item | Setting data | Setting range | Setting side | | |
|----------|---|----------------------------------|------------------------------------|--|---|---|-------------|------|
| © [103] | | | | | 1 | | | |
| s2 [104] | | | | | 0 | | | |
| s2 [105] | | | | | 0 | 1 January | | |
| s2 [106] | | | Fixed value | Specify a fixed value to monitor the CPU status. | 1 | User | | |
| s2 [107] | + | | | | 5 | | | |
| © [108] | | | | | 1 | | | |
| ⊚[109] | CPU status monitoring setting | t transmission | User frame output start pointer | Specify the start pointer of the to which the user frame number for condition agreement transmission for this block is set. 0 : No specification (at constant cycle transmission and notification) 1 to 100 : Start pointer | 0, 1 to 100 | User | | |
| ⊚[110] | * Error monitoring 11th * 11th block | Condition agreement transmission | Condition agreemen | Condition agreemen | Number of user frame transmissions | Specify the number of user frame transmissions (outputs) for condition agreement transmission for this block. 0 : No specification (at constant cycle transmission and notification) 1 to 100 : Number of transmissions | 0, 1 to 100 | User |
| ⊚[111] | * | | Modem connection data No. | Specify the data number for modem function connection when making notification in condition agreement transmission for this block. 0 : No specification (at data transmission and constant cycle transmission) BB8H to BD5H : Connection data number (flash ROM) 8001H to 801FH: Connection data number (buffer memory) | 0, ВВ8н to ВD5н, 8001н to 801Fн | User | | |

(2) Canceling the programmable controller CPU monitoring

| Device | Item | Setting data | Setting range | Setting side |
|------------------------|-------------------|---|---------------|--------------|
| s2 [0] | Execution type | Specify '0н'. | 0 | User |
| @[1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| ⊚[2] | Request type | Specify the request. 3: Cancel of the programmable controller CPU monitoring | 3 | User |
| © [3] to © [111] | For system | - | - | System |

Program Example

Program to register the programmable controller CPU monitoring
 The following program registers the programmable controller CPU monitoring and sends the
 monitoring result from the CH1 side interface.
 The following setting is to send content of devices from M0 to M15 and devices from D100 to
 D109 to the external device through the constant cycle transmission. (Cycle time: 3 minutes)

·X24 Instruction pulse PLS ΕN ENO /ar_Flag_Inst d 2 Sets execution type Var_Flag_Inst Var_Flag_Normal MOVP ΕN ENO 0 d ControlData[0] Sets request type MOVE ΕN ENO 2 ControlData[2] d Sets cycle time unit MO' ΕN ENO to minute · 2 ar_ControlData[3] d Sets cycle time MOVP ΕN ENO to 3 minutes 3 d ControlData[4] Sets programmable controller MOVP CPU monitoring function to ΕN ENO constant cycle transmission · 1 d ar_ControlData[5] Sets transmission method MOVE ΕN ENO to data transmission 0 ControlData[6] s d Sets output start pointer ΕN ENO · 1 d ControlData[7] Sets number of user MO ΕN ENO frame transmissions 2 ControlData[8] d Turns data set flag ON SET ΕN ENO /ar_Flaq_Set d

(For the Q series C24 whose I/O signals are X/Y00 to X/Y1F)

[Structured ladder/FBD]

continued on the next page

| 3 | Var_Flag_Set · · · · · · · · · · · · · · · · MOVP | Poto number of registered |
|---|--|---|
| 5 | | Sets number of registered word blocks |
| | MOVP | Sets number of registered |
| | EN ENO | bit blocks |
| | MOVP | Sets device code |
| | EN ENO | Ceta device code |
| | | Sets start number of |
| | EN EN EN | monitoring device |
| | · · · · · · · · · · · · · · · · · · | |
| | | |
| | Var_ControlData[15] | |
| | | Sets number of registered points |
| | Var_ControlData[16] | |
| | MOVP EN ENO - · · · · · · · · · | Sets device code |
| | H90 s d Var_ControlData[22] | |
| | моvр • • • • • • • • • • • • • • • • • • • | Sets start number of monitoring device |
| | □ · · · · · · · · · · · · · · · · · · · | |
| | моvр ем ем ем ем ем на | |
| | Var_ControlData[24] | |
| | | Sets number of registered points (For bit device, specify it in units |
| | s d | of words) |
| | | Turns execution flag ON |
| | d | |
| | RST | Turns data set flag OFF |
| | d Var_Flag_Set | |
| 4 | | Pagiatara programmabla |
| 4 | Var_Flag_Exe ZP_CSET Image: Section of the section | Registers programmable controller CPU monitoring |
| | ···································· | |
| 5 | | Towns a second second stice |
| 5 | Var_Result[0] · Var_Result[1] · · · · · · · · · · · · · · · · · · · | Turns normal completion flag ON |
| | | Turne error completion |
| | | Turns error completion flag ON |
| | d —Var_Flag_Error | - |
| | | Turns execution flag OFF |
| | d | |
| | · • | |

[ST] PLS(X24, Var_Flag_Inst); (* Instruction pulse *) IF((Var_Flag_Inst=TRUE) & (Var_Flag_Normal=FALSE))THEN MOV(TRUE, 0, Var ControlData[0]); (* Sets execution type *) MOV(TRUE, 2, Var_ControlData[2]); (* Sets request type *) MOV(TRUE, 2, Var_ControlData[3]); (* Sets cycle time unit to minute *) MOV(TRUE, 3, Var ControlData[4]); (* Sets cycle time to 3 minutes *) MOV(TRUE, 1, Var_ControlData[5]); (* Sets programmable controller CPU monitoring function to constant cycle transmission. *) MOV(TRUE, 0, Var_ControlData[6]); (* Sets transmission method to data transmission *) MOV(TRUE, 1, Var ControlData[7]); (* Sets output start pointer *) MOV(TRUE, 2, Var_ControlData[8]); (* Sets number of user frame transmissions *) SET(TRUE, Var Flag Set); (* Turns data set flag ON *) END IF; IF(Var Flag_Set=TRUE)THEN MOV(TRUE, 1, Var_ControlData[10]);(* Sets number of registered word blocks *) MOV(TRUE, 1, Var_ControlData[11]);(* Sets number of registered bit blocks *) (* Sets the 1st block of the CPU monitoring to D100 to D109 *) MOV(TRUE, H0A8, Var ControlData[13]); (* Sets device code *) MOV(TRUE, 100, Var_ControlData[14]);(* Sets start number of monitoring device *) MOV(TRUE, 0, Var ControlData[15]); MOV(TRUE, 10, Var_ControlData[16]); (* Sets number of registered points *) (* Sets the 2nd block of the CPU monitoring to M0 to M15 *) (* Sets device code *) MOV(TRUE, H90, Var ControlData[22]); MOV(TRUE, 0, Var_ControlData[23]);(* Sets start number of monitoring device *) MOV(TRUE, 0, Var_ControlData[24]); MOV(TRUE, 1, Var ControlData[25]); (* Sets number of registered points. (For bit device, specify it in units of words.) *) SET(TRUE, Var Flag Exe); (* Turns execution flag ON *) RST(TRUE, Var Flag Set); (* Turns data set flag OFF *) END_IF; IF(Var Flag Exe=TRUE)THEN ZP CSET(TRUE, "00", 1, Var ControlData, Var Dummy, Var Result); (* Registers the programmable controller CPU monitoring *) END_IF; IF(Var_Result[0]=TRUE)THEN (* Execution finished *) IF(Var Result[1]=FALSE)THEN (* Normal completion *) (* Turns normal completion flag ON *) SET(TRUE, Var Flag Normal); ELSE (* Error completion *)

SET(TRUE, Var_Flag_Error); END_IF;

RST(TRUE, Var_Flag_Exe); END_IF; (* Turns execution flag OFF *)

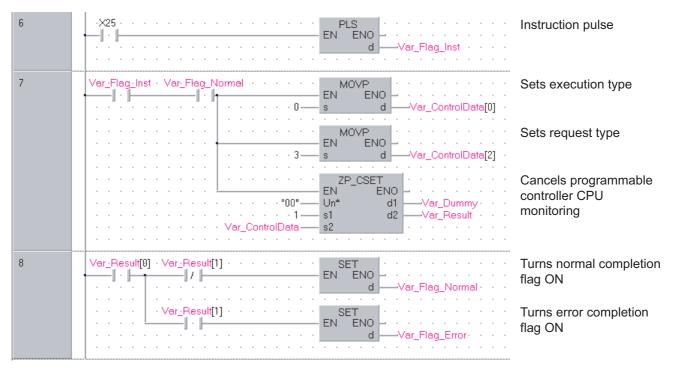
(* Turns error completion flag ON *)

INSTRUCTION

5

(2) Program to cancel the programmable controller CPU monitoring The following program cancels the programmable controller CPU monitoring of the CH1 side interface.

(For the Q series C24 whose I/O signals are X/Y00 to X/Y1F) [Structured ladder/FBD]



[ST]

END_IF;

PLS(X25, Var_Flag_Inst); (* Instruction pulse *) IF((Var_Flag_Inst=TRUE) & (Var_Flag_Normal=TRUE))THEN MOV(TRUE, 0, Var_ControlData[0]); (* Sets execution type *) MOV(TRUE, 3, Var_ControlData[2]); (* Sets request type *) ZP CSET(TRUE, "00", 1, Var ControlData, Var Dummy, Var Result); (* Cancels programmable controller CPU monitoring *) END_IF; IF(Var Result[0]=TRUE)THEN (* Execution finished *) IF(Var_Result[1]=FALSE)THEN (* Normal completion *) (* Turns normal completion flag ON *) SET(TRUE, Var_Flag_Normal); ELSE (* Error completion *) SET(TRUE, Var_Flag_Error); (* Turns error completion flag ON *) END IF;

G_PUTE

Serial

5.3.12 PUTE instruction

| G(P)_PUTE | | | | | | P: Executing co | ndition : _ | |
|--|-------------|-----------------|----------------|---|-------------------|----------------------|-------------------------|-----------------|
| Structured I GPU EN Un* s1 s2 | | 3D | ENO:= G | ST PUTE (EN, Un*, s1, s2 | 2, d); | G_PUTE | tes any of the GP_PU | - |
| Input argument | EN: Un*: | Start I/C | | he module o digits when expressing | the I/O number in | :Bit :ANY16 n | | |
| | s1: | | that stores | | | :Array of ANY16 [03] |] | |
| Output argument | s2: ENO: | | mber of the o | device that stores read r | egistration data | :ANY16 :Bit | | |
| | d: | | | N upon completion of th t the time of error comple | | :Array of bit [01] | | |
| | | Setting data *1 | Interna Bit | Word R, ZR | J\ | Vord U∭\G∭ | Zn | Constant Others |
| | | s1 | - | 0 | | - | | |
| | | (s2) | - | 0 | | _ | | |
| | | d | 0 | 0 | | _ | | |
| | | | | | | | | |

*1: Local devices and file registers per program cannot be used as setting data.

Grant Function

This instruction registers a user frame.

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|--------|-------------------------------------|--|---------------|--------------|
| s1 [0] | Registration/deletion specification | Specify whether to register/delete the user frame of the number specified by (5) [2]. 1: Registered 3: Deleted | 1, 3 | User |
| s1[1] | Registration/deletion result | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | - | System |
| s1 [2] | Frame No. | Specify the user frame number. | 1000 to 1199 | User |
| st [3] | Number of registered bytes | 1 to 80: Number of bytes of the user frame to be registered. * Specify any number in the range from 1 to 80 as a dummy when '3: Deleted' is selected. | 1 to 80 | User |

Program Example

The following program registers a user frame as the registration number 3E8H.

(For the Q series C24 whose I/O signals are X/Y80 to X/Y9F)

[Structured ladder/FBD]

| 1 | ·×50 · · · · · · · · · · · · · · · · · · · | Registration request pulse |
|---|---|--|
| 2 | Var_Flag_Inst MOV EN ENO S d Var_ControlData[0] | Sets registration request |
| | H3E8 B d Var_ControlData[2] | Sets user frame number |
| | EN ENO | Sets number of registered bytes |
| | H3946 — s d ──Var_Frame[0] | User frame 0 |
| | EN ENO s ds ds d | User frame 1 |
| | EN ENO | User frame 2 |
| | EN MOV EN ENO | User frame 3 |
| | EN MOV EN ENO s d | User frame 4 |
| | TO EN ENO | Sets write enable in flash ROM side |
| | G_PUTE EN ENO | Registers user frame |
| 3 | Var_Result[0] · Var_Result[1] · · · · · · · · · · · · · · · · · · · | Turns normal completion flag ON |
| | Var_Result[1] · · · · · EN ENO dVar_Flag_Error· | Turns error completion flag ON |

[ST] PLS(X50, Var_Flag_Inst); (* Registration request pulse *) IF(Var_Flag_Inst=TRUE)THEN MOV(TRUE, 1, Var ControlData[0]); (* Sets registration request *) MOV(TRUE, H3E8, Var ControlData[2]); (* Sets user frame number *) MOV(TRUE, 10, Var_ControlData[3]); (* Sets number of registered bytes *) MOV(TRUE, H3946, Var_Frame[0]); (* User frame 0 *) MOV(TRUE, H3030, Var_Frame[1]); (* User frame 1 *) MOV(TRUE, H3030, Var_Frame[2]); (* User frame 2 *) MOV(TRUE, H4646, Var Frame[3]); (* User frame 3 *) MOV(TRUE, H3030, Var_Frame[4]); (* User frame 4 *) TO(TRUE, 1, H08, H2000, 1); (* Sets write enable in flash ROM side *) G_PUTE(TRUE, H08, Var_ControlData, Var_Frame[0], Var_Result); (* Registers user frame *) END_IF; IF(Var_Result[0]=TRUE)THEN (* Execution finished *) IF(Var_Result[1]=FALSE)THEN (* Normal completion *) SET(TRUE, Var_Flag_Normal); (* Turns normal completion flag ON *) ELSE (* Error completion *) SET(TRUE, Var_Flag_Error); (* Turns error completion flag ON *) END IF;

```
END_IF;
```

DULE DEDICATED

5.3.13 GETE instruction

G(P)_GETE P: Executing condition : 🕈 indicates any of the following Structured ladder/FBD instructions. ST G_GETE GP_GETE G_GETE ΕN ENO ENO:= G_GETE (EN, Un*, s1, s2, d); Un' d s1 s2 :Bit Input argument EN: Executing condition :ANY16 Un*: Start I/O number of the module (00 to FE: Higher two digits when expressing the I/O number in three digits) s1: Variable that stores control data :Array of ANY16 [0..3] Start number the device that stores the read registration data :ANY16 s2: Output argument ENO: :Bit Execution result :Array of bit [0..1] d: Variable that turns ON upon completion of the instruction d[1] also turns ON at the time of error completion. Setting Internal device J....\.... Others R, ZR U...\G... Constant Zn data *1 Bit Word Bit Word (s1) 0 s2) _ d _ *1: Local devices and file registers per program cannot be used as setting data.

Grant Function

This instruction reads a user frame.



Control Data

| Device | Item | Setting data | Setting range | Setting side |
|--------|---|---|---------------|--------------|
| s1[0] | Dummy | - | 0 | - |
| s1[1] | Read result | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | - | System |
| s1[2] | Frame No. specification | Specify the user frame number. | 1000 to 1199 | User |
| st [3] | Allowable number of bytes for read data | Specify the maximum number of bytes for storing the registered data of the read user frame to @. | 1 to 80 | User |
| | Number of registered bytes | The number of bytes of the registered data for the read user frame is stored. | 1 to 80 | System |

Program Example

The following program reads out the registration data of the user frame number 3E8H.

(For the Q series C24 whose I/O signals are X/Y80 to X/Y9F)

[Structured ladder/FBD]

| 1 | ·X51 ····· EN EN ENO ····· ····· ····· ····· ····· ····· ····· ····· | Read request pulse |
|---|--|---|
| 2 | Var_Flag_Inst MOV Image: State S | |
| | H3E8 H3E8 d Var_ControlData[2] | Sets user frame number Sets allowable number |
| | Image: Move and the second | of bytes for read data |
| | ······ ······ ······ FMOV ····· ····· ····· EN ENO ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ······ ······ ······ ····· ······ ····· ······ ······ ····· ······ ······ ······ ······ ····· ······· ······ ······ ······ | Clears user frame to 0 |
| | G_GETE EN ENO Un* d Var_ControlData s1 S2 | Reads user frame |
| 3 | Var_Result[0] · Var_Result[1] · · · · · · · · · · · · · · · · · · · | Turns normal completion flag ON |
| | ····· ····· SET ····· ····· ····· EN ENO d ····· ····· | Turns error completion flag ON |

G_GETE

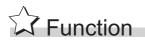
[ST] PLS(X51, Var_Flag_Inst); (* Read request pulse *) IF(Var_Flag_Inst=TRUE)THEN MOV(TRUE, 0, Var_ControlData[0]); MOV(TRUE, H3E8, Var_ControlData[2]); (* Sets user frame number *) MOV(TRUE, 80, Var_ControlData[3]); (* Sets allowable number of bytes for read data *) FMOV(TRUE, 0, 40, Var_Frame[0]); (* Clears user frame to 0 *) G_GETE(TRUE, H08, Var_ControlData, Var_Frame[0], Var_Result); (* Reads user frame *) END_IF; IF(Var_Result[0]=TRUE)THEN (* Execution finished *) IF(Var_Result[1]=FALSE)THEN (* Normal completion *) SET(TRUE, Var_Flag_Normal); (* Turns normal completion flag ON *) ELSE (* Error completion *) SET(TRUE, Var_Flag_Error); (* Turns error completion flag ON *) END_IF; END_IF;

5-106

ZP_UINI

5.3.14 UINI instruction

Serial **ZP_UINI** Executing condition : indicates the following instruction. Structured ladder/FBD ST ZP_UINI ZP_UINI ΕN ENO ENO:= ZP_UINI (EN, Un*, s, d); Un' d s :Bit Input argument Executing condition EN: Un*: :String Start I/O number of the module (00 to FE: Higher two digits when expressing the I/O number in three digits) Variable that stores control data :Array of ANY16 [0..9] s: Output argument Execution result :Bit ENO: Variable that turns ON upon completion of the instruction d: :Array of bit [0..1] d[1] also turns ON at the time of error completion. Setting Internal device J....\.... R, ZR Zn Constant Others U....\G.... data^{*1} Bit Word Word Bit (s)(d) *1: Local devices and file registers per program cannot be used as setting data.

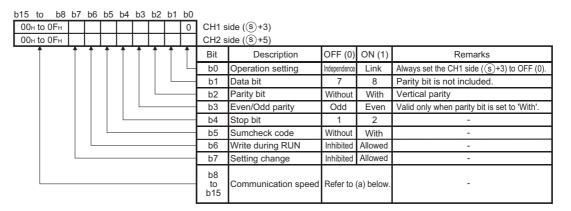


This instruction switches the mode, transmission specification, and host station number of the Q series C24.

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|-----------------------|--|--|---------------|--------------|
| ⑤ [0] | For system | Always specify '0'. | 0 | User |
| \$[1] | Execution result | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| ঙ[2] | Execution type | Specify the execution type. 0: Switches the execution type according to the setting in the area starting from (\$[3]]. 1: Returns the execution type according to the switch setting on GX Works2. | 0, 1 | |
| s [3] | CH1 Transmission specification setting | Set the transmission specifications for CH1. (Refer to (1).) | 0 to 0FFEн | |
| ঙ [4] | CH1 Communication protocol setting | Set the communication protocol for CH1. (Refer to (2).) | 0 to 8 | User |
| \$[5] | CH2 Transmission specification setting | Set the transmission specifications for CH2. (Refer to (1).) | 0 to 0FFFн | |
| \$[6] | CH2 Communication protocol setting | Set the communication protocol for CH2. (Refer to (2).) | 0 to 7 | |
| ⑤ [7] | Station No. setting | Set the host station number. | 0 to 31 | |
| \$[8] to \$[12] | For system | Always specify '0'. | 0 | |

(1) $\,\,\,^{(1)}$ $\,\,^{(2)}$ [3] (CH1 Transmission specification setting) and $\,^{(3)}$ [5] (CH2 Transmission specification setting) *1



(a) Communication speed

| Communication | Bit position | Communication | Bit position | Remarks |
|---------------|--------------|---------------|--------------|---|
| speed | b15 to b8 | speed | b15 to b8 | Kennarks |
| 50bps | 0Fн | 14400bps | 06н | • 230400bps is selectable only at |
| 300bps | 00н | 19200bps | 07н | , , , |
| 600bps | 01н | 28800bps | 08н | CH1 side ($($ (\mathbb{S})[3]). (Select 300bps at |
| 1200bps | 02н | 38400bps | 09н | CH2 side (⑤ [5]).) |
| 2400bps | 03н | 57600bps | 0Ан | • The sum of communication speeds |
| 4800bps | 04н | 115200bps | 0Вн | selected at CH1 side and CH2 side must be within 230400bps. |
| 9600bps | 05н | 230400bps | 0Сн | |

*1: Specify '0000H' at the CH side for which "MELSOFT connection" is specified in the communication protocol setting.

(2) $\$ (CH1 Communication protocol setting) and $\$ (6] (CH2 Communication protocol setting)

| Setting No. | Description | | Remarks |
|-------------|------------------------|----------|---|
| 0н | MELSOFT connection | | Specify '0000H' for the transmission specification setting. |
| 1н | | Format 1 | - |
| 2н | | Format 2 | - |
| 3н | MC protocol | Format 3 | - |
| 4н | | Format 4 | - |
| 5н | | Format 5 | - |
| 6н | Nonprocedural protocol | | - |
| 7н | Bidirectional protocol | | - |
| 8н | For link setting | | Setting is possible only for CH1 side (⑤[4]) |
| 9н | Pre-defined protocol | | Pre-defined protocol communication |

Precautions

The UINI instruction is applicable to the QJ71C24N (-R2/R4) of which the function version is B and the first five digits of the serial number are '06062' or higher.

Program Example

The following program changes settings of the Q series C24 mounted on the I/O numbers X/Y00 to X/Y1F as follows when X20 turns ON.

| | | Bit | | Setting | | |
|--------|--------------|-----------------|---------------|----------------------------|-------------------------|--------|
| Device | Position | Specified value | | Description | | value |
| | b0 | OFF | | Operation setting | Independence | |
| | b1 | ON | - | Data bit | 8 | t |
| | b2 | ON | | Parity bit | With | Ť |
| | b3 | OFF | CH1 | Even/Odd parity | Odd | Ť |
| | b4 | OFF | Transmission | Stop bit | 1 | 07E6H |
| s [3] | b5 | ON | specification | Sumcheck code | With | 072011 |
| | b6 | ON | setting | Write during RUN | Allowed | Ť |
| | b7 | ON | | Setting change | Allowed | • |
| | b8 to b15 | - | - | Communication speed | 19200bps | |
| s [4] | | - | | unication protocol setting | Link setting | 0008H |
| | b0 | ON | | Operation setting | Link | - |
| | b1 | ON | - | Data bit | 8 | |
| | b2 | ON | - | Parity bit | With | |
| | b3 | OFF | CH2 | Even/Odd parity | Odd | |
| | b4 | OFF | Transmission | Stop bit | 1 | 07E7H |
| s [5] | b5 | ON | specification | Sumcheck code | With | 0/2/11 |
| | b6 | ON | setting | Write during RUN | Allowed | † |
| | b7 | ON | - | Setting change | Allowed | † |
| | b8 to b15 | - | | Communication speed | 19200bps | |
| S [6] | | - | | unication protocol setting | MC protocol Format 5 | 0005H |
| s [7] | | - | Station | No. setting | 1 | 0001H |

| 1 | ·×20 ···Y2 ···Y3 ···×6 ···×00 ····· FMOV I↑ I ····I / I····I / I····I / I····I ··· EN H0 ····· H0 ····· s d -Var_ControlData[0] n | UINI instruction command |
|---|---|--|
| | H0 s dVar_ControlData[0 | Always sets 0 |
| | H0 s dVar_ControlData[1 | Clears control data to 0 |
| | H0 s d | |
| | H7E6 s d | Sets CH1 transmission |
| | H8 s dVar_ControlData[4 | Sets CH1 communication protocol |
| | H7E7 s dVar_ControlData[5 | Sets CH2 transmission specification |
| | H5 s d -Var_ControlData[6 | • |
| | H1 — s d —Var_ControlData[7 | |
| | ZP_UINI EN ENO Var_ControlData s | Switches mode |
| | EN ENO d Var_Flag | Turns interlock signal for*1 communication stop ON |
| 2 | Var_Result[0] · Var_Result[1] · Process on normal comple | tion Normal completion |
| | Process on error completio | · · · · · |
| | BRST CN EN EN C Var_Flag | Turns interlock signal for communication stop OFF*1 |
| 3 | Var Flag *1. | ss |

[Structured ladder/FBD]

*1: Create a program so that the data communication process does not run while the interlock signal for communication stop is ON.

```
[ST]
IF(LDP(TRUE,X20)
                                                     (* UINI instruction command *)
&(Y2=FALSE)
                                               (* CH1 mode switching request *)
&(Y9=FALSE)
                                               (* CH2 mode switching request *)
&(X6=FALSE)
                                               (* CH1 mode switching *)
&(X0D=FALSE))THEN
                                               (* CH2 mode switching *)
    (* Runs if there is no mode switching *)
    FMOV(TRUE, H0, 13, Var ControlData[0]); (* Clears control data to 0 *)
    MOV(TRUE, H0, Var_ControlData[0]);
                                               (* Always sets 0 *)
    MOV(TRUE, H0, Var_ControlData[1]);
                                               (* Clears execution result to 0 *)
    MOV(TRUE, H0, Var ControlData[2]);
                                               (* Sets execution type *)
    MOV(TRUE,H7E6,Var_ControlData[3]); (* Sets CH1 transmission specification *)
    MOV(TRUE,H8,Var_ControlData[4]); (* Sets CH1 communication protocol *)
    MOV(TRUE, H7E7, Var_ControlData[5]); (* Sets CH2 transmission specification *)
    MOV(TRUE, H5, Var_ControlData[6]);(* Sets CH2 communication protocol *)
    MOV(TRUE, H1, Var_ControlData[7]);
                                              (* Sets host station number *)
    ZP UINI(TRUE, "00", Var ControlData, Var Result); (* Switches mode *)
    SET(TRUE, Var_Flag); (* Turns interlock signal for communication stop ON *)*1
END_IF;
```

| IF(Var_Result[0]=TRUE)THEN | (* Execution finished *) |
|---|--|
| IF(Var_Result[1]=FALSE)THEN | (* Normal completion *) |
| (* Process on normal completion *) | |
| ELSE | (* Error completion *) |
| (* Process on error completion *) | |
| END_IF; | |
| RST(TRUE, Var_Flag); (* Turns interlock sig | gnal for communication stop OFF *) ^{*1} |

END_IF;

(* Do not perform the data communication process during interlock signal for communication stop ON *)

IF(Var_Flag=FALSE)^{*1} THEN

| ····· | , |
|----------------------------------|----|
| (* Data communication process *) | |
| ·i_` | نا |

END_IF;

*1: Create a program so that the data communication process does not run while the interlock signal for communication stop is ON.

5.3.15 CPRTCL instruction

Serial G(P)_CPRTCL P: Executing condition : 🕈 indicates any of the following instructions. Structured ladder/FBD ST G_CPRTCL GP_CPRTCL G_CPRTCL ENO ΕN _ Un' d ENO:= G_CPRTCL (EN, Un*, n1, n2, s, d); n1 n2 s Input argument EN: Executing condition :ANY16 Un*: Start I/O number of the module :ANY16 (00 to FE: Higher two digits when expressing the I/O number in three digits) ·ANY16 n1: Channel to communicate with other devices 1: Channel 1 (CH1 side) 2: Channel 2 (CH2 side) n2: Number of consecutive protocol executions (1 to 8) :ANY16 Start number of the device in which control data are stored :Array of ANY16 [0..17] s: Output argument ENO: :Bit Execution result d: Variable that turns ON upon completion of the instruction :Array of bit [0..1] d[1] also turns ON at the time of error completion.

| Setting | Internal | device | R, ZR | JIII\III | | U∭\G∭ | Zn | Constant | Others |
|--------------------|----------|--------|--------------|----------|------|-------|----|----------|--------|
| data ^{*1} | Bit | Word | ., <u></u> , | Bit | Word | 0(G | | К, Н | Guidie |
| n1 | _ | C |) | | | - | | 0 | - |
| n2 | - | C | \sim | | | - | | 0 | - |
| \$ | - | C |) | | | - | | - | _ |
| d | 0 | C |) | | | - | | Ι | - |

*1: Local devices and file registers per program cannot be used as setting data.

G_CPRTCL

This instruction executes the protocols and functional protocols written to the flash ROM by predefined protocol support function.

Control Data

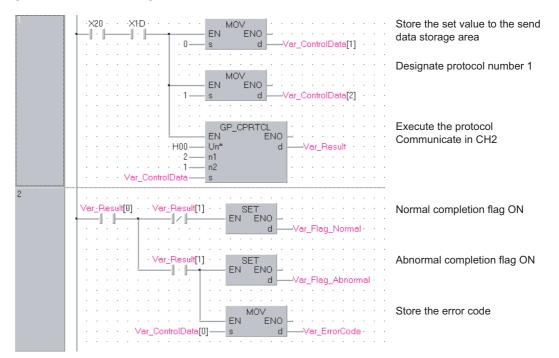
| Device | Item | Setting data | Setting range | Setting side |
|--------------|-----------------------|--|---------------|--------------|
| | | The instruction completion status is stored. | | |
| | | When executing multiple protocols, the execution | | |
| ⑤ [0] | Completion status | result of the protocol executed at last is stored.*1 | - | System |
| | | 0 : Normal completion | | |
| | | Other than 0 : Error completion (error code) ^{*2} | | |
| | | The number of executions is stored. | | |
| © [1] | Number of executions | Protocols with errors are included in the count. | 1 to 8 | System |
| ⑤ [1] | | When settings of the setting data and control data | 1100 | System |
| | | contain an error, "0" is stored. | | |
| ⑤ [2] | | Set the first protocol number or functional protocol | | |
| 0[2] | Execution protocol | number to be executed. | 1 to128, | |
| to | number designation | : | 201 to 207 | User |
| ⑤ [9] | | 201 10 201 | | |
| @[a] | | number to be executed. | | |
| | | When the communication type of the first protocol | | |
| | | executed is "Receive only" or "Send & receive", the | | |
| | | matched receive packet number is stored. | | |
| s[10] | | "0" is stored with the following condition. | | |
| | | When the communication type is "Send only" | | |
| | | • If the error occurs to the first protocol executed | | |
| | | When the functional protocol is executed | | |
| to | Verification match | : | | |
| | receive packet number | When the communication type of the 8th protocol | 0, 1 to 16 | System |
| | | executed is "Receive only" or "Send & receive", the | | |
| | | matched receive packet number is stored. | | |
| | | "0" is stored with the following condition. | | |
| s[17] | | When the communication type is "Send only" | | |
| | | If the error occurs to the 8th protocol executed | | |
| | | • When the number of the executed protocols is | | |
| | | less than 8 | | |
| | | When the functional protocol is executed | | |

*1: When executing multiple protocols, if an error occurs to the nth protocol, the protocols after the nth protocol are not executed.

*2 : For details of the error code at the error completion, refer to Q Corresponding Serial Communication Module User's Manual (Basic) or MELSEC-L Serial Communication Module User's Manual (Basic).

Program Example

This instruction executes the protocol specified in Var_ControlData[2] when X20 turns ON. [Structured ladder/FBD]



[ST]

IF((X20=TRUE) & (X1D=TRUE))THEN

MOV(TRUE, 0, Var_ControlData[1];

(* Store the set value to the send data storage area *) MOV(TRUE, 1, Var_ControlData[2];(* Designate protocol number 1 *) GP_CPRTCL(TRUE, H00, 2, 1, Var_ControlData, Var_Result);

(* Execute the protocol Communicate in CH2 *)

END_IF;

IF(Var_Result[0]=TRUE)THEN

IF(Var_Result[1]=FALSE)THEN

SET(TRUE, Var_Flag_Normal);(* Normal completion flag ON *)

ELSE

SET(TRUE, Var_Flag_Abnormal);(* Abnormal completion flag ON *) MOV(TRUE, Var_ControlData[0], Var_ErrorCode);

(* Store the error code *)

END_IF; END_IF;

 \sim

.

חחוח

חחוח

5.4 Network Dedicated Instruction

5.4.1 RIRD instruction

| | | | | | | J_ | _RIRD, G_RIRI |
|--|--------------------------|--------------------------------------|--|--|----------------------------------|--|--|
| | | | | | | l | CC-Link CC IE C CC IE F |
| J(P)_RIRD G(P)_RIRD | | | | | | P: Executing c | ondition : 🖣 |
| Structu I RRD EN ENO Jn* d1 s d2 | red ladder. | G_RIRD | - E | hannand | *, s, d1, d2); *, s, d1, d2); | instructions. J_RIRD G_RIRD | cates any of the following JP_RIRD GP_RIRD |
| Input argument | EN: Jn*: | Network 254: Netw | | the host station (1 to a lied in "Valid module c | | :Bit :ANY16 n | |
| | Un*: | (00 to FE | • | the module o digits when express | ing the I/O numbe | :ANY16 er in | |
| Output argument | s: ENO: d1: d2: | Execution Start num Variable t | that stores in result inber of the that turns C | control data device that stores rea N upon completion o at the time of error co | of the instruction | :Array of ANY16 [0 :Bit :ANY16 :Array of bit [01] | 4] |
| | | Setting | | l device R, ZR | J | U:;\G:; | Zn Constant Other |
| | | data *1 | Bit | Word | Bit | Word | |
| | - | data *1 © (d1) | Bit _ _ | | Dit | | |

C Function

This instruction reads data for the specified number of points from the buffer memory of the CC-Link module or the device of the programmable controller CPU module on the specified station.

E Control Data

| Device | Item | Setting data | Setting range | Setting side |
|--------------|-------------------------------------|--|---|--------------|
| s [0] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) For error codes when target station is anything other than master/local module, refer to the manual of the target station. | - | System |
| ি [1] | Target station No. | Specify the station number of the target station. | 0 to 64 ^{*1} 0 to 120 | |
| s [2] | Access code, Attribute code | Specify the access code and attribute code of the device to be read. b15 b8b7 b0 Access code Attribute code | Refer to (1) and (2). | User |
| s [3] | Buffer memory address or device No. | Specify the start address of the buffer memory or the start number of the device. | Within the device range* ² | |
| <u></u> [4] | Number of read points | Specify the number of data to be read (in units of words). | 1 to 32 ^{*3} 1 to 480 ^{*4} | |

*1 : For G(P)_RIRD, the setting range shall be 0 to 64.

- *2: For details, refer to the manual for the local station or the intelligent device station from which data are read.
- When the random access buffer is specified, specify the start address of the random access buffer as 0. *3 : The value indicates the maximum number of data to be read.
 - Specify the value within the buffer memory capacity of the local station or the intelligent device station, or the receive buffer area setting range set by a parameter.
- *4 : When reading device data from the programmable controller CPU other than the QCPU (Q mode), QCPU (A mode) or QnACPU/AnUCPU, the setting range shall be 1 to 32 words.

(1) Buffer memory of the CC-Link module

| Buffer memory | | Access code | Attribute code |
|-------------------------------------|---|-------------|----------------|
| Buffer in an inte | Buffer in an intelligent device station | | |
| | Random access buffer | 20н | |
| | Remote input | 21н | |
| Buffer in a master or local station | Remote output | 22н | 04н |
| | Remote register | 24н | |
| | Link special relay | 63н | |
| | Link special register | 64н | |

| Device *1 | Name | Device type | | Unit | *2 | A.I. I. I. *2 |
|---------------------------------|------|-------------|------|-------------|---------------------------|------------------|
| Device ^{*1} | Name | Bit | Word | Onit | Access code ^{*2} | Attribute code*2 |
| Input relay | х | 0 | - | Hexadecimal | 01н | |
| Output relay | Y | 0 | - | Hexadecimal | 02н | |
| Internal relay | М | 0 | - | Decimal | 03н | |
| Latch relay | L | 0 | - | Decimal | 83н | |
| Link relay | В | 0 | - | Hexadecimal | 23н | |
| Timer (contact) | Т | 0 | - | Decimal | 09н | |
| Timer (coil) | Т | 0 | - | Decimal | 0Ан | |
| Timer (current value) | Т | - | 0 | Decimal | 0Сн | |
| Retentive timer (contact) | ST | 0 | - | Decimal | 89н | |
| Retentive timer (coil) | ST | 0 | - | Decimal | 8Ан | |
| Retentive timer (current value) | ST | - | 0 | Decimal | 8Сн | 05н |
| Counter (contact) | С | 0 | - | Decimal | 11н | |
| Counter (coil) | С | 0 | - | Decimal | 12н | |
| Counter (current value) | С | - | 0 | Decimal | 14н | |
| Data register ^{*3} | D | - | 0 | Decimal | 04н | |
| Link register ^{*3} | W | - | 0 | Hexadecimal | 24н | |
| File register | R | - | 0 | Decimal | 84н | |
| Link special relay | SB | 0 | - | Hexadecimal | 63н | |
| Link special register | SW | - | 0 | Hexadecimal | 64н | |
| Special relay | SM | 0 | - | Decimal | 43н | 1 |
| Special register | SD | - | 0 | Decimal | 44н | |

(2) Device memory of the programmable controller CPU module

*1 : Devices other than those listed above cannot be accessed.

When accessing a bit device, specify it with 0 or a multiple of 16.

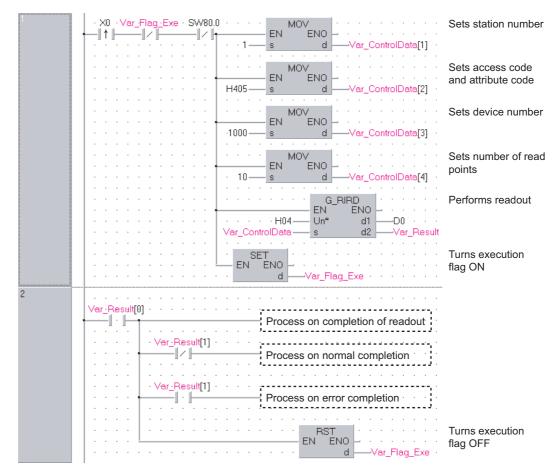
*2: For access code/attribute code when target station is anything other than master/local module, refer to the manual of the target station.

*3 : D65536 and the following devices of extended data registers as well as W10000 and the following devices of extended link registers cannot be specified.

Program Example

The following program reads out 10-word data, which start from D1000 of the number 1 local station connected to the master module mounted on the I/O numbers from X/Y40 to X/Y5F, and stores the data in the devices starting from D0 when X0 turns ON.

(When the refresh device of the link special register (SW) is set to SW0.)



[Structured ladder/FBD]

| [ST] IF(X0=TRUE) &(Var_Flag_Exe=FALSE) &(SW80.0=FALSE))THEN MOV(TRUE,1, Var_ControlData[1]); MOV(TRUE,H0405, Var_ControlData[2]); MOV(TRUE, 1000, Var_ControlData[3]); MOV(TRUE, 10, Var_ControlData[3]); G_RIRD(TRUE, H04, Var_ControlData, D0, SET(TRUE, Var_Flag_Exe); END_IF; | (* Sets device number *) (* Sets number of read points *) Var_Result);(* Performs readout *) |
|---|--|
| IF(Var_Result[0]=TRUE)THEN | (* Execution finished *) |
| (* Process on completion of readout *) | |
| IF(Var_Result[1]=FALSE)THEN | |
| (* Process on normal completion *) | |
| ELSE | (* Error completion *) |
| (* Process on error completion *) | |
| END_IF; | |
| RST(TRUE, Var_Flag_Exe); | (* Turns execution flag OFF *) |

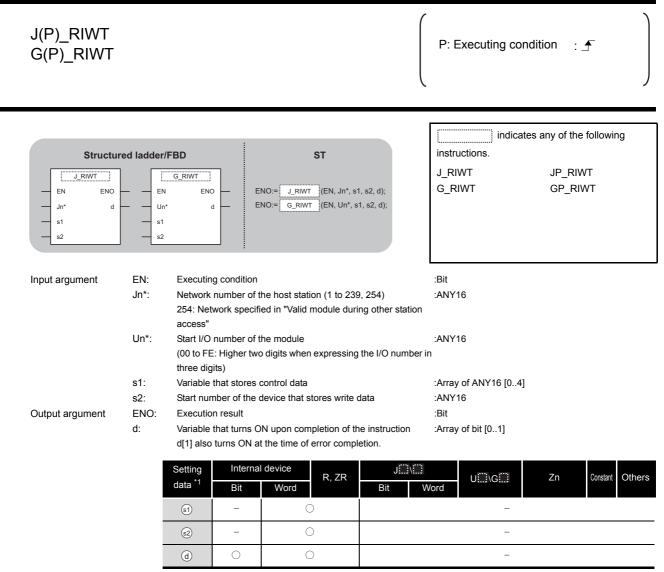
END_IF;

MODULE DEDICATED INSTRUCTION

5.4.2 RIWT instruction

J_RIWT, G_RIWT

CC-Link CC IE C CC IE F



*1: Local devices and file registers per program cannot be used as setting data.

☆ Function

This instruction writes the data for the specified number of points to the buffer memory of the CC-Link module or the device of the programmable controller CPU module on the specified station.

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|---------|-------------------------------------|--|---|--------------|
| (5) [0] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) For error codes when target station is anything other than master/local module, refer to the manual of the target station. | - | System |
| st][1] | Target station No. | Specify the station number of the target station | 0 to 64 ^{*1} 0 to 120 | |
| st)[2] | Access code and attribute code | Specify the access code and attribute code of the device to be read. | Refer to (1) and (2). | User |
| গ্ৰ [3] | Buffer memory address or device No. | Specify the start address of the buffer memory or the start number of the device. | Within the device range*2 | |
| st][4] | Number of write points | Specify the number of data to be written (in units of words). | 1 to 10 ^{*3} 1 to 480 ^{*4} | |

- *1 : For G(P)_RIWT, the setting range shall be 0 to 64.
- *2: For details, refer to the manual for the local station or the intelligent device station to which data are written.
- When the random access buffer is specified, specify the start address of the random access buffer as 0.*3 :When writing device data to the programmable controller CPU other than the QCPU (Q mode), QCPU (A
- mode) or QnACPU/AnUCPU, the setting range shall be 1 to 10 words.*4 : The value indicates the maximum number of data to be written.
- The value indicates the maximum number of data to be written.
 Specify the value within the buffer memory capacity of the local station or the intelligent device station, or the send buffer area setting range set by a parameter.

(1) Buffer memory of the CC-Link module

| Buffer memory category | | Access code | Attribute code |
|-------------------------------------|-----------------------|-------------|----------------|
| Buff | 00н | | |
| | Random access buffer | 20н | |
| | Remote input | 21н | |
| Buffer in a master or local station | Remote output | 22н | 04н |
| | Remote register | 24н | |
| | Link special relay | 63н |] |
| | Link special register | 64н | |

| Device ^{*1} | Name | Devi | ce type | Unit | Access code ^{*2} | Attribute code*2 |
|---------------------------------|------|------|---------|-------------|---------------------------|------------------------------|
| Device | Name | Bit | Word | Onic | | Attribute code ^{*2} |
| Input relay | Х | 0 | - | Hexadecimal | 01н | |
| Output relay | Y | 0 | - | Hexadecimal | 02н | |
| Internal relay | М | 0 | - | Decimal | 03н | |
| Latch relay | L | 0 | - | Decimal | 83н | |
| Link relay | В | 0 | - | Hexadecimal | 23н | |
| Timer (contact) | Т | 0 | - | Decimal | 09н | |
| Timer (coil) | Т | 0 | - | Decimal | 0Ан | |
| Timer (current value) | Т | - | 0 | Decimal | 0Сн | |
| Retentive timer (contact) | ST | 0 | - | Decimal | 89н | |
| Retentive timer (coil) | ST | 0 | - | Decimal | 8Ан | |
| Retentive timer (current value) | ST | - | 0 | Decimal | 8CH | 05н |
| Counter (contact) | С | 0 | - | Decimal | 11н | |
| Counter (coil) | С | 0 | - | Decimal | 12н | |
| Counter (current value) | С | - | 0 | Decimal | 14н | |
| Data register ^{*3} | D | - | 0 | Decimal | 04н | 1 |
| Link register ^{*3} | W | - | 0 | Hexadecimal | 24н | |
| File register | R | - | 0 | Decimal | 84н | 1 |
| Link special relay | SB | 0 | - | Hexadecimal | 63н | 1 |
| Link special register | SW | - | 0 | Hexadecimal | 64н | 1 |
| Special relay | SM | 0 | - | Decimal | 43н | 1 |
| Special register | SD | - | 0 | Decimal | 44н | 1 |

(2) Device memory of the programmable controller CPU module

*1 : Devices other than those listed above cannot be accessed.

When accessing a bit device, specify it with 0 or a multiple of 16.

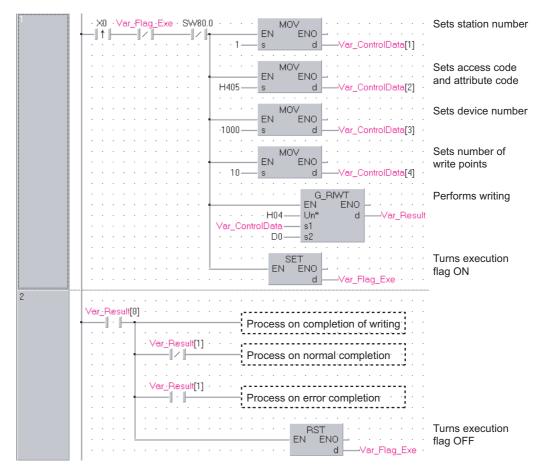
*2: For access code/attribute code when target station is anything other than master/local module, refer to the manual of the target station.

*3 : D65536 and the following devices of extended data registers as well as W10000 and the following devices of extended link registers cannot be specified.

Program Example

The following program stores 10-word data, which are stored in the devices starting from D0, to the devices starting from D1000 of the number 1 local station connected to the master module mounted on the I/O numbers from X/Y40 to X/Y5F when X0 turns ON.

(When the refresh device of the link special register (SW) is set to SW0.)



[Structured ladder/FBD]

| MOV(TRUE, H0405, Var_ControlData[2]); | , |
|--|---|
| MOV(TRUE, 1000, Var_ControlData[3]); MOV(TRUE, 10, Var_ControlData[4]); | |
| G_RIWT(TRUE, H04, Var_ControlData, D0 | |
| SET(TRUE, Var_Flag_Exe); | (* Turns execution flag ON *) |
| END_IF; | |
| IF(Var_Result[0]=TRUE)THEN | (* Execution finished *) |
| (* Process on completion of writing *) | |
| IF(Var_Result[1]=FALSE)THEN | (* Normal completion *) |
| (* Process on normal completion *) | |
| ELSE | (* Error completion *) |
| (* Process on error completion *) | |
| END_IF; | |
| DOT/TOUE Ver Flee Evely (* Turne evention f | |

RST(TRUE, Var_Flag_Exe); (* Turns execution flag OFF *) END_IF;

G_RIRCV

CC-Link

5.4.3 RIRCV instruction

| | | | | | (| r | |
|--|------------|------------|---------------|--|--------------------|------------------------------|-------------------------|
| G(P)_RIRC\ | / | | | | | P: Executing cor | ndition : 🕂 |
| Structured I | | BD | | ST | | instructions. | es any of the following |
| G_RIF — EN — Un* — s1 — s2 | ENO | - EN | O:= G_RIRC | ₩ (EN, Un*, s1, s2 | . d1, d2); | G_RIRCV | GP_RIRCV |
| Input argument | EN: | Executing | g condition | | | :Bit | |
| | Un*: | | - | e module digits when express | ing the I/O number | :ANY16 in | |
| | s1: | Variable 1 | hat stores c | ontrol data | | :Array of ANY16 [04] | |
| | s2: | Variable 1 | hat stores in | terlock signal | | :Array of ANY16 [02] | |
| Output argument | ENO: | Execution | | | | :Bit | |
| | d1: d2: | Variable | that turns Of | evice that stores rea I upon completion o t the time of error co | f the instruction | :ANY16 :Array of bit [01] | |
| | | Setting | Internal | device R, ZR | | U\G | Zn Constant C |
| | | data *1 | | VVOI (C) | Bit | Word | |
| | | data *1 | Bit _ | 0 | | - | |
| | | | | | | - | |
| | _ | s1) | - | 0 | | | |

*1: Local devices and file registers per program cannot be used as setting data.

Grant Function

This instruction automatically performs handshaking with an intelligent device station and reads data from the buffer memory of the specified intelligent device station.

This instruction is applicable with a module having a handshake signal, such as the AJ65BT-R2(N).

G_RIRCV

5

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|--------|--------------------------------|---|------------------------|--------------|
| s1 [0] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | - | System |
| s1[1] | Station No. | Specify the station number of the intelligent device station. | 0 to 64 | User |
| s1 [2] | Access code, Attribute code | Set '0004н'. | 0004н | User |
| s1 [3] | Buffer memory address | Specify the start address of the buffer memory. | *1 | User |
| s1 [4] | Number of read points | Specify the number of data to be read (in units of words). | 1 to 480 ^{*2} | User |

*1 : For details, refer to the manual for the intelligent device station from which data are read.

*2 : The value indicates the maximum number of data to be read.

Specify the value within the buffer memory capacity of the intelligent device station or the receive buffer area setting range set by a parameter.

| Device | Item | Setting data | Setting range | Setting side |
|--------|--------------------|--|---------------|--------------|
| 0.00 | b15 to b8 b7 to b0 | RY: Request device | 0 to 127 | User |
| s2 [0] | © [0] 0 RY | Set the high-order 8 bits to 0. | 0 | User |
| | b15 to b8 b7 to b0 | RX: Completion device | 0 to 127 | User |
| s2[1] | | RWr: Error code storage device | 0 to 15. FFн | User |
| | | Set FFH when no error code storage device exists. | 0 10 10,111 | 0301 |
| | | 0: Completes with the content of one device (RXn). | | |
| | b15 to b0 | 1: Completes with the content of two devices (RXn, | | |
| © [2] | Completion mode | RXn + 1). | 0/1 | User |
| | Completion mode | (RXn + 1 turns ON upon abnormal completion of | | |
| | | the instruction.) | | |

(1) Interlock signal storage device

*3 :

The same error code as that for the completion status of control data are stored in the error code storage device.

Program Example

The following program reads 11-word data, which are stored in buffer memory starting from the buffer memory address 400H of the number 63 intelligent device station (AJ65BT-R2(N)) connected to the master module mounted on the I/O numbers X/Y00 to X/Y1F, and stores the data in the devices starting from D40.

The interlock signal storage is set to request device: RY2, completion device: RX2, error code storage device: RWr2, and completion mode: 1.

(When the refresh device of the link special register (SW) is set to SW0.)

[Structured ladder/FBD]

| 1 | Var_Flag_Inst · Var_Flag_Exe · SW83.E · · · · MOV · · · · · · · · · · · · · · · · · · · | Sets station number |
|---|--|--|
| | 63 s d Var_ControlData[1] | |
| | MOV EN ENO s d | Sets access code and attribute code |
| | H400 s d Var_ControlData[3] | Sets buffer memory address |
| | MOV EN ENO s d | Sets number of read points |
| | H2 s d Var_InterlockData[0] | Sets request device |
| | H202 S d Var_InterlockData[1] | Sets completion device and error code storage area |
| | H1sH2 | Sets completion mode |
| | G_RIRCV EN ENO Un* d1Un* d1 _ | Performs readout |
| 2 | Var_Flag_Inst SW83.E MEP SET SET I I EN EN EN I I I I I | Turns execution flag ON |
| 3 | Var_Result[0] Var_Result[1] Image: Second | |
| | RST EN ENO dVar_Flag_Inst | Turns read request OFF |
| | EN ENO dVar_Flag_Exe | Turns execution flag OFF |

5

[ST] IF((Var_Flag_Inst=TRUE) (* Read request ON *) &(Var_Flag_Exe=FALSE) (* Execution flag *) (* Data link status of station number 63 *) &(SW83.E=FALSE))THEN (* Sets control data *) MOV(TRUE, 63, Var ControlData[1]); (* Sets station number *) (* Sets access code and attribute code *) MOV(TRUE,H4, Var_ControlData[2]); MOV(TRUE, H400, Var ControlData[3]); (* Sets buffer memory address *) MOV(TRUE, 11, Var_ControlData[4]); (* Sets number of read points *) (* Sets interlock signal storage device *) MOV(TRUE, H2, Var_InterlockData[0]); (* Sets request device *) MOV(TRUE, H202, Var_InterlockData[1]); (* Sets completion device and error code storage area *) MOV(TRUE, H1, Var_InterlockData[2]); (* Sets completion mode *) G RIRCV(TRUE, H00, Var ControlData, Var InterlockData, D40, Var Result); (* Performs readout *) END_IF; IF(MEP((Var_Flag_Inst=TRUE) & (SW83.E=FALSE)))THEN (* Read request is ON and data link status of station number 63 is OFF (rising pulse) *) SET(TRUE, Var_Flag_Exe); (* Turns execution flag ON *) END_IF; IF(Var_Result[0]=TRUE)THEN (* Execution finished *) IF(Var Result[1]=FALSE)THEN (* Normal completion *) (* Process on normal completion *) ELSE (* Error completion *) (* Process on error completion *) END_IF; RST(TRUE, Var Flag Inst); (* Turns read request OFF *) RST(TRUE, Var_Flag_Exe); (* Turns execution flag OFF *) END_IF;

5.4.4 RISEND instruction



| CC Link | |
|---------|---|
| CC-LINK | l |

| G(P)_RISEN | ID | | | | | P: Executing con | dition : <u>f</u> |
|---|--------------------|--|---|--|------------------|---|-------------------------------------|
| Structured Is G_RIS EN Un* s1 s2 | | | ST g_risend (EN | l, Un*, s1, s2, d1, | d2); | indicate instructions. G_RISEND | s any of the following GP_RISEND |
| Input argument | Un*: s1: s2: | (00 to FE: High three digits) Variable that st | per of the modu ner two digits w tores control d tores interlock | hen expressing t | he I/O number i | :Bit :ANY16 n :Array of ANY16 [04] :Array of ANY16 [02] :Bit | |
| Output argument | d1: d2: | Start number o Variable that tu | of the device th urns ON upon o | at stores write da completion of the ne of error compl | instruction | :ANY16 :Array of bit [01] | |
| | | lata ^{*1} B | | | J 🛄 \ 🛄 Bit V | Vord UIII\GIII | Zn Constant C |
| | | GI | | 0 | | _ | |
| | | © - | | 0 | | | |
| | | | | | | | |
| | | d2 (|) | 0 | | - | |

G_RISEND

5

EDICATED

Grant Function

This instruction automatically performs handshaking with an intelligent device station and writes data to the buffer memory of the specified intelligent device station.

This instruction is applicable with a module having a handshake signal, such as the AJ65BT-R2(N).

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|---------|--------------------------------|---|------------------------|--------------|
| st) [0] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | - | System |
| s1[1] | Station No. | Specify the station number of the intelligent device station. | 0 to 64 | User |
| s1 [2] | Access code, Attribute code | Set '0004н'. | 0004н | User |
| s1 [3] | Buffer memory address | Specify the start address of the buffer memory. | *1 | User |
| s1 [4] | Number of write points | Specify the number of data to be written (in units of words). | 1 to 480 ^{*2} | User |

*1 : For details, refer to the manual for the intelligent device station to which data are written.

*2 : The value indicates the maximum number of data to be written.

Specify the value within the buffer memory capacity of the intelligent device station or the receive buffer area setting range set by a parameter.

| Device | Item | Setting data | Setting range | Setting side |
|--------|---------------------------------------|--|---------------|--------------|
| \sim | b15 to b8 b7 to b0 | RY: Request device | 0 to 127 | User |
| s2 [0] | © [0] 0 RY | Set the high-order 8 bits to 0. | 0 | User |
| | © [1] b15 to b8 b7 to b0 RWr *3 RX | RX: Completion device | 0 to 127 | User |
| s2 [1] | | RWr: Error code storage device | 0 to 15, | User |
| | | Set FFH when no error code storage device exists. | FFн | USEI |
| | | 0: Completes with the content of one device (RXn). | | |
| | b15 to b0 | 1: Completes with the content of two devices (RXn, | | |
| s2 [2] | Completion mode | RXn + 1). | 0/1 | User |
| | Completion mode | (RXn + 1 turns ON upon abnormal completion of | | |
| | | the instruction.) | | |

(1) Interlock signal storage device

*3 : The same error code as that for the completion status of control data are stored in the error code storage device.

The following program writes 1-word data of D10 to the buffer memory address 111H of the number 63 intelligent device station (AJ65BT-R2(N)) which is connected to the master module mounted on the I/O numbers from X/Y00 to X/Y1F.

The interlock signal storage settings are set to request device: RY4, completion device: RX4, error code storage device: RWr1, and completion mode: 1.

(When the refresh device of the link special register (SW) is set to SW0.)

[Structured ladder/FBD]

| 1 | Var_Flag.Inst · Var_Flag.Exe · SW83.E · · · · MOV · · · · · · · · · · · · · · · · · · · | Sets station number |
|---|---|--|
| | s d | |
| | H4— EN H4— s d —Var_ControlData[2] | Sets access code and attribute code |
| | | |
| | | Sets buffer memory address |
| | H111 s d Var_ControlData[3] | |
| | MOV | Sets number of write |
| | | points |
| | □ · · · · · · · · · · · · · · · · · · · | points |
| | | Sets request device |
| | EN ENO | |
| | | |
| | | Sets completion device |
| | EN ENO | and error code storage |
| | H104—sd —Var_interlockData[1] | area device |
| | MOV | Sets completion mode |
| | · · · · · · · · · · · · · · · · · · · | Cete completion mede |
| | ····································· | |
| | | Sets data to be written |
| | · · · · · · · · · · · · · · · · · · · | to intelligent device |
| | dD10 | station |
| | · · · · · · · · · · · · · · · · · · · | |
| | GP_RISEND | Performs writing |
| | H00 Un* d1 D10 · · · · · | |
| | s1 d2 | |
| | | |
| | | |
| 2 | Var_Flag_Inst · SW83.E · MEP · · · · · · · · · · · · · · · · · · · | Turns execution flag ON |
| 3 | Var_Result[0] · Var_Result[1] · · · · · · | |
| 5 | Process on normal completion | |
| | Var_Result[1] | |
| | Process on error completion | |
| | BST | Turns write request OFF |
| | | Turns write request OFF |
| | d —Var_Flag_Inst | |
| | · · · · · · · · · · · · · · · · · · · | |
| | RST EN ENO | Turns execution |
| | d —Var_Flag_Exe | flag OFF |

[ST] IF((Var_Flag_Inst=TRUE) (* Write request ON *) &(Var_Flag_Exe=FALSE) (* Execution flag *) (* Data link status of station number 63 *) &(SW83.E=FALSE))THEN (* Sets control data *) MOV(TRUE, 63, Var ControlData[1]); (* Sets station number *) MOV(TRUE, H4, Var_ControlData[2]); (* Sets access code and attribute code *) MOV(TRUE, H111, Var ControlData[3]); (* Sets buffer memory address *) MOV(TRUE, 1, Var_ControlData[4]); (* Sets number of write points *) (* Sets interlock signal storage device *) MOV(TRUE, H4, Var_InterlockData[0]); (* Sets request device *) MOV(TRUE, H104, Var_InterlockData[1]); (* Sets completion device and error code storage area device *) MOV(TRUE, H1, Var_InterlockData[2]); (* Sets completion mode *) (* Sets data to be written to intelligent device station *) MOV(TRUE, 11, D10); GP_RISEND(TRUE, H00, Var_ControlData, Var_InterlockData, D10, Var_Result); (* Performs writing *) END IF; IF(MEP((Var Flag Inst=TRUE) & (SW83.E=FALSE)))THEN (* Write request is ON and data link status of station number 63 is OFF (rising pulse) *) SET(TRUE, Var_Flag_Exe); (* Turns execution flag ON *) END_IF; IF(Var Result[0]=TRUE)THEN (* Execution finished *) IF(Var_Result[1]=FALSE)THEN (* Normal completion *) Process on normal completion *) ELSE (* Error completion *) Process on error completion *) END IF; (* Turns write request OFF *) RST(TRUE, Var_Flag_Inst); RST(TRUE, Var_Flag_Exe); (* Turns execution flag OFF *)

END_IF;

G_RIFR

5.4.5 RIFR instruction

| | | | | | | | | CC-L | .ir |
|--|-------------------|-------------------------------------|---|---|-------------------|--------------------------|------------------------------|-----------------|-----|
| G(P)_RIFR | | | | | | P: Executing col | ndition : 🕤 | | _ |
| Structured la G_RI Un* n1 n2 n3 | | - | NO:= <u>G_</u> F | ST IFR (EN, Un*, n1, n2, r | 13, d); | indications. G_RIFR | tes any of the fo GP_RIFR | llowin | g |
| Input argument | EN: Un*: | Start I/O | g condition number of t :: Higher two | he module 9 digits when expressing | the I/O number in | :Bit :ANY16 | | | |
| | n1: n2: | Random Offset va | t device sta access buff lue of speci | tion number (1 to 64) er specification (FFн) fied intelligent device au iffer of the master statio | | :ANY16 :ANY16 | | | |
| Output argument | n3: ENO: d: | No proce Execution | essing is per n result | ts (0 to 4096) formed with setting '0'. levice that stores read d | ata | :ANY16 :Bit :ANY16 | | | |
| | | Setting data ^{*1} n1 | Internal Bit | device Word R, ZR | J\ Bit W | /ord | | onstant K, H | 0 |
| | t | n2 | 0 | 0 | | - | | 0 | |
| | | n3 | 0 | 0 | | _ | | 0 | |

*1: Local devices and file registers per program cannot be used as setting data.

_

Grant Function

This instruction reads data from the auto-refresh buffer of the specified station.

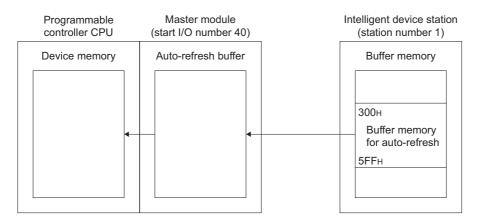
0

d

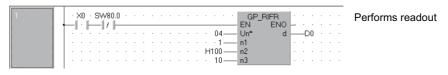
The instruction is applicable with a module having an auto-refresh buffer, such as the AJ65BT-R2(N).

The following program reads out 10-word data from buffer memory starting from the offset value 100 of the auto-refresh buffer of the master module (400H in the intelligent device station) and stores the data in the devices starting from D0 when X0 turns ON.

(When the refresh device of the link special register (SW) is set to SW0.)



[Structured ladder/FBD]



[ST]

IF((X0=TRUE) & (SW80.0=FALSE))THEN

GP_RIFR(TRUE, H04, 1, H100, 10, D0); END_IF; (* Performs readout *)

G_RITO

5.4.6 RITO instruction

| | | | CC-Link |
|-----------------------------|-------------|--|--|
| G(P)_RITO | | | P: Executing condition : |
| Structured I | adder/FB | D ST | indicates any of the following instructions. |
| EN Un* n1 n2 n3 | ENO d | ENO:= (EN, Un*, n1, n2, n3, d); | |
| Input argument | EN: Un*: | Executing condition Start I/O number of the module (00 to FE: Higher two digits when expressing the I/ | :Bit :ANY16 /O number in |
| | n1: | three digits) Intelligent device station number (1 to 64) Random access buffer specification (FFн) | :ANY16 |
| | n2: | Offset value of specified intelligent device auto-ref or random access buffer of the master station | fresh buffer :ANY16 |
| | n3: | Number of write points | :ANY16 |
| Output argument | ENO: d: | Execution result Start number of the device that stores write data | :Bit :ANY16 |
| | | Setting data *1 Bit Word R, ZR E | JIII\III Bit Word UIII\GIII Zn Constant - Constant K, H Oth |

5

0

_

_

_

n2

n3

d

 \bigcirc

 \bigcirc

_

This instruction writes the data to the auto-refresh buffer of the specified station.

0

0

0

The instruction is applicable with a module having an auto-refresh buffer, such as the AJ65BT-R2(N).

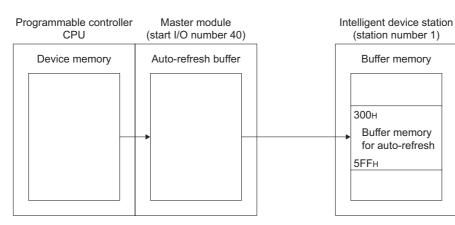
_

_

_

*1: Local devices and file registers per program cannot be used as setting data.

The following program write 10-word data which are stored in the devices starting from D0 into buffer memory starting the offset value 100 of the auto-refresh buffer of the master module (400H in the intelligent device station) when X0 turns ON.



(When the refresh device of the link special register (SW) is set to SW0.)

[Structured ladder/FBD]

| 1 | | | | | | | | | | | | | | | | | | GP_F | RITO ENO | | | | | | : | Performs |
|---|---|---|---|--|---|---|---|---|---|---|---|---|---|---|---|---|-----|------|-------------|---|----|----|---|---|---|----------|
| | | | | | | | | | | | | | | | | | 04 | | d | H | -[| D0 | | • | | whiting |
| | | | | | | | | | | | | | | | | | 100 | | | | | | | ÷ | | |
| | • | • | • | | • | • | • | • | • | • | • | • | • | • | • | • | 10 | n3 | | • | • | | • | • | • | |



IF((X0=TRUE) & (SW80.0=FALSE))THEN GP_RITO(TRUE, H04, 1, H100, 10, D0); END_IF;

(* Performs writing *)

G_RLPASET

5.4.7 RLPASET instruction

| G(P)_RLPA | SET | | | | P: Executing condition : |
|--|-------------|-------------------------------|--------------------------|--|------------------------------|
| | | | | | |
| | | | | | indicates any of the followi |
| | | | | | instructions. |
| | | | | | G_RLPASET GP_RLPASET |
| Structured I G RLP EN Un* s1 s2 s3 s4 | | | NO:= G_RLP/ | ST ASET (EN, Un*, s1, s2, s3, s4, s5, d); | |
| Input argument | EN: Un*: | Start I/C (00 to F | | he module o digits when expressing the I/O | :Bit :ANY16 number in |
| | s1: | three di Variable | gits) e that stores o | control data | :Array of ANY16 [07] |
| | s2: | | | slave station setting data | :Array of ANY16 [063] |
| | s3: | | | reserved station specification da | |
| | s4: s5: | | | error invalid station specificatior send/receive and auto-refresh b | |
| | 30. | | nent data | | |
| Output argument | ENO: | | on result | | :Bit |
| | d: | | | N upon completion of the instru t the time of error completion. | ction :Array of bit [01] |
| | | Setting data ^{*1} | Internal Bit | device R, ZR Word Bit | U∭\G∭ Zn Constant |
| | | (s1) | - | 0 | - |
| | | s2 | - | 0 | - |
| | Ī | s 3 | - | 0 | - |
| | Ī | <u>\$4</u> | - | 0 | - |
| | | | | | |
| | | s5 | - | 0 | - |

Grant Function

This instruction sets the network parameters to the master station and starts up the data link.

AODULE DEDICATED

G_RLPASET

Control Data

| Device | Item | Setting data | Setting range*2 | Setting side |
|--------|---|--|-----------------|--------------|
| s1 [0] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| ৰা[1] | Setting flag | Specify the validity of each setting data from @ to @ . 0: Invalid ^{*1} 1: Valid b15 b14b13 b4 b3 b2 b1 b0 Slave station setting data(s2) Reserved station specification data(s3) Error invalid station specification data(s4) Send/receive and auto-refresh buffer assignment data(s5) 00: Remote net (Ver. 1 mode) 01: Remote net (Ver. 2 mode) 10: Remote net (Ver. 2 mode) 11: Cannot be used | _ | User |
| s1 [2] | Number of connected modules | Set the number of connected slave stations. | 1 to 64 | User |
| s1 [3] | Number of retries | Set the number of retries to be performed to a communication error station. | 1 to 7 | User |
| ঃা [4] | Number of automatic return modules | Set the number of slave stations that can be returned in one link scan. | 1 to 10 | User |
| st [5] | Operation specification when CPU is down | Specify the data link status when a master station programmable controller CPU error occurs. 0: Stop 1: Continue | 0, 1 | User |
| st [6] | Scan mode specification | Specify the link scan mode for sequence scan. 0: Asynchronous 1: Synchronous | 0, 1 | User |
| s1[7] | Delay time specification | Set '0' for the delay time. | 0 | User |

*1: For the setting data for which invalid is specified, default parameter is applied.

*2: Setting a value outside the setting range results in error completion of the instruction.

| | (1) Slave stat | ion setting data | | | | |
|----------------|-----------------------|--|---|---|---------------|--------------|
| Device | Item | Settin | g data | | Setting range | Setting side |
| | | Set the slave station type, the number | per of occupied slave | e stations, and | | |
| | | the station number as shown below | r. | | | |
| | | Default parameter setting is '0101H number of occupied slave stations: compatible remote I/O station)' | →Station nu →Number of →Type of s to 0140H (station nu | of occupied slave stations lave station mber: 1 to 64, | - | |
| | | Setting of station number | | | | |
| | | 1 to 64 (BIN setting) | | | 1 to 40н | |
| | | Setting of the number of occupied s | slave stations | | | |
| | | Number of occupied slave | Setting | I | | |
| O | | stations | <u></u> 1н | | 1 to 4н | |
| s2 [0] | Setting for 1 to 64 | 1 station 2 stations | <u>1н</u> 2н | | 1 10 411 | 11 |
| to | modules ^{*3} | 3 stations | 3н | | | User |
| s2 [63] | | 4 stations | <u></u> | | | |
| | | | | | | |
| | | Setting of slave station type ^{*4} | | | | |
| | | Type of slave sta | tion | Setting | | |
| | | Ver.1 compatible remote I/O statio | n | 0н | | |
| | | Ver.1 compatible remote device st | ation | 1н | | |
| | | Ver.1 compatible intelligent device | station | 2н | | |
| | | Ver.2 compatible single remote de | vice station | 5н | | |
| | | Ver.2 compatible single intelligent | device station | 6н | 0 to Fн | |
| | | Ver.2 compatible double remote de | evice station | 8н | | |
| | | Ver.2 compatible double intelligent | t device station | 9н | | |
| | | Ver.2 compatible quadruple remote | | Вн | | |
| | | Ver.2 compatible quadruple intellig | | Сн | | |
| | | Ver.2 compatible octuple remote d | | Ен | | |
| | | Ver.2 compatible octuple intelligen | t device station | Fн | | |
| | | | | | | |

*3: Set the same number which was set for Number of connected modules in the control data.

*4 : Setting a value outside the setting range in the setting of slave station type results in error completion of the instruction.

G_RLPASET

(2) Reserved station specification data

| Device | Item | Setting data | | | | | | | | | | | Setting range | Setting side |
|--------|---------------------------|---|--------------|-----|-------|--------|---------|---------|----------|---------|--------|---|---------------|--------------|
| | | Specify the r 0: Not sp 1: Specifi | ecifie ed | d | | | | | | | | | | |
| s3 [0] | Specification for 1 to 64 | (-2)[0] | b15 | b14 | b13 | b12 | to | b3 | b2 | b1 | b0 | ו | | |
| to | stations ^{*5} | (s3)[0] | 16 | 15 | 14 | 13 | to | 4 | 3 | 2 | 1 | | - | User |
| 63 [3] | stations - | (s3)[1] | 32 | 31 | 30 | 29 | to | 20 | 19 | 18 | 17 | | | |
| | | (s3[2] | 48 | 47 | 46 | 45 | to | 36 | 35 | 34 | 33 | | | |
| | | s3[3] | 64 | 63 | 62 | 61 | to | 52 | 51 | 50 | 49 | | | |
| | | | | 1 | to 64 | in the | table i | ndicate | es a sta | ation n | umber. | | | |
| | | Default parameter setting is '0: Not specified' for all stations. | | | | | | | | | | | | |

*5 : Set the parameter up to the largest station number set in the slave station setting data.

*6: Set the parameter only to the start station number of the module for the remote station/local station/ intelligent device station that occupies two or more stations.

(3) Error invalid station specification data

| Device | Item | | | | | Settin | g data | a | | | | | Setting range | Setting side |
|---------------|-------------------------------|---|---|-----|---------|------------------|---------|---------|----------|----------|--------|--|---------------|--------------|
| | | Specify the 0 0: Not sp 1: Specif | ecifie | | statio | n. ^{*8} | | | | | | | | |
| s4 [0] | On a sifile stime for 4 to 04 | | b15 | b14 | b13 | b12 | to | b3 | b2 | b1 | b0 | | | |
| to | Specification for 1 to 64 | (s4)[0] | 16 | 15 | 14 | 13 | to | 4 | 3 | 2 | 1 | | _ | User |
| | stations ^{*7} | (s4)[1] | 32 | 31 | 30 | 29 | to | 20 | 19 | 18 | 17 | | | 0000 |
| s4 [3] | | s4[2] | 48 | 47 | 46 | 45 | to | 36 | 35 | 34 | 33 | | | |
| | | s4[3] | 64 | 63 | 62 | 61 | to | 52 | 51 | 50 | 49 | | | |
| | | | | 1 | 1 to 64 | in the | table i | ndicate | es a sta | ation nu | umber. | | | |
| | | Default para | Default parameter setting is '0: Not specified' for all stations. | | | | | | | | | | | |

*7: Set the parameter up to the largest station number set in the slave station setting data.

*8 : Set the parameter only to the start station number of the module for the remote station/local station/ intelligent device station that occupies two or more stations.

Reserved station specification has a priority when an error invalid station and reserved station are specified for the same station.

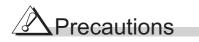
(4) Send/receive and auto-refresh buffer assignment data

| Device | Item | Setting data | Setting range | Setting side |
|--------------|-----------------------------|---|---|--------------|
| 63 [0] | Specification for 1 | Specify the buffer memory size assignment at transient transmission for local stations and intelligent device stations. | Send/receive buffer ^{*10} : Он (no setting) 40н to 1000н 0 (word) (no setting) 64 to 4096 (words) | |
| to © [77] | to 26 modules ^{*9} | (s5)[75] Send buffer size (s5)[76] Receive buffer size (s5)[77] Auto-refresh buffer size Default parameter setting is 'send buffer size: 40H, receive | Auto-refresh buffer ^{*11} : Он (no setting) 80н to 1000н 0 (word) (no setting) 128 to 4096 (words) | User |
| | | buffer size: 40H, auto-refresh buffer size: 80H'. | | |

*9: Set the assignment data, in ascending order, for the stations set for a local station or intelligent device station in the slave station setting data.

*10 : Keep the total of the send/receive buffer size within 1000H (4096 (words)). Specify the size added seven words to the size of send/receive data as the send/receive buffer size. Setting a value outside the setting range results in error completion of the instruction.

*11: Keep the total of the auto-refresh buffer size within 1000н (4096 (words)).
 Specify the necessary auto-refresh buffer size for each intelligent device station.
 Setting a value outside the setting range results in error completion of the instruction.

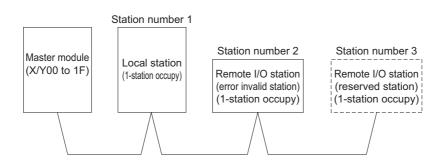


The RLPASET instruction is applicable to the QJ61BT11 of which the function version is B and the first five digits of the serial number are '03042' or higher.

The QJ61BT11N and LJ61BT11 is compatible with the RLPASET instruction.

Program Example

The following program sets the network parameter to the master module mounted on the I/O number X/Y00 to X/Y1F, and starts up the data link.



[Structured ladder/FBD]

| 1 | SM400 | Reads SB0040 to SB01FF |
|---|--|--|
| | 28 n3 EN ENO H0 n1 d H640 n2 SW40 448 n3 SW40 | Reads SW0040 to SW01FF |
| 2 | SM402 SB6E: EN EN I · I <td>Parameter setting command</td> | Parameter setting command |
| 3 | Ver_Flag_Inst | Clears completion status |
| | EN ENO 15 s d Var_ControlData[1] | Sets all of setting flags to Valid |
| | | Sets number of connected modules |
| | EN ENO 3 — s d —Var_ControlData[3] | Sets number of retries Sets number of automatic return |
| | EN ENO | modules Sets operation specification when |
| | | CPU is down to stop Sets scan mode specification to asynchronous |
| | | Set delay time specification |
| 4 | Ver_Flag_Inst H2101 S d Ver_SlaveStation[0] | First module: local station, 1-station occupy, station number 1 |
| | H102 s d Var_SlaveStation[1] | Second module: Remote I/O station, 1-station occupy, station number 2 |
| | H103 s d Var_SlaveStation[2] | Third module: Remote I/O station, 1-station occupy, station number 3 |
| 5 | Var_Flag_Inst · · · · · · · · · · · · · · · · · · · | Reserved station specification: station number 3 |
| 6 | Var_Flag_Inst · · · · · · · · · · · · · · · · · · · | Error invalid station specification: station number 2 |
| 7 | Ver_Flag_Inst | First module: local station, send buffer 100 words |
| | EN MOV 100 s d Var_BufferSize[1] | Receive buffer 100 words |
| | 0 s d Var_BufferSize[2] | Auto-refresh buffer 0 word |
| 8 | Var_Flag_Inst | Performs parameter setting and data link start |
| 9 | Var_Result[0] | Turns parameter setting command OFF |
| | Var_Result[1] | Refresh command |
| | Var_Result[1] Process on error completion | Control program start command |

5.4 Network Dedicated Instruction 5.4.7 RLPASET instruction [ST] FROM(TRUE, H0, H5E4, 28, K4SB40); (* Reads SB0040 to SB01FF *) FROM(TRUE, H0, H640, 448, SW40); (* Reads SW0040 to SW01FF*) IF((SM402=TRUE) & (SB6E=TRUE))THEN (* Parameter setting command *) SET(TRUE, Var Flag Inst); END IF: IF(Var_Flag_Inst=TRUE)THEN (* Parameter setting command ON *) MOV(TRUE, 0, Var ControlData[0]); (* Clear completion status *) MOV(TRUE, 15, Var_ControlData[1]); (* Sets all of setting flags to Valid *) MOV(TRUE, 3, Var_ControlData[2]); (* Sets number of connected modules *) MOV(TRUE, 3, Var ControlData[3]); (* Sets number of retries *) MOV(TRUE, 1, Var ControlData[4]); (* Sets number of automatic return modules *) MOV(TRUE, 0, Var_ControlData[5]); (* Sets operation specification when CPU is down to stop *) MOV(TRUE, 0, Var_ControlData[6]); (* Sets scan mode specification to asynchronous *) MOV(TRUE, 0, Var ControlData[7]); (* Set delay time specification *) MOV(TRUE, H2101, Var_SlaveStation[0]); (* First module: local station, 1-station occupy, station number 1 *) MOV(TRUE, H0102, Var SlaveStation[1]); (* Second module: Remote I/O station, 1-station occupy, station number 2*) MOV(TRUE, H0103, Var SlaveStation[2]); (* Third module: Remote I/O station, 1-station occupy, station number 3 *) MOV(TRUE, H4, Var_ReservedStation[0]); (* Reserved station specification: station number 3 *) MOV(TRUE, H2, Var ErrorInvalidStation[0]); (* Error invalid station specification: station number 2 *) MOV(TRUE, 100, Var BufferSize[0]); (* First module: local module, send buffer 100 words *) MOV(TRUE, 100, Var BufferSize[1]); (* Second module: local station, receive buffer 100 words *) MOV(TRUE, 0, Var_BufferSize[2]); (* Third module: local station, auto-refresh buffer 0 words *) GP_RLPASET(TRUE, H00, Var_ControlData, Var_SlaveStation, Var ReservedStation, Var ErrorInvalidStation, Var BufferSize, Var Result); (* Performs parameter setting *) END IF; IF(Var Result[0]=TRUE)THEN (* Execution finished *) IF(Var Result[1]=FALSE)THEN (* Normal completion *) (* Refresh command *) SET(TRUE, SB3); SET(TRUE, Var Flag Exe); (* Control program start command *) ELSE (* Error completion *) (* Process on error completion *) END_IF; RST(TRUE, Var Flag Inst); (* Turns parameter setting command OFF *) END IF;

5.4.8 READ instruction

J_READ, G_READ

CC IE C CC IE F NET/H Ether

J(P)_READ P: Executing condition · 🔺 G(P)_READ indicates any of the following Structured ladder/FBD ST instructions. J_READ JP_READ J_READ G_READ ſ G_READ GP_READ ΕN ENO ΕN ENO (EN, Jn*, s1, s2, d1, d2); ENO:= J_READ d1 d1 Un Jn* ENO:= G_READ (EN, Un*, s1, s2, d1, d2); s1 d2 s1 d2 s2 s2 Input argument EN: Executing condition ·Bit Jn*: Network number of the host station (1 to 239, 254) :ANY16 254: Network specified in "Valid module during other station access" Start I/O number of the module ·ANY16 Un*: (00 to FE: Higher two digits when expressing the I/O number in three digits) Variable that stores control data :Array of ANY16 [0..17] s1: Start number of the target station's device from which data are :ANY s2: read Output argument ENO: :Bit Execution result Start number of the host station's device that stores read data :ANY16 d1: d2. Variable that turns ON upon completion of the instruction :Array of bit [0..1] d2[1] also turns ON at the time of error completion. Setting Internal device J....\.... Others R, ZR U....\G.... Zn Constant *1 data Bit Word Word Bi _ (s1) _ (s2) \bigcirc _ (d1) _ _ d2) _ *1: Local devices and file registers per program cannot be used as setting data.

Gamma Function

This instruction reads data from a word device of another station.

Control Data

| Device | Item | | | Setting range | Setting side | | | |
|---------|------------------------------|------------------------------|--|--|---------------------------------------|-----------------|--------------------|--------|
| | | | b15 | to b7 | to 0 | b0 | | |
| ্রা [0] | Error completion type | Spec 0: Cl st 1: Cl | ify the clock lock data at arting from | n type (bit 7) data setup status at the the time of error compl (a) [11]. the time of error comp | ror completion. ot set in the area | 0001н, 0081н | User | |
| s1[1] | Completion status | 0 Othe | er than 0 | pletion status is stored. Normal completion Error completion (erro | - | | - | System |
| | | Specify | | l used by the host statior | 1. | | | |
| | | | | scription | Set | ting value | | |
| s1 [2] | Channel used by host station | | SECNET/H | | 1 to 8 | | 1 to 10 | User |
| | | | | oller Network | 1 to 10 1 to 2 | | | |
| | | CC-L | ink IE Field | Network | <u> </u> | | | |
| | | Specif | y the type of | of the target station C | PU. | | | |
| | | Sett | ing value | Desc | ription | | 0000н, | |
| | | Ethernet | 0000н | Target station CPU/hos data are the same as '0 | | CPU (Specified | 0000н, 03FFн | |
| | | Eth | 03FFн ^{*1} | Target station CPU/hos | t system (| CPU | | |
| s1 [3] | Target station's CPU type | | 0000н | Target station CPU/hos data are the same as '0 | | CPU (Specified | | User |
| | | MELSECNET/H CC-Link IE | 03E0н ^{*2} | Multi-CPU No. 1/target system) | station CI | PU (single CPU | 0000н, 03Е0н to | |
| | | LSE CC-L | 03E1н ^{*2} | Multi-CPU No. 2 | | | 03ЕЗн, | |
| | | ME | 03E2н ^{*2} | Multi-CPU No. 3 | | 03FFн | | |
| | | | 03E3н ^{*2} | Multi-CPU No. 4 | | | | |
| | | | 03FFн ^{*1} | Target station CPU/hos | t system (| CPU | | |
| s1 [4] | Target station network No. | | 239 : Netw | number of the target sta ork number ify this when 254 has b | | n Jn. | 1 to 239, 254 | User |
| | | Specify | the station | number of the target stat | ion. | | | |
| | | | | Setting value | | Description | | |
| | | | LSECNET/F | 1 | | 1 to 64 | | |
| | | - | ernet -Link IE Con | troller Network | | | | |
| | Torget atotics No. | | - | n is Universal model QCI | 1 to120 | 1 10 105 | Lloor | |
| st [5] | Target station No. | | | n is anything other than nodel QCPU | 1 to 125 | User | | |
| | | CC | -Link IE Field | | | | | |
| | | Master station 125 (7DH) | | | | | | |
| | | | Local statio station | n or the intelligent device | e | 1 to 120 | | |
| st [6] | _ | Reserv | ed | | | | 0 | User |

| Device | Item | | Setting data | | Setting range | Setting side |
|---------|--------------------------------|---|--|----------------------|---------------|--------------|
| | | 1 For instruction exec | | | | |
| | | Specify the number of instruction resends when the instruction is not | | | 0 to 15 | User |
| st][7] | Number of resends | completed within the | | | | |
| | | ② At instruction comp | oletion | | _ | System |
| | | The number of reser | nds (result) is stored. | | | oystem |
| | | Specify the monitoring | | | | |
| | | If the instruction is not | | | | |
| | | number of times specif | ied in 🗊 [7]. | | | |
| | | | | | | |
| | | De | escription | Setting value | | |
| | | | 0 to TCP retransmission | | | |
| | | | timer value: Monitoring is | | | |
| s1 [8] | Arrival monitoring time | | performed by the TCP | | 0 to 32767 | User |
| 0[0] | | Ethernet | retransmission timer value. | 0 to 16383 | | |
| | | | (TCP retransmission timer value + 1) to 16383: | | | |
| | | | Monitoring time (unit: | | | |
| | | | second) | | | |
| | | | 0: 10 seconds | | | |
| | | MELSECNET/H | 1 to 32767: 1 to 32767 | 0 to 32767 | | |
| | | CC-Link IE | seconds | 0 10 02101 | | |
| | | | | | | |
| | | Specify the number of | read data. | | | |
| | | De | escription | Setting value | | |
| | Read data length | Ethernet 1 to 960 | | | 1 to 8192 Use | User |
| s1 [9] | | MELSECNET/H (word) | | | | |
| | | CC-Link IE Field Network | | | | |
| | | CC-Link IE Controller Network (word) | | | | |
| | | | | | | |
| st [10] | - | Reserved | - | User | | |
| | | Valid/invalid status of the | ne data in the area starting fro | om 🗊 [12] is stored. | | _ |
| s1[11] | Clock set flag ^{*3} | 0: Invalid | - | System | | |
| | | 1: Valid | | | | |
| | | Clock data at the time | of error completion are stored | d in BCD format. | | |
| 0.000 | | b15 to | b8 b7 to | b0 | | |
| s1 [12] | Clock data at the time of | (s1) [12] Month (01) | | | | |
| to | error completion ^{*3} | (s1) [13] Hour (00⊢ (s1) [14] Second (00 | | | - | System |
| st [15] | | (s1) [14] Second (00 (s1) [15] Year (00н to 99н | | | | |
| | | | 00H (Sun.) to | | | |
| | | Network number of the | station where an error was o | leterted is stored | | |
| | Error-detected network No. | | or was detected at the host s | | | |
| s1[16] | | number is not stored.) | S. Was deteoled at the host s | | - | System |
| | | 1 to 239: Network num | ber | | | |
| | | | where an error was detected | is stored. | | |
| | | (However, when an err | or was detected at the host s | station, the network | | |
| | | number is not stored.) | | | | |
| | | Sotti | ng value | Description | | |
| ্রা[17] | | MELSECNET/H | - | to 64 | | |
| | Error-detected station No. *3 | | | | _ | System |
| | | Ethernet 1 to 120 CC-Link IE Controller Network - | | to 120 | – Syste | Gystelli |
| | | | | | | |
| | | CC-Link IE Field Network - | | | | |
| | | Master station 125 (7DH) | | | | |
| | | Local station or the intelligent device 1 to 120 | | | | |
| | 1 | | | | | |

- *1: Specification is possible when the host station is a network module or Ethernet module of function version D or later.
 - (Specification is not possible for other modules. An access is always made to the target station CPU.)
- *2 : Specification is possible when the versions of the QCPU and the network module on the host station and the target station are as indicated below.

(Specification is not possible for other modules. An access is always made to the target station CPU.)

- Network module: The first five digits of the serial number are '06092' or higher.
- QCPU: The first five digits of the serial number are '06092' or higher.
- *3 : Data are stored only when 1 is set in bit 7 of Error completion type ((i) [0]).

The following program reads out data from the devices from D250 to D254 in the station number 4 (target station) and stores the data to the devices from D700 to D704 of the station number 1 (host station).

Var_Flag_Inst Sets error MOV 1 ΕN ENO completion type H81 d ar_ControlData[0] s Sets channel used MO ENO ΕN by host station 1 d _ControlData[2] S Sets target station's MOV ΕN ENO CPU type ΗN d ar_ControlData[3] s Sets target station MO\ ΕN ENO network number ar_ControlData[4] · 1 d S Sets target station MC ΕN ENO number 4 d ControlData[5] S MΟV ENO ΕN 0 d ControlData[6] Sets monitoring MON ΕN ENO time Π _ControlData[8] \mathbf{S} d Sets data length MO\ ΕN ENO by the word 5 ControlData[9] d S MOV ΕN ENO Ω d ar_ControlData[10] s 2 Exe SB47 SW0A0.3 Sets number of Var Flag MOV ΕN ENO resends 5 /ar_ControlData[7] s d JP_READ Performs readout ΕN ENO d1 D700 Jn* 1 Var_ControlData s1 d2 Var Result D250 s^2 3 Var_Result[0] 1 Process on completion of readout Execution finished Var_Result[1] 121 Process on normal completion Normal completion Var_Result[1] ŀ Process on error completion Error completion Stores error code MOΛ ΕN ENO Var_ControlData[1] Var_ErrorCode d s

[Structured ladder/FBD]

| [ST] | |
|---|--|
| IF (LDP(TRUE,Var_Flag_Inst) THEN | |
| MOV(TRUE,H81,Var_ControlData[0]); | (* Sets error completion type *) |
| MOV(TRUE,1,Var_ControlData[2]); | (* Sets channel used by host station *) |
| MOV(TRUE,H0,Var_ControlData[3]); | (* Sets target station's CPU type *) |
| MOV(TRUE,1,Var_ControlData[4]); | (* Sets target station network number *) |
| MOV(TRUE,4,Var_ControlData[5]); | (* Sets target station number *) |
| MOV(TRUE,0,Var_ControlData[6]); | (* Osta and a 'tan'a a time *) |
| MOV(TRUE,0,Var_ControlData[8]); | (* Sets monitoring time *) |
| MOV(TRUE,5,Var_ControlData[9]); | (* Sets data length by the word *) |
| MOV(TRUE,0,Var_ControlData[10]); END IF; | |
| IF((Var_Flag_Exe=TRUE) AND (SB47=FALSE |) AND (SWOAD 3=FALSE)) THEN |
| MOV(TRUE, 5, Var_ControlData[7]); | |
| | 0,D700,Var_Result);(* Performs readout *) |
| END IF; | ,, |
| IF(Var_Result[0]=TRUE)THEN | (* Execution finished *) |
| (* Process on completion of reado | t *) |
| · | J |
| IF(Var_Result[1]=FALSE)THEN | (* Normal completion *) |
| (* Process on normal completion * | *) |
| ELSE | (* Error completion *) |
| (* Process on error completion *) | |
| MOV(TRUE, Var ControlData[1], V | /ar_ErrorCode);(* Stores error code *) |
| END IF; | _ // |
| | |

END_IF;

5.4.9 SREAD instruction

J_SREAD, G_SREAD

CC IE C CC IE F NET/H Ether

| J(P)_SREAD G(P)_SREAD | | | | | | | P: Exe | cuting con | dition | |
|---|-------------|--------------------------------------|-----------------------------|--|---------------|-------------------------|--------------------------------|---------------|---------------------------------|----------------------|
| Structured <u>JSREAD</u> EN ENO Jn* d1 s1 d2 s2 d3 | | BD SREAD ENO d1 d2 d3 | | NO:= J_SREA | | | instructic J_SREA G_SREA | ons. D | es any of th JP_SRI GP_SR | |
| Input argument | EN: Jn*: | Network | | he host static | | 254) g other station | :Bit :ANY16 | | | |
| | Un*: | access" Start I/O | number of t : Higher two | he module | | he I/O number | :ANY16 | | | |
| | s1: s2: | Variable f Start num | that stores of | control data arget station' | s device fror | n which data a | - | ANY16 [017] | l | |
| Output argument | ENO: | read Executior | n result | | | | :Bit | | | |
| ouput argumont | d1: | | | nost station's | device that | stores read da | read data :ANY16 | | | |
| | d2: | | | N upon com | | | :Array of I | oit [01] | | |
| | d3: | Variable | | at the time of N upon comp vice) | | | :Bit | | | |
| | | Setting data ^{*1} | Internal Bit | device Word | R, ZR | J\ | L Word | J\G | Zn | Constant Others |
| | | s1) | - | С |) | | | _ | | |
| | | s2 | _ | С |) | | | _ | | |
| | | d1 | - | C |) | | | - | | |
| | | d2 | 0 | С |) | | | - | | |
| | | d3 | 0 | С |) | | | _ | | |
| | _ | | | | *1: Local | devices and fi | le registers p | per program c | annot be u | sed as setting data. |

5-150

Grant Function

This instruction reads data from a word device of another station.

Control Data

For the control data of the SREAD instruction that reads the word device memory of another station, refer to READ instruction.

The control data of the SREAD instruction are the same as those of the READ instruction. Accordingly, this section omits the explanation.

The following program example of the SREAD instruction is different from that of the READ instruction by assigning the read notification device $_{(3)}$ at the end of arguments.

Var_Flag_Inst Sets error MOV ΕN ENO completion type H81 ar_ControlData[0] d s Sets channel used MOV ΕN ENO by host station 1 _ControlData[2] d Sets target MOV ΕN ENO station's CPU type HO _ControlData[3] d Sets target station MOV ΕN ENO network number 1. ar_ControlData[4] d Sets target station MOV ΕN ENO number 4d ar_ControlData[5] MOV ENO ΕN 0. _ControlData[6] d Sets monitoring MOV ΕN ENO time 0 ar ControlData[8] d Sets data length MOV ENO ΕN by the word 5 ar_ControlData[9] d MOV ENO ΕN n. d /ar_ControlData[10] s $\overline{2}$ √ar Flag Exe SB47 Sets number of SWRAR 2 MOV ΕN ENO resends 5 ControlData[7] s d J_SREAD Performs readout EN ENO Jn* d1 -D700 1 Var_Result Var_ControlDate s1 d2 D250 s2 d3 Var. Flag З Var_Result[0] Process on completion of readout Execution finished Ŀ Var_Result[1] Normal completion Process on normal completion 171 /ar_Result[1] Error completion Process on error completion - - - - - -- - - -Stores error code MO\ ΕN ENO Var_ControlData[1] d Var_ErrorCode s

[Structured ladder/FBD]

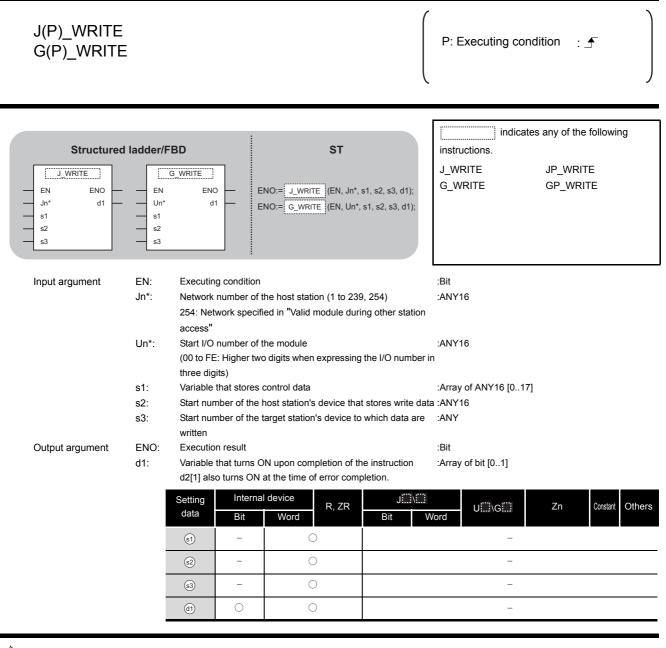
| [ST] | |
|---|--|
| IF (Var_Flag_Inst=TRUE) THEN | |
| MOV(TRUE,H81,Var_ControlData[0]); | (* Sets error completion type *) |
| MOV(TRUE,1,Var_ControlData[2]); | (* Sets channel used by host station *) |
| MOV(TRUE,H0,Var_ControlData[3]); | (* Sets target station's CPU type *) |
| MOV(TRUE,1,Var_ControlData[4]); | (* Sets target station network number *) |
| MOV(TRUE,4,Var_ControlData[5]); | (* Sets target station number*) |
| MOV(TRUE,0,Var_ControlData[6]); | |
| MOV(TRUE,0,Var_ControlData[8]); | (* Sets monitoring time *) |
| MOV(TRUE,5,Var_ControlData[9]); | (* Sets data length by the word *) |
| MOV(TRUE,0,Var_ControlData[10]); END IF; | |
| IF((Var_Flag_Exe=TRUE) AND (SB47=FALSE) | AND (SWIDAD 3=FALSE)) THEN |
| MOV(TRUE, 5, Var_ControlData[7]); | |
| J_SREAD(TRUE,1,Var_ControlData,D250,I | |
| | (* Performs readout *) |
| END_IF; | , , , , , , , , , , , , , , , , , , , |
| IF(Var_Result[0]=TRUE)THEN | (* Execution finished *) |
| (* Process on completion of readout *) | |
| | |
| IF(Var_Result[1]=FALSE)THEN | ("Normal completion ") |
| (* Process on normal completion *) | |
| ELSE | (* Error completion *) |
| (* Process on error completion *) | |
| · | |
| MOV(TRUE, Var_ControlData[1], Var | |
| END_IF; | |

END_IF;

5.4.10 WRITE instruction

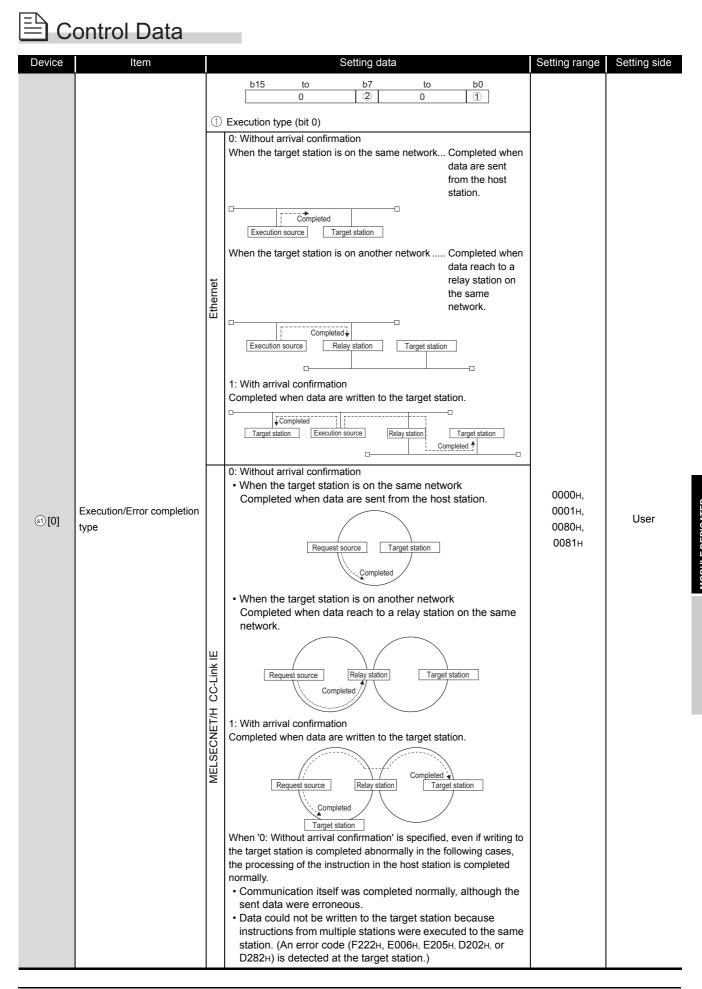
J_WRITE, G_WRITE

CC IE C CC IE F NET/H Ether



Grant Function

This instruction writes data to a word device of another station.



J_WRITE, G_WRITE

| Device | Item | | | Setting data | | Setting range | Setting side |
|---------|---------------------------------|--|---|--|------------------------|-----------------|--------------|
| ্রা [0] | Execution/Error completion type | Specify 1 0: Clock startir 1: Clock | r completion the clock da data at the ng from () (data at the () [11]. | 0000н, 0001н, 0080н, 0081н | User | | |
| s1 [1] | Completion status | 0 | : | pletion status is stored. Normal completion Error completion (error o | code) | _ | System |
| | | Specify | the channel | used by the host station. | | | |
| | | | D | escription | Setting value | | |
| st [2] | Channel used by host station | Ethernet MELSECNET/H | | 1 | 1 to 8 | 1 to 10 | User |
| | | CC-Link IE Controller Network 1 to 10 | | | | | |
| | | CC-Link IE Field Network 1 to 2 | | | | - | |
| | | Setti | the type of type of the type of type of the type of t | he target station CPU. Descr Target station CPU/host | system CPU (Specified | 0000н, 03FFн | |
| | | Ethernet | | data are the same as '03 | | | |
| | | | 03FFн ^{*1} 0000н | Target station CPU/host Target station CPU/host data are the same as '03 | system CPU (Specified | | |
| st [3] | Target station's CPU type | ET/H IE | 03E0н ^{*2} | Multi-CPU No. 1/target s system) | tation CPU (single CPU | 0000н, | User |
| | | MELSECNET/H CC-Link IE | 03E1н ^{*2} | Multi-CPU No. 2 | | 03E0н to | |
| | | VELS CC | 03E2н ^{*2} | Multi-CPU No. 3 | | 03E3н, 03FFн | |
| | | ≥ . | 03E3н ^{*2} | Multi-CPU No. 4 | | | |
| | | | 03FFн ^{*1} | Target station CPU/host system CPU | | | |
| s1 [4] | Target station network No. | | Specify the network number of the target station. 1 to 239 : Network number 254 : Specify this when 254 has been set in Jn. | | | 1 to 239, 254 | User |

| Device | Item | Setting data | Setting range | Setting side |
|--------|--------------------|---|-----------------------|--------------|
| | | Specify the station number of the target station. | | |
| | | (1) Station number specification | | |
| | | Setting value Descri | otion | |
| | | MELSECNET/H 1 to 64 | | |
| | | Ethernet 1 to 120 | | |
| | | CC-Link IE Controller Network 1 to 120 | | |
| | | CC-Link IE Field Network - | | |
| | | Master station 125 (7DH |) | |
| | | Local station or the intelligent device 1 to 120 station | | |
| | | To increase the data reliability when the station number is s executing the instruction with setting Execution/Error compl ((s) [0]) to '1: With arrival confirmation' is recommended. (2) Group specification (target station is anything other than | etion type | |
| | | CC-Link IE Field Network) | 1 to 120, | User |
| | | 81н to A0н: All stations in group numbers 1 to 32 | 110 120, 125 (7Dн) | |
| s1[5] | Target station No. | (Setting is available when Execution type is set to '0: W | . , | |
| | | arrival confirmation' in <a>[0] .) | FFH | |
| | | Group No.1 · · · 81н Group No.2 · · · 82н | | |
| | | to | | |
| | | Group No.32 · · · A0 _H | | |
| | | (3) All stations specification | | |
| | | FFH: All stations of the target network number (Except | the host | |
| | | station.) (Setting is available when Execution type is set to '0: W | /ithout | |
| | | | hillout | |
| | | arrival confirmation' in (s) [0].) To specify a group or all stations. | | |
| | | Specify '0000H' or '03FFH' for the target station's CPU typ Group specification cannot be set for the station of the CO Field Network. It cannot be confirmed if the data are written to the target normally. Confirm the device of the target station of the w | C-Link IE station | |
| s1 [6] | - | destination. (Fixed value) | 0 | User |
| 0.01 | | For instruction execution | | |
| | | Specify the number of instruction resends when the instruct | ion is not | |
| | | | <u> </u> | User |
| | | completed within the monitoring time specified in (s) [8]. (Se available when Execution type is set to '1: With arrival confi | | 0.001 |
| st][7] | Number of resends | (i) [0].) | | |
| | | ② At instruction completion | | |
| | | The number of resends (result) is stored. (Setting is availab | le when – | System |
| | | Execution type is set to '1: With arrival confirmation' in 🗐 [0 | 1.) | |

| Device | Item | | Setting data | | Setting range | Setting side |
|--------------------------|--|---|--|--------------------|---------------|--------------|
| | | is available when Exects (s1 [0].) | time required for instruction cution type is set to '1: With ar completed within this time, it | | | |
| | | | | | | |
| ্রা [8] | Arrival monitoring time | Ethernet | O to TCP retransmission timer value: Monitoring is performed by the TCP retransmission timer value. (TCP retransmission timer value + 1) to 16383: Monitoring time (unit: second) | 0 to 16383 | 0 to 32767 | User |
| | | MELSECNET/H CC-Link IE | 0: 10 seconds 1 to 32767: 1 to 32767 seconds | 0 to 32767 | | |
| | | Specify the number of | write data. | | | |
| | | | escription | Setting value | | |
| st [9] | Write data length | Ethernet MELSECNET/H CC-Link IE Field Network | | 1 to 960 (word) | 1 to 8192 | User |
| | | CC-Link IE | Controller Network | | | |
| s1[10] | (Reserved) | | - | - | | |
| ৱা[11] | Clock set flag ^{*3} | Valid/invalid status of t 0: Invalid 1: Valid | _ | System | | |
| ঃ† [12] to sî [15] | Clock data at the time of error completion ^{*3} | Clock data at the time b15 tc (s) [12] Month (01) (s) [13] Hour (00- (s) [14] Second (00) (s) [15] Year (00+ to 99+ | _ | System | | |
| ঙা [16] | Error-detected network No. | | e station where an error was o ror was detected at the host s ber | | _ | System |
| | | Number of the station where an error was detected is stored. (However, when an error was detected at the host station, the network number is not stored.) | | | | |
| | | Setting value Description MELSECNET/H 1 to 64 | | - | | |
| st][17] | Error-detected station No. | Ethernet 1 to 120 CC-Link IE Controller Network - Master station 125 (7DH) | | | - | System |
| | | | | I25 (7Dн) | | |
| | | Local station or the station | he intelligent device | l to 120 | | |

*1: Specification is possible when the host station is a network module or Ethernet module of function version D or later.

(Specification is not possible for other modules. An access is always made to the target station CPU.)

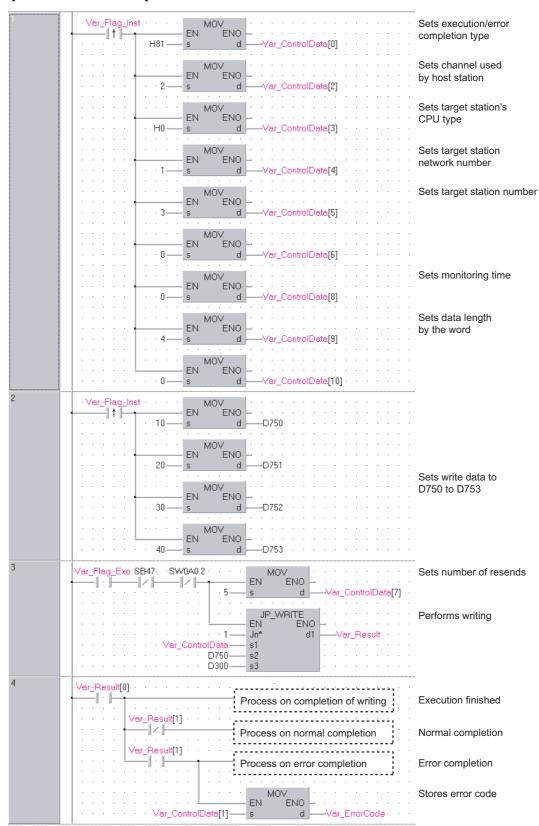
*2 : Specification is possible when the versions of the QCPU and the network module on the host station and the target station are as indicated below.

(Specification is not possible for other modules. An access is always made to the target station CPU.) • Network module: The first five digits of the serial number are '06092' or higher.

• QCPU: The first five digits of the serial number are '06092' or higher.

*3 : Data are stored only when 1 is set in bit 7 of Error completion type ((s) [0]).

The following program writes data which are stored in the devices from D750 to D753 of the station number 2 (host station) to the devices from D300 to D303 of the station number 3 (target station).

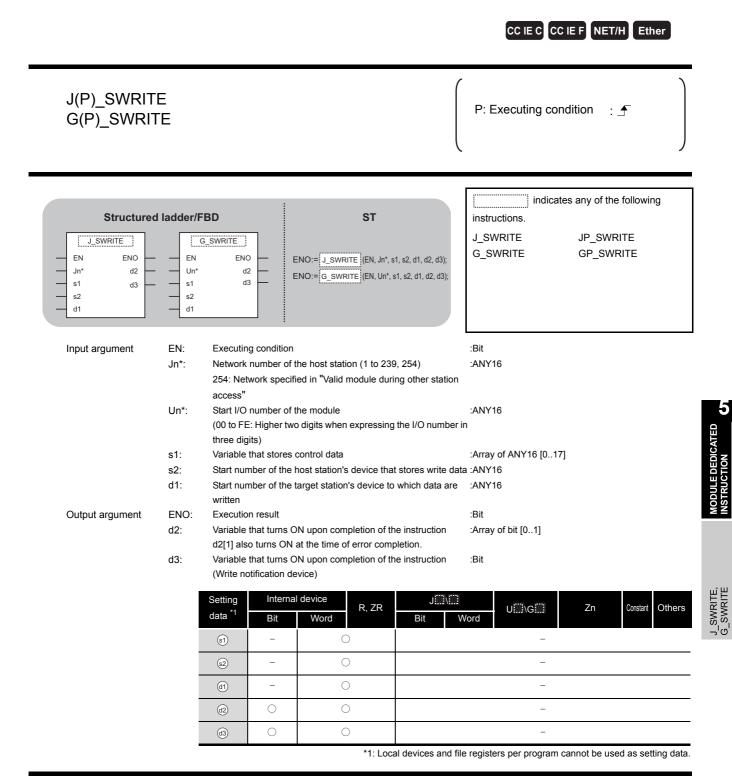


[Structured ladder/FBD]

| [ST] | |
|--|--|
| IF (Var_Flag_Inst=TRUE) THEN | |
| MOV(TRUE,H81,Var_ControlData[0]); | (* Sets execution/error completion type *) |
| MOV(TRUE,2,Var_ControlData[2]); | (* Sets channel used by host station *) |
| MOV(TRUE,H0,Var_ControlData[3]); | (* Sets target station's CPU type *) |
| MOV(TRUE,1,Var_ControlData[4]); | (* Sets target station network number *) |
| MOV(TRUE,3,Var_ControlData[5]); | (* Sets target station number *) |
| MOV(TRUE,0,Var_ControlData[6]); | |
| MOV(TRUE,0,Var_ControlData[8]); | (* Sets monitoring time *) |
| MOV(TRUE,4,Var_ControlData[9]); | (* Sets data length by the word *) |
| MOV(TRUE,0,Var_ControlData[10]); | |
| END_IF; | |
| IF (LDP(TRUE,Var_Flag_Inst2)) THEN | (* Cata write data to DZE0 to DZE2 *) |
| MOV(TRUE,10,D750); | (* Sets write data to D750 to D753 *) |
| MOV(TRUE,20,D751); MOV(TRUE,30,D752); | |
| MOV(TRUE, 30, D752); | |
| END IF; | |
| IF((Var_Flag_Exe=TRUE) AND (SB47=FALSE) | AND (SW0A0 2=FALSE)) THEN |
| MOV(TRUE, 5, Var_ControlData[7]); | |
| JP WRITE(TRUE,1,Var ControlData,D750 | |
| | (* Performs writing *) |
| END IF; | 、 3 , |
| IF(Var_Result[0]=TRUE)THEN | (* Execution finished *) |
| | |
| (* Process on completion of writing *) | |
| IF(Var_Result[1]=FALSE)THEN | (* Normal completion *) |
| IF(Var_Result[1]=FALSE)THEN (* Process on normal completion *) | |
| ELSE | (* Error completion *) |
| ELSE (* Process on error completion *) | |
| MOV(TRUE, Var_ControlData[1], Va | |
| | (* Stores error code *) |
| END IF; | |
| END_IF; | |
| , | |

5.4.11 SWRITE instruction

J_SWRITE, G_SWRITE



Grant Function

This instruction writes data to a word device of another station.

Control Data

For the control data of the SWRITE instruction that writes data to the word device memory of another station, refer to WRITE instruction.

The control data of the SWRITE instruction are the same as those of the WRITE instruction. Accordingly, this section omits the explanation.

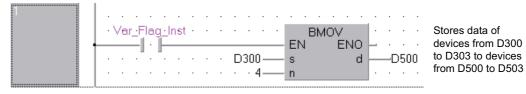
The following program example of the SWRITE instruction is different from that of the WRITE instruction by assigning the write notification device (3) at the end of arguments.

[Structured ladder/FBD]

(1) Program on the request source (station number 2) of the SWRITE instruction

| 1 | Var_Flag_Inst · · · · MOV EN EI | 10 | Sets execution/error |
|---|---|---|---------------------------------------|
| | · · · · · · · · · · · · · · · · · · · | dVar_ControlData[0] | completion type |
| | MOV | VO - · · · · · · · · · · · · · · · · · · | Sets channel used by host station |
| | · · · · · · · · · · · · · · · · · · · | dVar_ControlData[2] | |
| | | vo <mark>-</mark> | Sets target station's CPU type |
| | · · · · · · H0 — s · · · · · · · · · · · · · · · | d Var_ControlData[3] | |
| | | d —Var ControlData[4] | Sets target station network number |
| | | | Sets target station number |
| | | NO | |
| | | · · · · · · · · · · · · · · · · · · · | |
| | | NOVar_ControlData[6] · · · · · · · | |
| | | | Sets monitoring time |
| | · · · · · · · · 0 — s | dVar_ControlData[8] | |
| | MOV | 10 | Sets data length by the word |
| | · · · · · · · · · · · · · · · · · · · | d Var_ControlData[9] | by the word |
| | · · · · · · · · · · · · · · Et | VO - · · · · · · · · · · · · · · · · · · | |
| 2 | · · · · · · · · 0 — s | _dVar_ControlData[10] · · · · · · · | |
| 2 | | 10 | |
| | · · · · · · · 10—s | dD750 · · · · · · · · · · · · · · · · · · · | |
| | | NOD751 | |
| | | D/31 | Sets write data to D750 to D753 |
| | | чо – | |
| | | · · · · · · · · · · · · · · · · · · · | |
| | | NO | |
| 3 | Var_Flag_Exe SB47· · SW0A0.2 · · · | | Sets number of resends |
| | · · · · · · · · · · · · · · · · · · · | S d Var_ControlData[7] | |
| | | · JP_SWRITE · · · · · · · · · · · · · · · · · · · | Performs writing |
| | Var_ControlData | Jn* d2Var_Result · · · | |
| | D750- | — s2 · · · · · · · · | |
| 4 | | | |
| | Var_Result[0] · · · · · · · · · · · · · · · · · · · | Process on completion of writing | Execution finished |
| | Var_Result[1] | Process on normal completion | Name a completion |
| | | Process on normal completion | Normal completion |
| | | Process on error completion | Error completion |
| | | | Stores error code |
| | · · · · · · · · · Var_ControlData[1]· | EN ENO Var_ErrorCode | |
| | | | |

(2) Program on the request target (station number 3) of the SWRITE instruction



[ST]

(1) Program on the request source (station number 2) of the SWRITE instruction

| | IF (Var_Flag_Inst=TRUE) THEN | , |
|---|--|--|
| | MOV(TRUE,H81,Var_ControlData[0]); | (* Sets execution/error completion type *) |
| | MOV(TRUE,2,Var_ControlData[2]); | (* Sets channel used by host station *) |
| | MOV(TRUE,H0,Var_ControlData[3]); | (* Sets target station's CPU type *) |
| | MOV(TRUE,1,Var_ControlData[4]); | (* Sets target station network number *) |
| | MOV(TRUE,3,Var_ControlData[5]); | (* Sets target station number *) |
| | MOV(TRUE,0,Var_ControlData[6]); | (* Oata manitarian time *) |
| | MOV(TRUE,0,Var_ControlData[8]); MOV(TRUE,4,Var_ControlData[9]); | (* Sets monitoring time *) (* Sets data length by the word *) |
| | MOV(TRUE,4, val_controlData[9]); | |
| | END_IF; | |
| | IF (Var_Flag_Inst2=TRUE) THEN | |
| | MOV(TRUE,10,D750); | (* Sets write data to D750 to D753 *) |
| | MOV(TRUE,20,D751); | |
| | MOV(TRUE,30,D752); | |
| | MOV(TRUE,40,D753); | |
| | END_IF; IF((Var_Flag_Exe=TRUE) AND (SB47=FALSE). | ΔΝΙΟ (SWOAO 2=ΕΔΙ SE)) ΤΗΕΝ |
| | MOV(TRUE, 5, Var_ControlData[7]); | |
| | JP_SWRITE(TRUE,1,Var_ControlData,D75 | |
| | _ 、 | (* Performs writing *) |
| | END_IF; | |
| | IF(Var Result[0]=TRUE)THEN | (* Execution finished *) |
| | IF(Var_Result[0]=TRUE)THEN | |
| | (* Process on completion of writing *) | |
| | IF(Var_Result[1]=FALSE)THEN | (* Normal completion *) |
| | (* Process on normal completion *) | |
| | ELSE | (* Error completion *) |
| | (* Droccoc on error completion *) | |
| | (* Process on error completion *) | |
| | MOV(TRUE, Var_ControlData[1], Va | r_ErrorCode);(* Stores error code *) |
| | END_IF; | |
| | END_IF; | |
| | | |
|) | Program on the request target (station number 3 |) of the SWRITE instruction |
| | IF(Var_Flag=TRUE) THEN | |
| | BMOV(TRUE D300 4 D500) | |

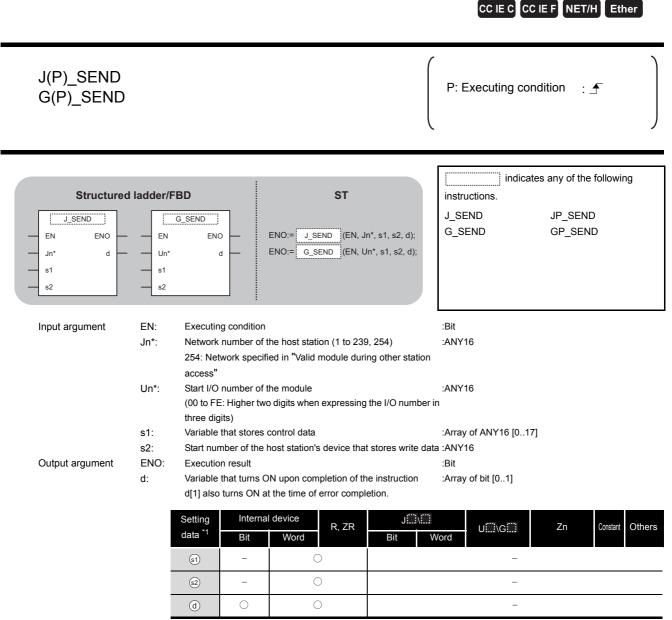
BMOV(TRUE,D300,4,D500);

(* Stores data of devices from D300 to D303 to devices from D500 to D503 *) END_IF;

(2)

5.4.12 SEND instruction

J_SEND, G_SEND



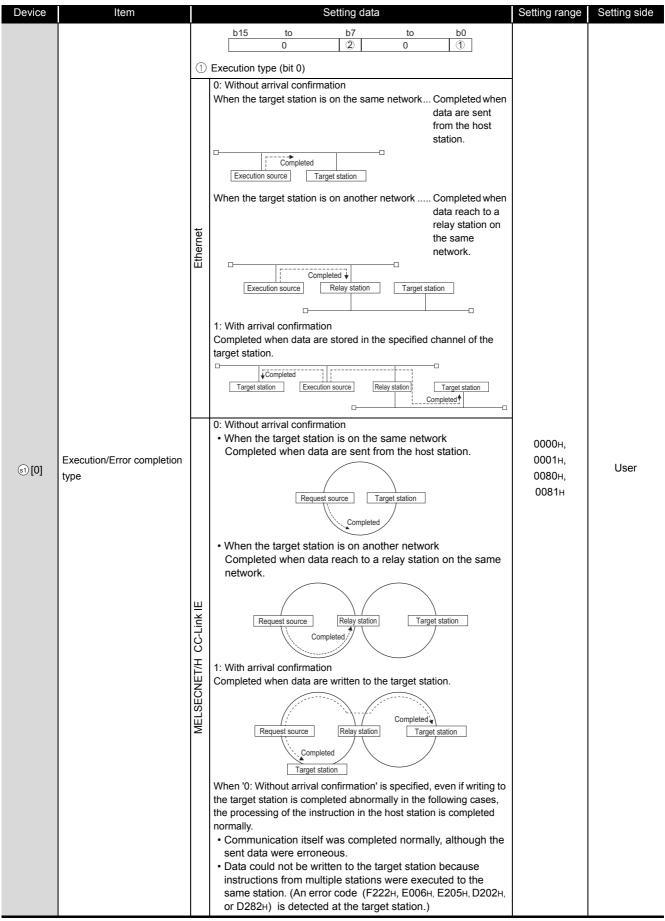
*1: Local devices and file registers per program cannot be used as setting data.

Granitical Function

This instruction sends data to another station.

J_SEND, G_SEND

Control Data



| Device | Item | Setting data | | Setting range | Setting side |
|---------|---------------------------------|---|--|-------------------------------------|--------------|
| st) [0] | Execution/Error completion type | ② Error completion type (bit 7) Specify the clock data setup status at the time of error completion. O: Clock data at the time of error completion is not set in the area starting from (a) [11]. 1: Clock data at the time of error completion is set in the area starting from (a) [11]. | | 0000н, 0001н, 0080н, 0081н | User |
| ৱা[1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error completion) | de) | - | System |
| ৱ্য [2] | Channel used by host station | CC-Link IE Controller Network | Setting value | 1 to 8 | User |
| (3] ال | Target station channel | Ethernet | res data. ^{*2} Setting value 1 to 64 1 to 8 | 1 to 64 | User |
| ട)[4] | Target station network No. | Specify the network number of the target station 1 to 239 : Network number 254 : Specify this when 254 has (Network specified in 'Valio station access') | been set in Jn. | 1 to 239, 254 | User |

| Device | Item | Setting data Specify the station number of the target station. (1) Station number specification | Setting range | Setting side | |
|--------|---|--|--------------------------|--------------|--|
| | | Setting value Descri | otion | | |
| | | MELSECNET/H 1 to 64 | | | |
| | | Ethernet | | | |
| | | CC-Link IE Controller Network | | | |
| | | Host station is Universal model QCPU 1 to120 | | | |
| | | Host station is anything other than Universal model QCPU | | | |
| | | CC-Link IE Field Network - | | | |
| | | Master station 125 (7DF | 1) | | |
| | | Local station or the intelligent device 1 to 120 station | | | |
| | | To increase the data reliability when the station number is s executing the instruction with setting Execution/Error complete | | | |
| st [5] | [5] Target station No. | ((s) [0]) to '1: With arrival confirmation' is recommended. (2) Group specification (target station is anything other than CC-Link IE Field Network) 81H to A0H: All stations in group numbers 1 to 32 (Setting is available when Execution type is set to '0: W | 125 (7Dн) 81н to А0н, | User | |
| | | arrival confirmation' in (s) [0].) | | | |
| | | Group No.1 · · · 81н Group No.2 · · · 82н | | | |
| | | to | | | |
| | | Group No.32 · · · А0н | | | |
| | | (3) All stations specification FFH: All stations of the target network number (Except the host station.) (Setting is available when Execution type is set to '0: Without | | | |
| | | arrival confirmation' in (1) [0].) | | | |
| | | To specify a group or all stations. | | | |
| | Specify '0000 Group specifi Field Networ It cannot be | Specify '0000H' or '03FFH' for the target station's CPU type. Group specification cannot be set for the station of the CR Field Network. It cannot be confirmed if the data are written to the target normally. Confirm the device of the target station of the written to the target station. | C-Link IE station | | |
| | | destination. | | | |
| s1[6] | - | (Fixed value) | 0 | User | |
| | | For instruction execution Specify the number of instruction resends when the instruct | ion is not | | |
| | | completed within the monitoring time specified in $\textcircled{3}$ [8]. (See | etting is 0 to 15 | User | |
| st [7] | Number of resends | available when Execution type is set to '1: With arrival confi (1) (0].) | rmation' in | | |
| | | ② At instruction completion | | | |
| | | The number of resends (result) is stored. (Setting is available | e when the – | System | |
| | | Execution type is set to '1: With arrival confirmation' in (3) [0 | | - | |

| Device | Item | | Setting data | | Setting range | Setting side |
|--------------------------|--|---|--|---------------|---------------|--------------|
| | | Specify the monitoring is available when Exect in (a) [0].) If the instruction is not a number of times specifi | | | | |
| | | De | escription | Setting value | | |
| ട)[8] | Arrival monitoring time | Ethernet | 0 to TCP retransmission timer value: Monitoring is performed by the TCP retransmission timer value. (TCP retransmission timer value + 1) to 16383: Monitoring time (unit: second) | 0 to 16383 | 0 to 32767 | User |
| | | MELSECNET/H CC-Link IE | 0: 10 seconds 1 to 32767: 1 to 32767 seconds | 0 to 32767 | | |
| s1 [9] | Send data length | Specify the number of s | 1 to 960 | User | | |
| s1 [10] | (Reserved) | | - | | - | - |
| s1][11] | Clock set flag ^{*1} | Valid/invalid status of the data in the area starting from (a) [12] is stored. 0: Invalid 1: Valid | | | - | System |
| ৱা [12] to ৱা [15] | Clock data at the time of error completion ^{*1} | b15 to (a) [12] <u>Молth (01н</u> (a) [13] <u>Ноиг (00н</u> (a) [14] <u>Second (00</u> | (12] Month (01н to 12н) Year (00н to 99н) Last two digits | | | |
| ঃা [16] | Error-detected network | Network number of the (However, when an err number is not stored.) 1 to 239: Network num | _ | System | | |
| ্রা [17] | Error-detected station No.*1 | Setting value Description MELSECNET/H 1 to 64 Ethernet 1 to 120 CC-Link IE Controller Network - Master station 125 (7DH) Local station or the intelligent device 1 to 120 | | _ | System | |

*1 : Data are stored only when 1 is set in bit 7 of Error completion type ((() [0]).

*2: Logical channel setting is not available for the CC-Link IE network module.

5.4 Network Dedicated Instruction

5.4.12 SEND instruction

MODULE DEDICATED INSTRUCTION

J_SEND, G_SEND

Program Example

The following program sends data of the devices from D750 to D753 of the station number 1 (host station) to the channel 5 of the station number 2 (target station).

For the method for reading the data, which are sent by the SEND instruction, from the channel 5 of the station number 2 (target station), refer to the following sections.

For reading out data in a main program

Section 5.4.13 RECV instruction

For reading out data in an interrupt program

Section 5.4.14 RECVS instruction

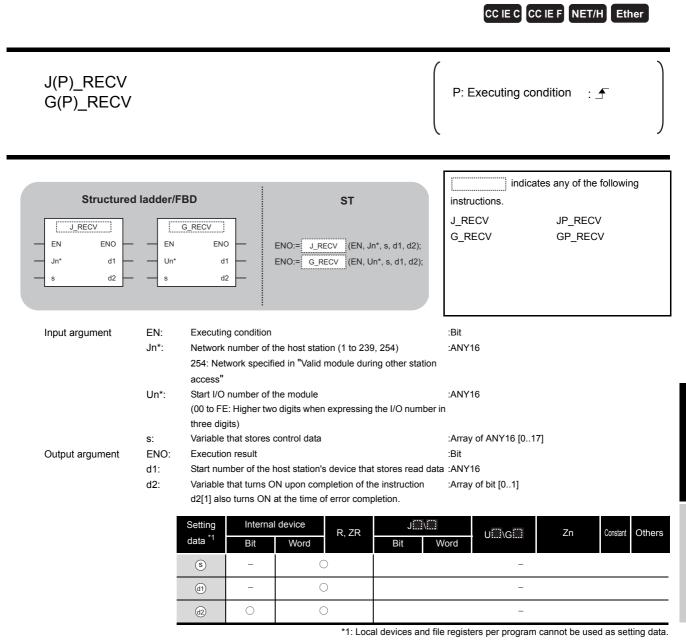
[Structured ladder/FBD]

| 1 | ·Var_Flag_Inst MOV Inst EN EN ENO ···································· | Sets execution/error completion type |
|---|---|--------------------------------------|
| | EN ENO s d Var_ControlData[2] | Sets channel used by host station |
| | Image: Movement of the state of the stat | Sets target station channel |
| | MOV EN EN S d Var_ControlData[4] | Sets target station network number |
| | EN ENO 2 s d Var_ControlData[5] | Sets target station number |
| | EN ENO 0 s d Var_ControlData[6] | |
| | EN ENO 0 s d Var_ControlData[8] | Sets monitoring time |
| | EN ENO 4 s d Var_ControlData[9] | Sets data length by the word |
| | MOV MOV EN ENO S d Var_ControlData[10] | |
| 2 | ·Var_Flag:Inst MOV ↑ EN ENO 0 10 s d | |
| | EN ENO 20 | Sets send data to D750 to D753 |
| | | |
| | | |
| 3 | Var_Flag_Exe SB47 SW0A0.1 MOV Image: Strategy of the strategy of | Sets number of resends |
| | JP_SEND EN | Sends data |
| 4 | Var_Result[0] Process on completion of sending | Execution finished |
| | Var_Result[1] Process on normal completion | Normal completion |
| | Process on error completion | Error completion |
| | MOV EN EN ENO s d | Stores error code |

[ST] IF (Var_Flag_Inst=TRUE) THEN MOV(TRUE,H81,Var_ControlData[0]); (* Sets execution/error completion type *) MOV(TRUE,3,Var_ControlData[2]); (* Sets channel used by host station *) MOV(TRUE,H5,Var_ControlData[3]); (* Sets target station channel *) MOV(TRUE,1,Var ControlData[4]); (* Sets target station network number *) (* Sets target station number *) MOV(TRUE,2,Var_ControlData[5]); MOV(TRUE,0,Var_ControlData[6]); MOV(TRUE,0,Var_ControlData[8]); (* Sets monitoring time *) MOV(TRUE,4,Var_ControlData[9]); (* Sets data length by the word *) MOV(TRUE,0,Var ControlData[10]); END IF; IF (Var_Flag_Inst2=TRUE) THEN MOV(TRUE, 10, D750); (*Sets send data to D750 to D753 *) MOV(TRUE,20,D751); MOV(TRUE, 30, D752); MOV(TRUE,40,D753); END IF; IF((Var_Flag_Exe=TRUE) AND (SB47=FALSE) AND (SW0A0.1=FALSE)) THEN MOV(TRUE, 5, Var_ControlData[7]); (* Sets number of resends *) JP_SEND(TRUE,1,Var_ControlData,D750,Var_Result);(* Sends data *) END IF: IF(Var Result[0]=TRUE)THEN (* Execution finished *) -----(* Process on completion of sending *) IF(Var_Result[1]=FALSE)THEN (* Normal completion *) _____ (* Process on normal completion *) ELSE (* Error completion *) (* Process on error completion *) ···· / MOV(TRUE, Var_ControlData[1], Var_ErrorCode);(* Stores error code *) END IF; END IF;

5.4.13 RECV instruction

J_RECV, G_RECV



Grant Function

This instruction reads received data (for main program).

Control Data

| Device | Item | Setting data | | Setting range | Setting side | |
|--------------|---------------------------------|--|--|---------------------------------|--------------|--------|
| ®[0] | Error completion type | b15 to b7 to b0 0 0 0 0 0 1 Error completion type (bit 7) Specify the clock data setup status at the time of error completion. 0 0 0: Clock data at the time of error completion is not set in the area starting from (§ [11]). 1: Clock data at the time of error completion is set in the area starting from (§ [11]). The instruction completion status is stored. The instruction completion status is stored. | | 0000н, 0080н ng | User | |
| ® [1] | Completion status | 0 : No Other than 0 : Err | _ | System | | |
| ঙ [2] | Host station channel | Description Setting value MELSECNET/H 1 to 8 CC-Link IE Controller Network CC-Link IE Field Network | | 1 to 8 | User | |
| s [3] | Channel used by sending station | Channel used by the se 1 to 8: Channel | ending station is stored. | | _ | System |
| \$[4] | Network No. of sending station | Network number of the 1 to 239: Network n | | System | | |
| ٤[5] | Sending station No. | Station number of the sending station is stored. Setting value Description MELSECNET/H 1 to 64 Ethernet 1 to 120 CC-Link IE Controller Network - Master station 125 (7DH) Local station or the intelligent device 1 to 120 | | | System | |
| § [6] | (Reserved) | | _ | | _ | - |
| §[7] | (Reserved) | | _ | | | _ |
| | | Specify the monitoring When the instruction is completes abnormally. | not completed within th | e monitoring time, it | | |
| (8) | Arrival monitoring time | Ethernet CC-Link IE MELSECNET/H | 0 to TCP retransmissi timer value: Monitoring performed by the TCP retransmission timer v (TCP retransmission t value + 1) to 16383: Monitoring time (unit: second) 0: 10 seconds 1 to 32767: 1 to 32767 seconds | g is alue. mer 0 to 16383 | 0 to 32767 | User |

| Device | Item | Setting data | Setting range | Setting side |
|----------------------|--|--|---------------|--------------|
| ි [9] | Receive data length | The number of received data stored in (1) to (1) + n is stored.0: No receive data1 to 960: Number of words of receive data | - | System |
| s [10] | (Reserved) | _ | - | - |
| ঙ [11] | Clock set flag ^{*1} | Valid/invalid status of the data in the area starting from (\$)[12] is stored. 0: Invalid 1: Valid | - | System |
| ©[12] to ©[15] | Clock data at the time of error completion ^{*1} | Clock data at the time of error completion are stored in BCD format. b15 to b8 b7 to b0 (s) [12] Month (01H to 12H) Year (00H to 99H) Last two digits b0 (s) [13] Hour (00H to 23H) Day (01H to 31H) b14 b15 b14 Second (00H to 59H) Minute (00H to 59H) b16 b17 b17 | - | System |
| § [16] | Error-detected network No. *1 | Network number of the station where an error was detected is stored. (However, when an error was detected at the host station, the network number is not stored.) 1 to 239: Network number | _ | System |
| s [17] | Error-detected station No.*1 | Number of the station where an error was detected is stored. (However, when an error was detected at the host station, the network number is not stored.) Setting value Description MELSECNET/H 1 to 64 Ethernet 1 to 120 CC-Link IE Controller Network - Master station 125 (7DH) Local station or the intelligent device 1 to 120 | _ | System |

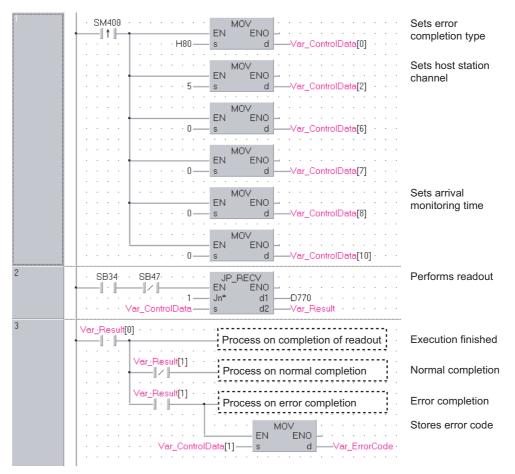
*1 : Data are stored only when 1 is set in bit 7 of Error completion type (([§][0]).

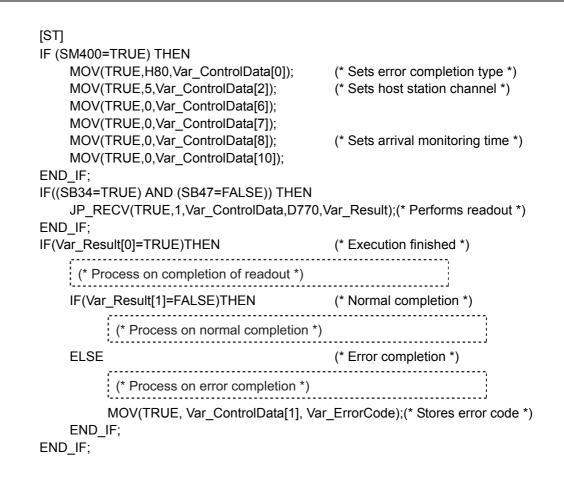
Program Example

The following program reads out data, which is sent from the station number 1 by the SEND instruction, from the channel 5 of the station number 2 (host station) and stores the data to the devices from D770 to D773 of the station number 2 (host station) when SB0034 turns ON. For the SEND instruction, refer to the following section.

Section 5.4.12 SEND instruction

[Structured ladder/FBD]





5.4.14 RECVS instruction

Z_RECVS

CC IE C CC IE F NET/H Ether

Z_RECVS

| Structured EN Jn* s1 s2 | Iadder/F ECVS ENO d | | NO:= Z_RE | ST CVS (EN, U | Jn*, s1, s2, d) | | Z_RI | ECVS | es the follo | wing instr | ruction. |
|-------------------------------------|------------------------------|------------------------|----------------|------------------|-----------------|---------------|----------|---------------|--------------|------------|----------|
| Input argument | EN: | Executin | g condition | | | | :Bit | | | | |
| | Un*: | | number of t | | | | :String | 9 | | | |
| | | (00 to FE three dig | - | o digits wher | n expressing | the I/O numbe | er in | | | | |
| | s1: | Variable | that stores of | control data | | | :Array | of ANY16 [017 | 7] | | |
| | s2: | Start nur | nber of the h | nost station's | s device that | stores read d | ata :ANY | 16 | | | |
| Output argument | ENO: | Executio | n result | | | | :Bit | | | | |
| | d: | Dummy | | | | | :Bit | | | | |
| | | Setting | Internal | device | R, ZR | J\[| | U∭\G∭ | Zn | Constant | Others |
| | | data ^{*1} | Bit | Word | , | Bit | Word | 0 | | | |
| | | s1 | - | C |) | | | - | | | |
| | Ī | s2 | _ | C |) | | | _ | | | |
| | | d | 0 | C |) | | | _ | | | |

*1: Local devices and file registers per program cannot be used as setting data.

This instruction reads received data (for interrupt program).

Control Data

| Device | Item | Setting data | | Setting range | Setting side |
|--------------------------|---------------------------------|--|--|---------------|--------------|
| §[0] | Completion type | b15 to 0 (Fixed) | b0 | 0 | User |
| ঙ[1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error completion) | _ | System | |
| | | Specify the channel of host station that stores Description | receive data. Setting value | | |
| ঙ[2] | Host station channel | MELSECNET/H Ethernet CC-Link IE Controller Network | 1 to 8 | 1 to 8 | User |
| | | CC-Link IE Field Network | 1 to 2 | | |
| ⑤ [3] | Channel used by sending station | Channel used by the sending station is stored. 1 to 8: Channel | - | System | |
| ⑤ [4] | Network No. of sending station | Network number of the sending station is store 1 to 239: Network number | ed. | _ | System |
| ঙ [5] | Sending station No. | Station number of the sending station is stored Setting value MELSECNET/H Ethernet CC-Link IE Controller Network | Description 1 to 64 1 to 120 | _ | System |
| | | CC-Link IE Field Network Master station slave station | - 125 (7DH) 1 to 120 | | |
| \$[6] \$[7] \$[8] | System area | _ | - | _ | |
| ি [9] | Receive data length | The number of received data stored in (1) to (1) + n is stored. 0 : No receive data 1 to 960 : Number of words of receive data | | - | System |
| \$ [10] to \$ [17] | System area | _ | | - | - |

Z_RECVS

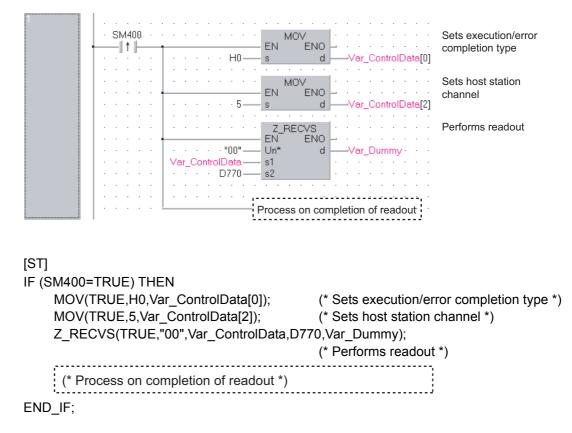
Program Example

The following program reads data, which is sent from the station number 1 by the SEND instruction, from the channel 5 of the station number 2 (host station) and stores the data to the devices from D770 to D773 of the station number 2 (host station) when an interruption program starts up.

For the SEND instruction, refer to the following section.

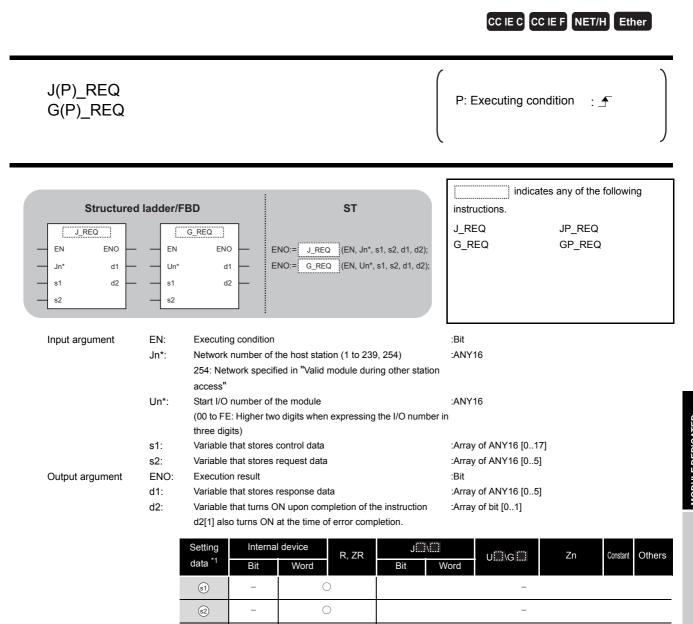
Section 5.4.12 SEND instruction

[Structured ladder/FBD]



5.4.15 REQ instruction

J_REQ, G_REQ



*1: Local devices and file registers per program cannot be used as setting data.

_



Remotely runs or stops a programmable controller on another station.

(d1)

d2)

_

Also, reads/writes clock data from/to a programmable controller on another station.

Control Data

| Device | Item | | | Setting data | Setting range | Setting side |
|---------|------------------------------|---|--|---|--|--------------|
| ട്ര [0] | Error completion type | Specify t 0: Clock from (1: Clock | b15 t completion typ he clock data s data at the time (a) [11]. data at the time (b) [11]. | 0011н, 0091н | User | |
| ৱা[1] | Completion status | 0 | ruction complet : No : than 0 : Err | - | System | |
| st [2] | Channel used by host station | Specify the channel used by the host station. 1 to 8: Channel | | | 1 to 8 | User |
| s) [3] | Target station's CPU type | | he type of the t ting value 0000н 03FFн ^{*1} 0000н 03E0н ^{*2} 03E1н ^{*2} 03E2н ^{*2} 03E3н ^{*2} 03E3н ^{*2} | Description Target station CPU/host system CPU (Specified data are the same as '03FFH'.) Target station CPU/host system CPU Target station CPU/host system CPU (Specified data are the same as '03FFH'.) Multi-CPU No. 1/target station CPU (single CPU system) Multi-CPU No. 2 Multi-CPU No. 3 Multi-CPU No. 4 | 0000н, 03FFн 0000н, 03E0н to 03E3н, 03FFн | User |
| গ্ৰ [4] | Target station network No. | Specify the network number of the target station. 1 to 239 : Network number 254 : Specify this when 254 has been set in Jn. (Network specified in 'Valid module during other station access') | | | 1 to 239, 254 | User |

5-182

| | Item | | Setting data | | Setting range | Setting side |
|--------------------|--------------------|---|--|---|--------------------|----------------|
| | | Specify the station number of | - | | | |
| | | (1) Station number speci | fication | | | |
| | | Setting va | lue D | escription | | |
| | | MELSECNET/H | 1 tr | o 64 | | |
| | | Ethernet | Ethernet | | | |
| | | CC-Link IE Controller Netwo | ork - | | | |
| | | Host station is Universa | al model QCPU 1 to | o120 | | |
| | | Host station is anything | other than | - 64 | | |
| | | Universal model QCPU | | o 64 | | |
| | | CC-Link IE Field Network | - | | | |
| | | Master station | 12 | 5 (7DH) | | |
| | | Local station or the inte | lligent device | o 120 | | |
| | | station | 1.0 | 0 120 | | |
| | | (2) Group specification (targ | et station is anything oth | er than | 1 to 120, | |
| | | CC-Link IE Field Network | | | 125 (7Dн) | |
| st) [5] | Target station No. | 81н to A0н: All stations in gro data writing and remote RUN | • | ailable only at clock | 81н to А0н, FFн | User |
| | | Grou | up No.1 · · · 81н | | | |
| | | | up No.2 · · · 82н | | | |
| | | | to | | | |
| | | Grou | | | | |
| | | (3) All stations specification | | | | |
| | | FFH: All stations of the ta | | | | |
| | | station.) (Available only | | | | |
| | | STOP) To specify a group or all stati | | | | |
| | | | | | | |
| | | Specify '0000H' or '03FFH' Group specification cannot | | | | |
| | | Field Network. | | | | |
| | | It cannot be confirmed if the | | - | | |
| | | normally. Confirm the devi destination. | ce of the target station of | t the write | | |
| s1[6] | _ | (Fixed value) | | | | |
| (v) | | (| | | 0 | User |
| | | | | | 0 | User |
| | | ① For instruction execution | | | | |
| | | Specify the number of resent | | s not completed | 0 0 to 15 | User User |
| st][7] | Number of resends | 0 | | s not completed | | |
| s ¹ [7] | Number of resends | Specify the number of resent | ecified in 🗊 [8]. | s not completed | 0 to 15 | User |
| ৱা[7] | Number of resends | Specify the number of resent within the monitoring time sp | ecified in 🗊 [8]. | s not completed | | |
| ৱা[7] | Number of resends | Specify the number of resent within the monitoring time sp ② At instruction completion | ecified in (a) [8]. Ilt) is stored | | 0 to 15 | User |
| ৱ্য[7] | Number of resends | Specify the number of resent within the monitoring time sp ② At instruction completion The number of resends (resu | ecified in (a) [8]. Ilt) is stored equired for the instruction | n completion. | 0 to 15 | User |
| ্রা[7] | Number of resends | Specify the number of resent within the monitoring time sp ② At instruction completion The number of resends (resu Specify the monitoring time r If the instruction is not compl | ecified in (a) [8]. It) is stored equired for the instruction eted within this time, it is | n completion. | 0 to 15 | User |
| s) [7] | Number of resends | Specify the number of resent within the monitoring time sp ② At instruction completion The number of resends (resu Specify the monitoring time r | ecified in (a) [8]. It) is stored equired for the instruction eted within this time, it is | n completion. | 0 to 15 | User |
| ্য [7] | Number of resends | Specify the number of resent within the monitoring time sp ② At instruction completion The number of resends (resu Specify the monitoring time r If the instruction is not compl | ecified in (a) [8]. It) is stored equired for the instruction eted within this time, it is (a) [7]. | n completion. | 0 to 15 | User |
| ৱ্য [7] | Number of resends | Specify the number of resent within the monitoring time sp ② At instruction completion The number of resends (resu Specify the monitoring time r If the instruction is not compl number of times specified in Descrip | ecified in (a) [8]. It) is stored equired for the instruction eted within this time, it is (a) [7]. | n completion. | 0 to 15 | User |
| ্রা [7] | Number of resends | Specify the number of resent within the monitoring time sp ② At instruction completion The number of resends (resu Specify the monitoring time r If the instruction is not compl number of times specified in Descrip 0 to | ecified in (a) [8]. IIt) is stored equired for the instruction eted within this time, it is (a) [7]. Stion | n completion. | 0 to 15 | User |
| | | Specify the number of resent within the monitoring time sp ② At instruction completion The number of resends (resu Specify the monitoring time r If the instruction is not compl number of times specified in Descrip 0 to time | ecified in (a) [8]. It) is stored equired for the instruction eted within this time, it is (a) [7]. Definition TCP retransmission | n completion. | 0 to 15 0 to 15 | User System |
| ্জ [7] | Number of resends | Specify the number of resent within the monitoring time sp ② At instruction completion The number of resends (resu Specify the monitoring time r If the instruction is not compl number of times specified in Descrip 0 to time perfer | ecified in (a) [8]. It) is stored equired for the instruction eted within this time, it is (a) [7]. Defining TCP retransmission r value: Monitoring is | n completion. resent by the Setting value | 0 to 15 | User |
| | | Specify the number of resent within the monitoring time sp ② At instruction completion The number of resends (resu Specify the monitoring time r If the instruction is not compl number of times specified in Descrip 0 to time perfor retra | ecified in (a) [8]. It) is stored equired for the instruction eted within this time, it is (a) [7]. Dition TCP retransmission r value: Monitoring is pormed by the TCP | n completion. | 0 to 15 0 to 15 | User System |
| | | Specify the number of resent within the monitoring time sp ② At instruction completion The number of resends (resu Specify the monitoring time r If the instruction is not compl number of times specified in Descrip 0 to time perfor retra (TCI | ecified in (a) [8]. It) is stored equired for the instruction eted within this time, it is (a) [7]. otion TCP retransmission r value: Monitoring is pormed by the TCP ansmission timer value. | n completion. resent by the Setting value | 0 to 15 0 to 15 | User System |
| | | Specify the number of resent within the monitoring time sp ② At instruction completion The number of resends (resu Specify the monitoring time r If the instruction is not compl number of times specified in Descrip 0 to time perfit retra (TCI valu Mon | ecified in (a) [8]. It) is stored equired for the instruction eted within this time, it is (a) [7]. Definition TCP retransmission r value: Monitoring is formed by the TCP ansmission timer value. P retransmission timer e + 1) to 16383: itoring time (unit: | n completion. resent by the Setting value | 0 to 15 0 to 15 | User System |
| | | Specify the number of resent within the monitoring time sp ② At instruction completion The number of resends (resu Specify the monitoring time r If the instruction is not compl number of times specified in Descrip 0 to time perfer retra (TCI valu | ecified in (a) [8]. It) is stored equired for the instruction eted within this time, it is (a) [7]. Definition TCP retransmission r value: Monitoring is formed by the TCP ansmission timer value. P retransmission timer e + 1) to 16383: itoring time (unit: | n completion. resent by the Setting value | 0 to 15 0 to 15 | User System |
| | | Specify the number of resent within the monitoring time sp ② At instruction completion The number of resends (resu Specify the monitoring time r If the instruction is not compl number of times specified in Descrip 0 to time perfor retra (TCI valu Mon seco MELSECNET/H | ecified in (a) [8]. It) is stored equired for the instruction eted within this time, it is (a) [7]. Pretransmission r value: Monitoring is pred by the TCP ansmission timer value. P retransmission timer e + 1) to 16383: itoring time (unit: ond) D seconds | n completion. resent by the Setting value 0 to 16383 | 0 to 15 0 to 15 | User System |
| | | Specify the number of resent within the monitoring time sp ② At instruction completion The number of resends (resu Specify the monitoring time r If the instruction is not compl number of times specified in Descrip 0 to time perfor retra (TCI valu Mon seco MELSECNET/H | ecified in (a) [8]. (a) is stored equired for the instruction eted within this time, it is (a) [7]. (b) TCP retransmission r value: Monitoring is pormed by the TCP ansmission timer value. P retransmission timer e + 1) to 16383: itoring time (unit: pnd) | n completion. resent by the Setting value | 0 to 15 0 to 15 | User System |

| Device | Item | Setting data | | Setting range | Setting side |
|------------------------|--|---|-----------|---------------|--------------|
| st) [9] | Request data length | Specify the number of request data (words). (Number of words of the data stored in request dat 4: Remote RUN 3: Remote STOP 2: Clock data read 6: Clock data write | 2 to 4, 6 | User | |
| st) [10] | Response data length | Number of response data (words) are stored. (Number of words of the data stored in response d 2: Remote RUN/STOP 6: Clock data read 2: Clock data write | _ | System | |
| s1 [11] | Clock set flag ^{*3} | Valid/invalid status of the data in the area starting f 0: Invalid 1: Valid | - | System | |
| জ [12] to জ [15] | Clock data on error completion ^{*3} | Clock data at the time of error completion are stored in BCD format. b15 to b8 b7 to b0 (a) [12] Month (01H to 12H) Year (00H to 99H) Last two digits b1 b1 b1 b0 (a) [13] Hour (00H to 23H) Day (01H to 31H) b1 b1 | | _ | System |
| গ্রা [16] | Error-detected network No. *3 | Network number of the station where an error was detected is stored. (However, when an error was detected at the host station, the network number is not stored.) 1 to 239: Network number | | - | System |
| ্ত [17] | Error-detected station No. | Ethernet CC-Link IE Controller Network CC-Link IE Field Network Master station | | _ | System |

*1: Specification is possible when the host station is a network module or Ethernet module of function version D or later.

(Specification is not possible for other modules. An access is always made to the target station CPU.)
 *2 : Specification is possible when the versions of the QCPU and the network module on the host station and the target station are as indicated below.

(Specification is not possible for other modules. An access is always made to the target station CPU.)

- Network module: The first five digits of the serial number are '06092' or higher.
- QCPU: The first five digits of the serial number are '06092' or higher.

*3 : This becomes valid only when 1 is set in bit 7 of Error completion type ((c) [0]).

(1) Remote RUN/STOP

| Request data (all set by the user) |
|------------------------------------|
|------------------------------------|

| Device | Item | Description | Remote RUN | Remote STOP |
|--------|------------------|--|---------------|----------------|
| | Dequest type | 0010н: When station number is specified in 🗊 [5] | 0 | |
| s2 [0] | Request type | 0030н: When all stations a group is specified in 🗊 [5] | 0 | 0 |
| @[4] | Sub-request type | 0001H: Remote RUN | 0 | 0 |
| ⊚[1] | oub request type | 0002н: Remote STOP |) | 0 |
| | | Specify whether to forcibly execute remote RUN/STOP. The forced | | |
| | | execution is a function that forces a station which has stopped by | | |
| | | remote STOP to RUN remotely from another station. | | |
| | | For remote RUN | | |
| s2 [2] | Operation mode | 0001H: No forced execution | 0 | 0 |
| | | 0003H: Forced execution (This setting can be specified for | | |
| | | remote RUN.) | | |
| | | For remote STOP | | |
| | | 0003н: (Fixed) | | |
| | | Specify the status of device memory in the CPU module only for | | |
| | | remote RUN. | | |
| | | 0000H: Not cleared (Note that the local devices are cleared.) | | |
| | | 0001H: Cleared (excluding the latch range and settings in remote RUN) | | |
| | | 0002H: Cleared (including the latch range and settings in remote | | |
| s2 [3] | Clear mode | RUN) | 0 | × |
| | | Clear mode (@[3]) allows specification to clear (initialize) the devices | | |
| | | in the CPU module at the start of CPU module operation activated by | | |
| | | remote RUN. | | |
| | | After performing the specified clear processing, CPU module runs | | |
| | | according to the setting that specified by Device Initial Value in GX | | |
| | | Works2. | | |

Response data^{*1} (all set by the system)

| Device | ltem | Description | Remote RUN | Remote STOP |
|--------|------------------|--|---------------|----------------|
| d] [0] | Request type | 0090н: When station number is specified in \textcircled{s} [5] | 0 | 0 |
| | | 00В0н: When all stations or a group is specified in \mathfrak{s} [5] | | |
| d1[1] | Sub-request type | 0001н: Remote RUN 0002н: Remote STOP | 0 | 0 |

*1: When "all stations or a group (81H to A0H, FFH)" is specified in (s) [5], no response data will be stored.

(2) Reading/writing the clock data

Request data (all set by the user)

| Device | Item | Setting data | Read clock data | Write clock data |
|--------|--|--|--------------------|---------------------|
| ⊚[0] | Request type | 0001н: Clock data read 0011н: Clock data write (When station number is specified in ⓒ)[5]) 0031н: Clock data write (When all stations or a group is specified in ⓒ)[5]) | 0 | 0 |
| ©[1] | Sub-request type | 0002н: Clock data read 0001н: Clock data write | 0 | 0 |
| @[2] | Change pattern Clock data to be changed | ① Change pattern (bit 0 to 7) Specify the items to be written in high-order byte of (2) [2] to (2) [5]. 0: Not changed 1: Changed ② Year to be changed (bit 8 to 15)*1 Store the year (last two digits) in BCD format. b15 b8 b7 b6 b5 b4 b3 b2 b1 b0 Year (00Ht099H) 0 Year (last two digits) Year (last two digits) Month Day Hour Minute Second Day of week | × | 0 |
| @[3] | | High-order 8 bits: Day (01н to 31н), low-order 8 bits: Month (01н to 12н) <u>b15 to b8 b7 to b0</u> Day (01н to 31н) Month (01н to 12н) | Х | 0 |
| ⊚[4] | Clock data to be changed (continued) | bits: Minute (00н to 59н), low-order 8 bits: Hour (00н to 23н) b15 to b8 b7 to b0 Minute (00н to 59н) Hour (00н to 23н) | × | 0 |
| € [5] | | High-order 8 bits: Day of week (00н (Sunday) to 06н (Saturday)), low- order 8 bits: Second (00н to 59н) <u>b15 to b8 b7 to b0</u> <u>Day of week (00н to 06н) Second (00н to 59н)</u> → 00н (Sun.) to 06н (Sat.) | × | 0 |

*1 : This function cannot change the first two digits of year data.

To change the year data including the first two digits, set the clock data using another function (such as GX Works2).

| Device | Item | Setting data | Read clock data | Write clock data |
|--------|------------------|---|--------------------|---------------------|
| d) [0] | Request type | 0081н: Clock data read 0091н: Clock data write (When station number is specified in ্র্যা [5]) 00В1н: Clock data write (When all stations or a group is specified in ্র্যা [5]) ^{*2} | 0 | 0 |
| @[1] | Sub-request type | 0002н: Clock data read 0001н: Clock data write | 0 | 0 |
| d)[2] | | High-order 8 bits: Month (01н to 12н), low-order 8 bits: Year (00н to 99н) ^{*3} <u>b15 to b8 b7 to b0</u> <u>Month (01н to 12н) Year (00н to 99н)</u> | 0 | × |
| @[3] | | High-order 8 bits: Hour (00H to 23H), low-order 8 bits: Day (01H to 31H) b15 to b8 b7 to b0 Hour (00H to 23H) Day (01H to 31H) Day (01 | 0 | × |
| @[4] | Read clock data | bits Second (00н to 59н), low-order 8 bits (00н to 59н) b15 to b8 b7 to b0 Second (00н to 59н) Minute (00н to 59н) | 0 | × |
| @[5] | | High-order 8 bits: (00н), low-order 8 bits: Day of week (00н (Sunday) to 06н (Saturday)) b15 to b8 b7 to b0 00н Day of week (00н to 06н) 00н (Sun.) to 06н (Sat.) | 0 | × |

Response data (all set by the system)

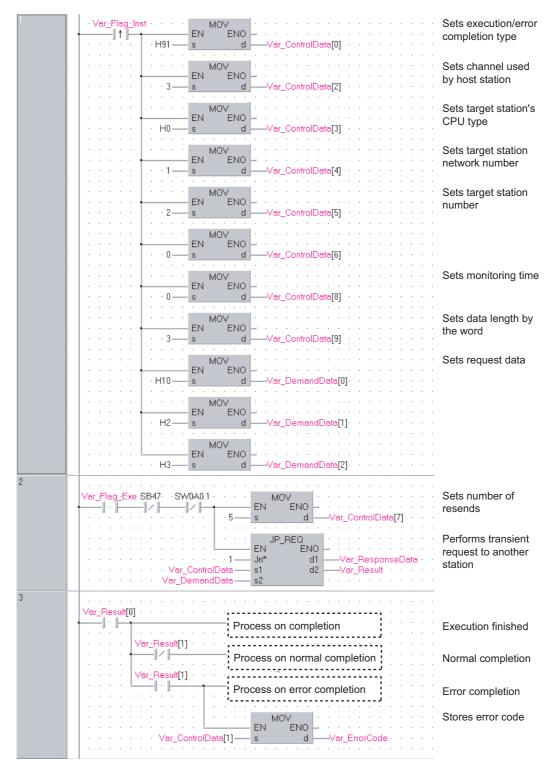
*2: When "all stations or a group (81H to A0H, FFH)" is specified in (s) [5], no response data will be stored.

*3 : Last two digits of year data

Program Example

The following program performs remote STOP to the QCPU, which is the station number 2 (target station).

[Structured ladder/FBD]



```
[ST]
IF (Var_Flag_Inst=TRUE) THEN
    MOV(TRUE,H91,Var_ControlData[0]); (* Sets execution/error completion type *)
    MOV(TRUE,3,Var_ControlData[2]);
                                        (* Sets channel used by host station *)
    MOV(TRUE,H0,Var ControlData[3]); (* Sets target station's CPU type *)
    MOV(TRUE,1,Var ControlData[4]);
                                        (* Sets target station network number *)
    MOV(TRUE,2,Var_ControlData[5]);
                                        (* Sets target station number *)
    MOV(TRUE,0,Var ControlData[6]);
    MOV(TRUE,0,Var_ControlData[8]);
                                        (* Sets monitoring time *)
    MOV(TRUE,3,Var_ControlData[9]);
                                        (* Sets data length by the word *)
    MOV(TRUE,H10,Var_DemandData[0]);(* Sets request data *)
    MOV(TRUE,H2,Var_DemandData[1]);
    MOV(TRUE,H3,Var DemandData[2]);
END_IF;
IF((Var_Flag_Exe=TRUE) AND (SB47=FALSE) AND (SW0A0.1=FALSE)) THEN
    MOV(TRUE, 5, Var ControlData[7]); (* Sets number of resends *)
    JP_REQ(TRUE,1,Var_ControlData,Var_DemandData,Var_ResponseData,Var_Result);
                                  (* Performs transient request to another station *)
END_IF;
IF(Var Result[0]=TRUE)THEN
                                        (* Execution finished *)
                . . . . . . . . . . . . . . . . . . .
                                           (* Process on completion *)
    IF(Var Result[1]=FALSE)THEN
                                        (* Normal completion *)
```

(* Process on normal completion *) ELSE (* Error completion *) (* Process on error completion *)

MOV(TRUE, Var_ControlData[1], Var_ErrorCode);(* Stores error code *) END_IF;

END_IF;

5.4 Network Dedicated Instruction

5.4.15 REQ instruction

5.4.16 ZNRD instruction

J_ZNRD

CC IE C NET/H Ether J(P)_ZNRD P: Executing condition : 1 indicates any of the following Structured ladder/FBD ST instructions. J_ZNRD JP_ZNRD J_ZNRD ΕN ENO ENO:= J_ZNRD (EN, Jn*, n1, s, n2, d1, d2); Jn* d1 n1 d2 s n2 :Bit Input argument FN. Executing condition Network number of the host station (1 to 239) ·ANY16 Jn*: n1: Target station number (1 to 64) :ANY16 s Target station's start device number where data to be read are :ANY16 stored Read data length :ANY16 n2: When the target station is Q/QnA/AnUCPU: 1 to 230 words When the target station is anything other than Q/QnA/ AnUCPU: 1 to 32 words ENO: :Bit Output argument Execution result The host station's start device number where readout data will :ANY16 d1: be stored (A contiguous area for the read data length is required.) d2: The host station's device that is turned on for one scan upon :Array of bit [0..1] completion of the instruction d2[1] also turns ON if the instruction execution has failed. Setting Internal device J ... \ Constant R, ZR U....\G.... Zn Others data *1,*2 Word Word Bit Bit n1 _ _ _ S _ _ _ _ n2 _ _ 0 _ _ _ d1) _ _ d2) *1: Local devices and file registers per program cannot be used as setting data.

*2: In addition to the setting data, the ZNRD instruction is executed using the following fixed values. Channel used by host station: Channel 1

Arrival monitoring time (monitoring time until instruction completion): 10 seconds Number of resends for arrival monitoring timeout: 5 times

Grant Function

This instruction reads data from devices of a programmable controller CPU on another station. (In units of words)

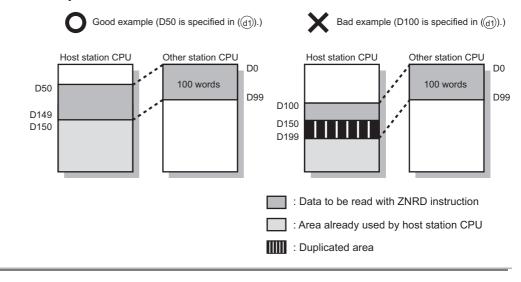
1. Specify devices of the target station's CPU within the range allowed for the host station CPU when reading data from the devices with the ZNRD instruction.

(Target station's start device number ${\scriptstyle \textcircled{\tiny sl}}$ where data to be read are stored)

- + (Read points 1) \leq (End device No. of host station's CPU^{*1})
- Specify the host station's start device number

 within the range allowed for storing read data.

(Example) When D150 and after the area in the host station's CPU has been already used



_ZNRD

Program Example

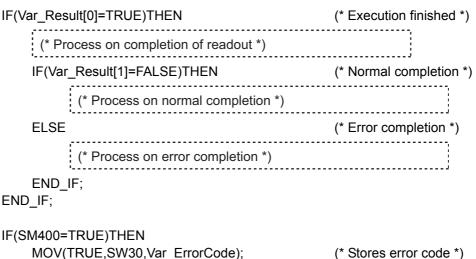
In this program example, when M101 turns ON, data in D250 to D254 of station No.4 (target station) are read out to D700 to D704 of station No.1 (host station).

M1-01- SB47-·SW0A0.3 · JP_ZNRD Performs readout ΕN ENO -||||| 1.1 -12 k · 1 — Jn* d1 -D700 Var_Result 4 n1 d2 D250 - 5 Var_Result[0] -1 ŀ Process on completion of readout No. of the Var_Result[1] —IZH Process on normal completion Propos Var_Result[1] Process on error completion -|| • ||-3 SM400 MOV Stores error code EN ENO - · -|| • ||-SW30 d -Var_ErrorCode S

[Structured ladder/FBD]

[ST]

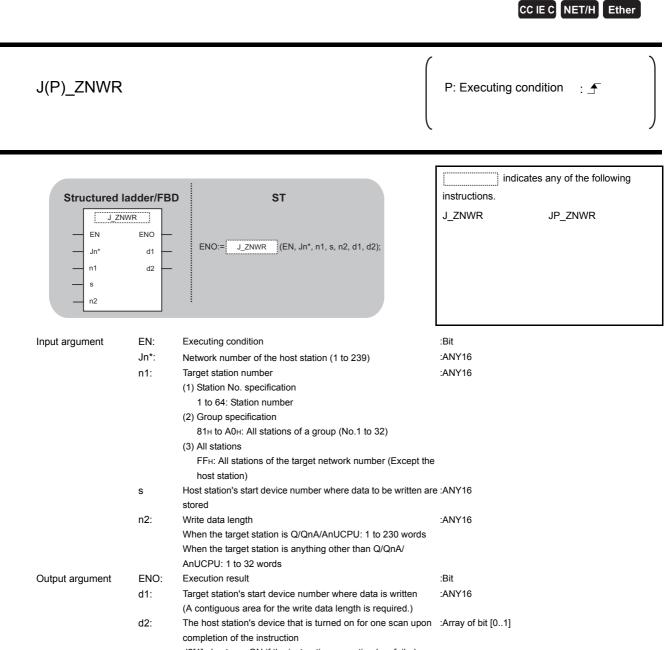
IF((M101=TRUE) &(SB47=FALSE) & (SW0A0.3=FALSE)) THEN JP_ZNRD(TRUE,1,4,D250,5,D700, Var_Result);(* Performs ZNRD instruction*) END_IF;



END_IF;

5.4.17 ZNWR instruction

J_ZNWR



d2[1] also turns ON if the instruction execution has failed.

_ZNWR

| Setting | Interna | l device | R, ZR | J[]]\[]] | | U∭\G∭ | Zn | Constant | Others |
|------------|---------|----------|-------|----------|------|---------|-------------|----------|--------|
| data *1,*2 | Bit | Word | , | Bit | Word | 0::\G:: | E 11 | K, H | |
| n1 | | 0 | | | | - | | 0 | - |
| S | - | 0 | - | | - | | | - | - |
| n2 | | 0 | | | | _ | | - | - |
| d1 | - | C |) | | | _ | | 0 | - |
| d2 | | 0 | | | | _ | | - | - |

*1: Local devices and file registers per program cannot be used as a device which is used in setting data.

*2: In addition to the setting data, the ZNWR instruction is executed using the following fixed values. Channel used by host station: Channel 2

Arrival monitoring time (monitoring time until instruction completion): 10 seconds Number of resends for arrival monitoring timeout: 5 times

Grant Function

This instruction writes data to devices of a programmable controller CPU on another station. (In units of words)

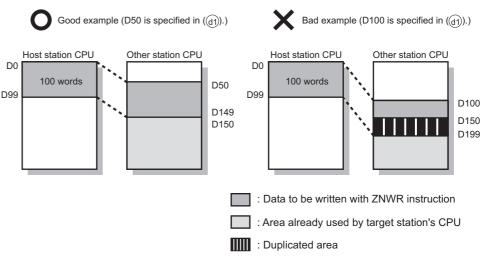
1. Specify devices of the target station's CPU within the range allowed for the host station CPU when writing data to the devices with the ZNWR instruction.

(Target station's start device number where data are written)

- + (Write points 1) \leq (End device No. of host station's CPU ^{*1})
- \star1 End device No. of the device in the host station CPU, and whose device name is same as in ${}_{\mathrm{(s)}}$

write data.

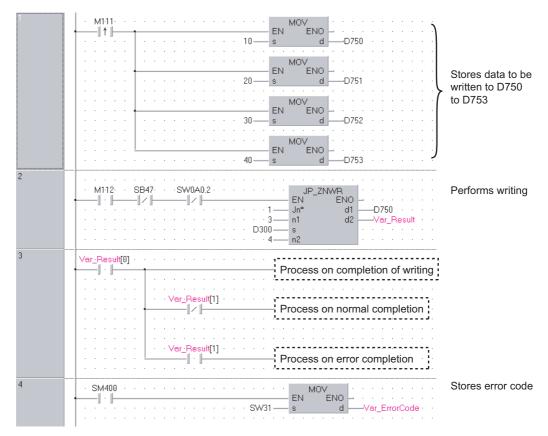
(Example) When D150 and after the area in the host station's CPU has been already used



Program Example

In this program example, when M112 turns ON, data in D750 to D753 of station No.2 (host station) are written to D300 to D303 of station No.3 (target station).

[Structured ladder/FBD]



| [ST] IF(M111=TRUE)THEN MOV(TRUE, 10, D750); MOV(TRUE, 20, D751); MOV(TRUE, 30, D752); MOV(TRUE, 40, D753); | (* Instruction flag ON *) |
|---|---|
| END_IF; | (* Stores data to be written to D750 to D753 *) |
| IF((M112=TRUE) &(SB47=FALSE) & (SW0 JP_ZNWR(TRUE,1,3,D300,4, D750, END_IF; | DA0.2=FALSE)) THEN Var_Result); (* Performs writing *) |
| IF(Var_Result[0]=TRUE)THEN | (* Completion of writing *) |
| (* Process on completion of writing | *) |
| | (* Normal completion *) |
| (* Process on normal complet | tion *) |
| ELSE | (* Error completion *) |
| (* Process on error completio | n *) |
| END_IF; END_IF; | |
| IF(SM400=TRUE)THEN MOV(TRUE,SW31,Var_ErrorCode); END_IF; | (* Stores error code *) |

5.4.18 RRUN instruction

Z_RRUN_J, Z_RRUN_U

CC IE C NET/H

| Z(P)_RRUN_ Z(P)_RRUN_ | | | P: Executing condition : |
|--|-------------------|--|---|
| Structured Z RRUN J EN Jn* s1 s2 s3 s4 | | BD ST ENO d ENO:= Z_RRUN_J (EN, Jn*, s1, s2, s3, s4, d); ENO:= Z_RRUN_U (EN, Un*, s1, s2, s3, s4, d); | indicates any of the following instructions. Z_RRUN_J ZP_RRUN_J Z_RRUN_U ZP_RRUN_U |
| Input argument | EN: Jn*: | Executing condition Network number of the target station (1 to 239, 254) 254: Network specified in "Valid module during other station access" | :Bit :String |
| | Un*: | Start I/O number of the host station network No. (00 to FE: Higher two digits when expressing the I/O number in | :String n |
| | s1: | three digits) Channel used by host station For the RRUN instruction, specify the channel used by host station that is the same as the one used for the RSTOP | :ANY16 |
| | s2: | instruction. Target station number (1) Station number specification Host station is Universal model QCPU: 1 to 120 Host station is anything other than Universal model QCPU: 1 to 64 (2) Group specification 81H to A0H: All stations of a group (No.1 to 32) (3) All stations | :ANY16 |
| | | FFн: All stations of the target network No. (Except the host station) To specify a group or all stations, specify '0000н' or '03FFн' for | r |
| | s3: | the target station's CPU type (s3). Target station's CPU type 0000H: Target station CPU/control CPU/host system CPU (Specified data are the same as '03FFH'.) 03E0H: Multi-CPU No. 1/ target station CPU (single CPU system) 03E1H: Multi-CPU No. 2 03E2H: Multi-CPU No. 3 03E3H: Multi-CPU No. 4 03E5H: Target station CPU/control CPU/cont system CPU | :ANY16 |
| Output argument | s4: ENO: d: | 03FFH: Target station CPU/control CPU/host system CPU Mode Execution result Variable that turns ON upon completion of the instruction d[1] also turns ON at the time of error completion. | :ANY16 :Bit :Array of bit [01] |

| Setting | Internal | device | R, ZR J∭∖ | | U\G | Zn | Constant | Others | |
|--------------------|----------|--------|-----------|-----|------|---------|----------|--------|---------|
| data ^{*1} | Bit | Word | N, 2N | Bit | Word | Uii\Gii | 20 | K, H | Cullero |
| (s1) | - | C |) | | | - | | 0 | - |
| s2 | - | C |) | | | - | | 0 | - |
| s 3 | - | C |) | - | | | 0 | - | |
| <u>s4</u> | - | C |) | | | - | | 0 | - |
| d | 0 | C |) | | | - | | - | - |

1: Local devices and file registers per program cannot be used as setting data.

Grant Function

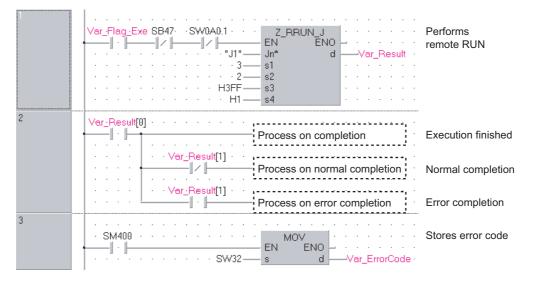
This instruction remotely switches a CPU module on another station to RUN.

Precautions

This instruction is applicable to the QJ71LP21 or QJ71BR11 with the function version B or later.

Program Example

The following program remotely switches the QCPU on the station number 2 (target station) to RUN.



[Structured ladder/FBD]

| [ST] IF((Var_Flag_Exe=TRUE) AND (SB47=FALSE) AND (Z_RRUN_J(TRUE,"J1",3,2,H3FF,H1,Var_Result) END_IF; IF(Var Result[0]=TRUE)THEN | · · · · · · · · · · · · · · · · · · · |
|--|---------------------------------------|
| · · · · · · · · · · · · · · · · · · · | () |
| (* Process on completion *) | |
| IF(Var_Result[1]=FALSE)THEN | (* Normal completion *) |
| (* Process on normal completion *) | |
| ELSE | (* Error completion *) |
| (* Process on error completion *) | |
| END_IF; | |
| END_IF; | |
| | |

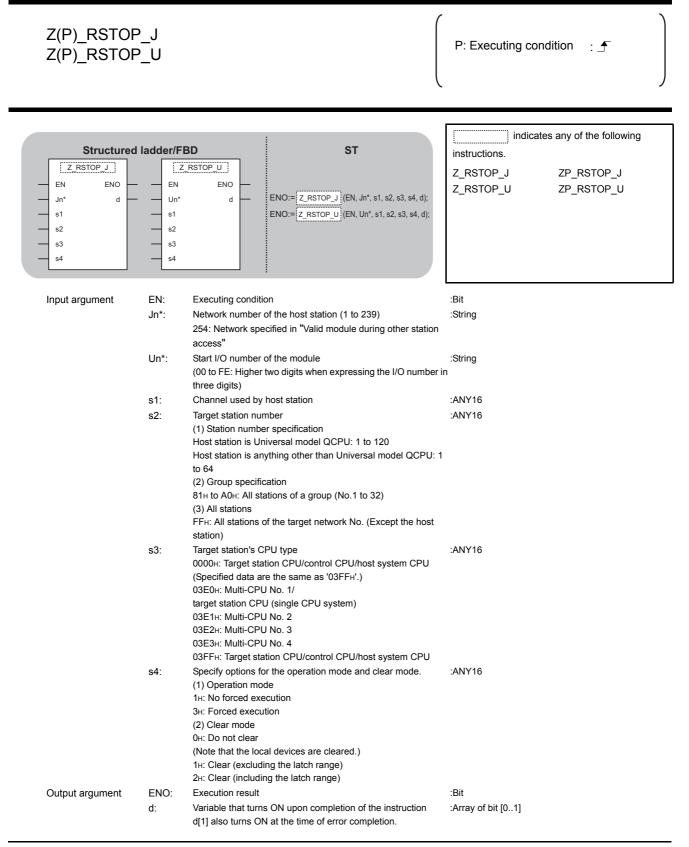
MOV(SM400,SW32,Var_ErrorCode);

(* Stores error code *)

5.4.19 RSTOP instruction

Z_RSTOP_J, Z_RSTOP_U

CC IE C NET/H



| Setting | Interna | l device | vice J | | | Zn | Constant | Others | |
|------------|---------------------------|----------|----------|--------------|-----------------|-----------------|----------------|----------|-----------|
| data *1 | ta ^{*1} Bit Word | | Bit | Word | 0: | | K, H | | |
| (s1) | - | C |) | | | - | | 0 | - |
| s2 | - | C |) | | | | | 0 | - |
| <u>s</u> 3 | - | C |) | - | | | 0 | - | |
| <u>\$4</u> | - | C |) | - | | | 0 | - | |
| d | 0 | C |) | - | | | | _ | - |
| | | | *1.1 003 | al devices a | nd file registe | are ner program | cannot be user | t as sot | ting data |

1: Local devices and file registers per program cannot be used as setting data

Grant Function

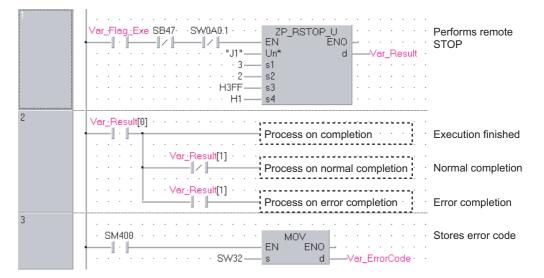
This instruction remotely switches a CPU module on another station to STOP.

Precautions

This instruction is applicable to the QJ71LP21 or QJ71BR11 with the function version B or later.

Program Example

The following program remotely switches the QCPU on the station number 2 (target station) to STOP.



[Structured ladder/FBD]

ODULE DEDICATED

[ST] IF((Var_Flag_Exe=TRUE) AND (SB47=FALSE) AND (SW0A0.1=FALSE)) THEN ZP_RSTOP_J(TRUE,"J1",3,2,H3FF,H1,Var_Result);(* Performs remote STOP *) END_IF; IF(Var_Result[0]=TRUE)THEN (* Execution finished *) (* Process on completion *) . . IF(Var_Result[1]=FALSE)THEN (* Normal completion *) (* Process on normal completion *) ELSE (* Error execution *) (* Process on error completion *) END IF; END_IF;

MOV(SM400, SW32, Var_ErrorCode); (* Stores error code *)

5.4.20 RTMRD instruction

Z_RTMRD_J, Z_RTMRD_U

CC IE C NET/H

| Z(P)_RTMRI Z(P)_RTMRI | | | | | P: Executing cond | dition : 🛓 | 7 | |
|---|---------------------|--|---|--------------------|---|--------------------------------------|------------------|--------|
| Structured EN ENO Jn* d1 s1 d2 s2 s3 | | BD ENO d1 d2 | ST ENO:= Z_RTMRD_J (EN, Jn' ENO:= Z_RTMRD_U (EN, Un | | indicate instructions. Z_RTMRD_J Z_RTMRD_U | s any of the f ZP_RTMR ZP_RTMR | D_J | ng |
| Input argument | EN: Jn*: Un*: | 254: Network s access" Start I/O numb | lition er of the host station (1 to 23 pecified in "Valid module du er of the module er two digits when expressin | ring other station | :Bit :String :String n | | | |
| | s1: s2: | | - | | :ANY16 :ANY16 | | | |
| | s3: | Target station's 0000H: Target s (Specified data 03E0H: Multi-C target station C 03E1H: Multi-C 03E2H: Multi-C 03E3H: Multi-C | station CPU/control CPU/hos are the same as '03FFH'.) PU No. 1/ :PU (single CPU system) PU No. 2 PU No. 3 | | :ANY16 | | | |
| Output argument | ENO: | Execution resu | | | :Bit | | | |
| | d1: d2: | Variable that tu | ores read clock data rns ON upon completion of s ON at the time of error con | | :Array of ANY16 [03] :Array of bit [01] | | | |
| | | Setting In data ^{*1} B | ternal device R, ZR | J∭\∭ Bit V | U∭\G∭ | Zn | Constant K, H | Others |
| | | s1 - | · 0 | | - | | 0 | - |
| | | 62 - | · O | | _ | | \bigcirc | - |
| | | <u>(53</u> – | · 0 | | | | 0 | - |
| | | - (1) | · 0 | | _ | | - | - |
| | | d2 C | | | _ | | - | - |

*1: Local devices and file registers per program cannot be used as setting data.

5

Z_RTMRD_J, Z_RTMRD_U

Grant Function

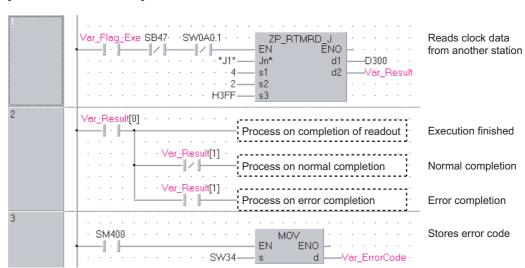
This instruction reads clock data from a CPU module on another station.

Precautions

This instruction is applicable to the QJ71LP21 or QJ71BR11 with the function version B or later.

Program Example

The following program reads out clock data from the QCPU on the station number 2 (target station) and stores the clock data in the station number 1 (host station).



[Structured ladder/FBD]

[ST]

IF((Var_Flag_Exe=TRUE) AND (SB47=FALSE) AND (SW0A0.1=FALSE)) THEN ZP_RTMRD_J(TRUE,"J1",4,2,H3FF,D300,Var_Result); (* Poads clock data from another station *)

(* Reads clock data from another station *)

END IF; IF(Var_Result[0]=TRUE)THEN (* Execution finished *) (* Process on completion of readout *) (* Normal completion *) IF(Var_Result[1]=FALSE)THEN Process on normal completion *) ELSE (* Error completion *) * Process on error completion *) END IF; END IF; MOV(SM400, SW33, Var ErrorCode); (* Stores error code *)

5.4.21 RTMWR instruction

Z_RTMWR_J, Z_RTMWR_U

CC IE C NET/H

| Z(P)_RTMW Z(P)_RTMW | | | P: Executing condition : |
|---|-------------------|--|---|
| Structured EN ENO Jn* d s1 s2 s3 s4 | | BD ST ENO:= Z.RTMWR_J d ENO:= Z.RTMWR_U (EN, Jn*, s1, s2, s3, s4, d); ENO:= Z.RTMWR_U (EN, Un*, s1, s2, s3, s4, d); | indicates any of the following instructions. Z_RTMWR_J ZP_RTMWR_J Z_RTMWR_U ZP_RTMWR_U |
| Input argument | EN: Jn*: | Executing condition Network number of the target station (1 to 239, 254) 254: Network specified in "Valid module during other station access" | :Bit :String on |
| | Un*: | Start I/O number of the host station network No. (00 to FE: Higher two digits when expressing the I/O numb three digits) | :String er in |
| | s1: s2: s3: | Channel used by host station (1 to 8) Target station number (1) Station number specification Host station is Universal model QCPU: 1 to 120 Host station is anything other than Universal model QCPU to 64 (2) Group specification 81H to A0H: All stations of a group (No.1 to 32) (3) All stations FFH: All stations of the target network No. (Except the host station) To specify a group or all stations, specify '0000H' or '03FFH the target station's CPU type 0000H: Target station CPU/control CPU/host system CPU (Specified data are the same as '03FFH'.) | st i' for :ANY16 |
| Output argument | s4: ENO: d: | (c)peonied data die the same die GOTTIT) 03EOH: Multi-CPU No. 1/ target station CPU (single CPU system) 03E1H: Multi-CPU No. 2 03E2H: Multi-CPU No. 3 03E3H: Multi-CPU No. 4 03FFH: Target station CPU/control CPU/host system CPU Variable that stores write clock data Execution result Variable that turns ON upon completion of the instruction d2[1] also turns ON at the time of error completion. | :Array of ANY16 [04] :Bit :Array of bit [01] |

| Setting | Interna | l device | R, ZR | J | 1 | U\G | Zn | Constant | Others |
|---------|---------|----------|----------------|-----|------|---------|----|----------|---------|
| data *1 | Bit | Word | , <u>2</u> , 1 | Bit | Word | 0::\G:: | 20 | K, H | Cullero |
| (s1) | - | C |) | | | - | | 0 | - |
| s2 | - | C |) | | | _ | | 0 | - |
| \$3 | - | C |) | | | _ | | 0 | - |
| s4) | - | C |) | | | _ | | - | - |
| d | 0 | C |) | | - | - | | | |

*1: Local devices and file registers per program cannot be used as setting data.

Grant Function

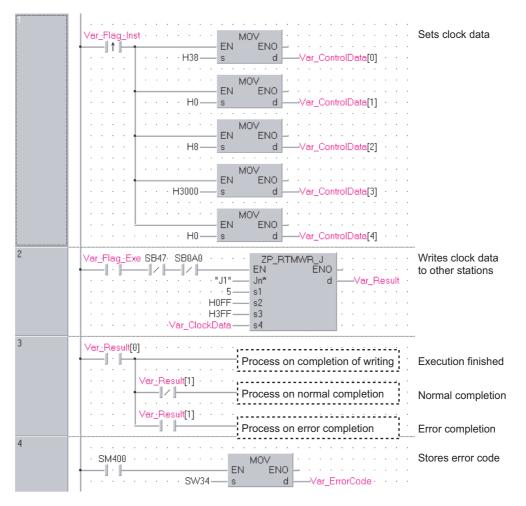
This instruction writes clock data to a CPU module on another station.

Precautions

This instruction is applicable to the QJ71LP21 or QJ71BR11 with the function version B or later.

Program Example

The following program writes the clock data (8:30:00) to all stations on the network number 1. [Structured ladder/FBD]



```
[ST]
IF (Var_Flag_Inst=TRUE) THEN
   MOV(TRUE,H38,Var_ClockData[0]); (* Sets clock data *)
   MOV(TRUE,H0,Var_ClockData[1]);
   MOV(TRUE,H8,Var ClockData[2]);
   MOV(TRUE,H3000,Var ClockData[3]);
   MOV(TRUE,H0,Var_ClockData[4]);
END IF;
IF((Var_Flag_Exe=TRUE) AND (SB47=FALSE) AND (SB0A0=FALSE)) THEN
   ZP_RTMWR_J(TRUE,"J1",5,H0FF,H3FF,Var_ClockData,Var_Result);
                             (* Writes clock data to other stations*)
END_IF;
IF(Var_Result[0]=TRUE)THEN (* Execution finished *)
       .....
    (* Process on completion of writing *)
                              /
   IF(Var_Result[1]=FALSE)THEN (* Normal completion *)
              .....
         (* Process on normal completion *)
                                 ......
   ELSE
                        (* Error completion *)
              -----
        (* Process on error completion *)
                               END IF;
END IF;
```

MOV(SM400, SW34, Var_ErrorCode);(* Stores error code *)

5.4.22 REMFR instruction

Z_REMFR

CC IE F NET/H

| Z(P)_REMFI | R | | | | | P: Executing co | ndition : 🖍 | |
|---|------|------------------|---------------------|--|---------------------|--------------------|----------------------|--------------|
| Structured | | 3D | | ST | | instructions. | tes any of the follo | owing |
| EN - EN - Jn* - n1 - n2 - n3 - n4 - n5 | EMFR | - - - - | NO:= <u>Z_rem</u> i | R (EN, Jn*, n1, n2, n3, n4, n5, d1, d | 2); | Z_REMFR | ZP_REMFR | |
| Input argument | EN: | Executin | g condition | | | :Bit | | |
| input diguniont | Jn*: | | • | ber (1 to 239) | | :String | | |
| | n1: | | number (1 | | | :ANY16 | | |
| | n2: | | | | | :ANY16 | | |
| | | | | er (1 to 120) | tion module | | | |
| | n3: | | | the target intelligent func Field Network, the higher | | :ANY16 | | |
| | | | | number in three digits. | two digits when | | | |
| | | | | ET/H, the higher three dig | aits when | | | |
| | | | | number in four digits. | , | | | |
| | n4: | | - | y start address | | :ANY16 | | |
| | | | | ddress of the buffer men | nory for the read | | | |
| | | destinati | on intellige | nt function module. | | | | |
| | n5: | Number | of read poi | nts (1 to 240 words) | | :ANY16 | | |
| Output argument | ENO: | Executio | n result | | | :Bit | | |
| | d1: | Start nur | nber of the | device that stores read of | data (host station) |) :ANY16 | | |
| | | • | | umber of the host statior | n's device that | | | |
| | d2: | | that turns (| DN upon completion of th at the time of error comp | | :Array of bit [01] | | |
| | | Setting | | al device | J\ | | | stant O |
| | | data *1 | Bit | Word R, ZR | Bit V | /ord | Zn K, | Н |
| | | n1 | - | 0 | | - | | |
| | | n2 | - | 0 | | _ | (| |
| | | n3 | I | 0 | | _ | (| |
| | | n4 | - | 0 | | _ | | |
| | | n5 | - | 0 | | - | (| \mathbf{D} |
| | | d1 | - | 0 | | - | - | - |
| | | | | | | | | |

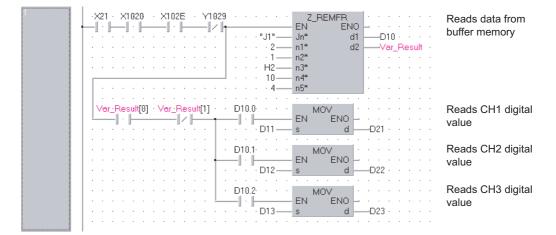
Grant Function

This instruction reads data from the buffer memory of an intelligent function module to the host station's word device (starting from (a)) on the intelligent device station/remote I/O station.

Program Example

The following program reads digital output values.

[Structured ladder/FBD]



[ST]

IF((X21=TRUE) AND (X1020=TRUE) AND (X102E=TRUE) AND (Y1029=FALSE))THEN Z_REMFR(TRUE,"J1",2,1,H2,10,4,D10,Var_Result); (* Reads data from buffer memory *) (*Reads digital values of CH1 to CH3 at once*) IF((Var_Result[0]=TRUE) AND (Var_Result[1]=FALSE))THEN IF(D10.0=TRUE)THEN MOV(TRUE,D11,D21); (* Reads CH1 digital output value *) END IF; IF(D10.1=TRUE)THEN MOV(TRUE,D12,D22); (* Reads CH2 digital output value *) END IF; IF(D10.2=TRUE)THEN MOV(TRUE,D13,D23); (* Reads CH3 digital output value *) END_IF; END_IF; END_IF;

5.4.23 REMTO instruction

Z_REMTO

CC IE F NET/H

| Z(P)_REMT | 0 | | | | P: Executing condition : _ | |
|---|---------------------------------|---------------|---------------|---|---|----|
| Structured | ladder/F | BD | | ST | indicates any of the followin instructions. | ıg |
| EN - EN - Jn* - n1 - n2 - n3 - n4 - n5 | EMTO = ENO = d1 = d2 = | - - - E | NO:= Z_REMT(|) (EN, Jn*, n1, n2, n3, n4, n5, d1, d2); | Z_REMTO ZP_REMTO | |
| Input argument | EN: | Executir | ng condition | | :Bit | |
| input diguniont | Jn*: | | | he host station (1 to 239) | :String | |
| | n1: | | l number (1 | , , | :ANY16 | |
| | n2: | | | | :ANY16 | |
| | | | station numb | | | |
| | n3: | | | he target intelligent function Field Network, the higher two | | |
| | | | | umber in three digits. | | |
| | | | | T/H, the higher three digits v | when | |
| | | | | umber in four digits. | | |
| | n4: | | | start address | :ANY16 | |
| | | | | ddress of the buffer memory | | |
| | | | | t function module. | | |
| | n5: | | | nts (1 to 240 words) | :ANY16 | |
| Output argument | ENO: | | on result | | :Bit | |
| | d1: | Start nu | mber of the | device that stores write data | (host station) :ANY16 | |
| | | Specifie | s the start n | umber of the host station's d | evice that | |
| | | stores w | vrite data. | | | |
| | d2: | | | N upon completion of the in at the time of error completion | | |
| | | Setting | | l device R, ZR | J∰\∭ U∭\G∭ Zn vu | 0 |
| | | data *1 | Bit | Word | Bit Word K,H | |
| | | n1 | - | 0 | - 0 | |
| | | n2 | - | 0 | - 0 | |
| | | n3 | - | 0 | - 0 | |
| | | n4 | - | 0 | - 0 | |
| | | n5 | - | 0 | - 0 | |
| | _ | | | | | |
| | - | d1 | - | 0 | | |

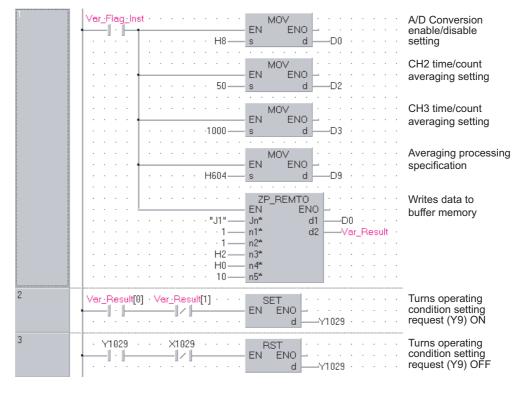
Grant Function

This instruction writes data to the buffer memory of an intelligent function module on the intelligent device station/remote I/O station.

Program Example

The following program makes the A/D conversion enable setting on channels.

[Structured ladder/FBD]



[ST]

IF(Var_Flag_Inst=TRUE)THEN MOV(TRUE,H8,D0); (* A/D Conversion enable/disable setting *) MOV(TRUE,50,D2); (* CH2 time/count averaging setting *) MOV(TRUE, 1000, D3); (* CH3 time/count averaging setting *) MOV(TRUE,H604,D9); (* Averaging processing specification *) ZP REMTO(TRUE,"J1",1,1,H2,H0,10,D0,Var Result); (* Writes data to buffer memory *) END IF; IF((Var_Result[0]=TRUE) AND (Var_Result[1]=FALSE))THEN SET(TRUE, Y1029); (* Turns operating condition setting request (Y9) ON *) END IF; IF((Y1029=TRUE) AND (X1029=FALSE))THEN RST(TRUE, Y1029); (* Turns operating condition setting request (Y9) OFF *) END_IF;

5.4.24 CCPASET instruction

G_CCPASET



| Structured ladder/FBD ST Imput argument ENO = @CCPASEF (EN, Un*, s1, s2, s3, s4, d); Imput argument EN: Executing condition EN: Start I/O number of the module : ANY16 Un*: Start I/O number of the module Start I/O number of the module : ANY16 Uoto EE: Higher two digits when expressing the I/O number in three digits Start I/O number of the host station's device that stores network : Array of ANY16 [03] Start I/O number of the host station's device that stores network : Array of ANY16 [0599] | G(P)_CCPAS | SET | | | | P: Executing col | ndition : <u>f</u> |
|---|---------------------------------------|--|---|--|---------------------|---|--------------------|
| Un*: Start I/O number of the module : ANY16 (00 to FE: Higher two digits when expressing the I/O number in three digits) : \$1: Variable that stores control data : Array of ANY16 [03] \$2: Start number of the host station's device that stores network : Array of ANY16 [07] station specification data. : Array of ANY16 [07] output argument ENO: Execution result : Bit d: Variable that turns ON upon completion of the instruction : Array of Bit [01] d[1] also turns ON at the time of error completion. : Array of Bit [01] : Other i: - - - - i: - - - | — EN — Un* — s1 — s2 — s3 | ENO - | ENO:=G_CCPAS | | 3, s4, d); | instructions. | |
| Output argument ENO: Execution result : Bit d: Variable that turns ON upon completion of the instruction : Array of Bit [01] d[1] also turns ON at the time of error completion. Setting Internal device R, ZR Jiii Uiii Giii Zn Constant Other s1 - | Input argument | Un*: Sta (00 s1: Var s2: Sta cor s3: Sta sta s4: Sta | rt I/O number of i to FE: Higher tw ee digits) iable that stores rt number of the figuration setting rt number of the tion specification rt number of the | o digits when express control data host station's device I data. host station's device data. host station's device | e that stores netwo | : ANY16 er in : Array of ANY16 [03] rk : Array of ANY16 [059 ed : Array of ANY16 [07] | 9] |
| - | Output argument | ENO: Exx d: Var d[1 Setting data ¹¹ sî sî sî sî sî sî | ecution result iable that turns C also turns ON a Internal d Bit - - - | 2N upon completion t the time of error co evice R, ZR O O | ompletion. J⊞∖⊞ | : Array of Bit [01] | Zn Constant Others |

Grant Function

This instruction sets parameters for master/local module (master station).

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|---------|-------------------------------|---|-------------------|-----------------|
| st) [0] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | - | System |
| ্রা [1] | Setting flag | Specify the validity of setting data from $_{s2}$ to $_{s3}$ in the range from b0 to b2. '0: Invalid' is specified, default parameter is applied. The supplementary setting and the network operation setting in the range from b8 to bA. $\begin{array}{c ccccccccccccccccccccccccccccccccccc$ | - | User |
| st [2] | Total number of slave station | Specify the number of connected slave stations. | 1 to 120 | User |
| st [3] | Constant link scan time | Set the constant link scan time. 0 : No setting 5 to 2000: Constant link scan time | 5 to 2000 (ms) | User |

(1) Network configuration setting data

Set the network configuration settings when network configuration setting data (b0) is enabled in the setting flag (<a>[1]).

| Device | | Item | Setting data | Setting range | Setting side |
|------------------------|-------|---|---|---------------|--------------|
| ⊚ [0] | 1st | Slave station setting information | Specify the station type and station number. b15 ~ b12 b11 ~ b8 b7 ~ b0 Station type 1 (Fixed) Station number 0 : Remote I/O station 1 : Remote device station 2 : Intelligent device station 3 : Local station | - | |
| s2 [1] | 131 | RX/RY offset | Specify the start number of RX/RY in units of 16 points. | 0 to 3FF0н | |
| s2 [2] | | RX/RY size | Specify the number of RX/RY in units of 16 points. | 0 to 2048 | |
| s2 [3] | | RWr/RWw offset | Specify the start number of RWr/RWw in units of 4 points. | 0 to 1FFCн | |
| s2 [4] | | RWr/RWw size | Specify the number of RWr/RWw in units of 4 points. | 0 to 1024 | User |
| © [5] to © [594] | | | • • • • | | 0361 |
| €2 [595] | | Slave station setting information | | | |
| © [596] | | RX/RY offset | | | |
| © [597] | 120th | RX/RY size | The same as from $\textcircled{2}$ [0] to $\textcircled{2}$ [4]. | | |
| © [598] | | RWr/RWw offset | | | |
| © [599] | | RWr/RWw size | | | |

(2) Reserved station specification data

Set the slave station as the reserved station when reserved station specification data (b1) is enabled in the setting flag (((1)).

| Device | Item | | Setting data | | | | | | | | | | Setting side | | | | | | | | |
|----------------------|--------------------------------|---|--------------|---|---|----------------------------------|----------------------------------|----------------------------------|----------------------------------|---------------------------------|--|--|--|--------------|--|---------------------------------------|---------------------------------------|---|--|------|--|
| 3 [0] to 3 [7] | Reserved station specification | Specify the reserve 0: Not specified 1: Specified (3)[0] (3)[1] (3)[2] (3)[2] (3)[3] (3)[4] (3)[5] (3)[6] (3)[6] | | b14 15 31 47 63 79 95 | b13 14 30 46 62 78 94 | 13 29 45 61 77 93 | 12 28 44 60 76 92 | 11 27 43 59 75 91 | 10 26 42 58 74 90 | 9 25 41 57 73 89 | 8 24 40 56 72 88 104 | 7 23 39 55 71 87 103 | 6 22 38 54 70 86 102 | | 4 20 36 52 68 84 100 | 3 19 35 51 67 83 99 | 2 18 34 50 66 82 98 | b0 1 17 33 49 65 81 97 | | User | |
| | | \$3[7] | _ | - | - | _ | Nu | umbe | ers ir | n the | | | | 117 e the | | | | | | | |

(3) Error invalid station setting data

Set the slave station as the error invalid station when error invalid station setting data(b2) is enabled in the setting flag (s[1])

| Device | Item | Setting data | | | | | | | | | | | Sett sic | 0 | | | | | | | |
|----------------------|---|---|---|--|---|---|----------------------------------|----------------------------------|----------------------------------|-----------------------|--|--|--|--|--|---------------------------------------|---------------------------------------|----------------------|--|----|--|
| ⊛ [0] to ⊛ [7] | Error invalid station setting ^{*1} | (4) (2) (3) (3) (4) (4) (5) | (Def b15 16 32 48 64 80 96 | fault b14 15 31 47 63 79 95 | b) b13 14 30 46 62 78 94 | b12 13 29 45 61 77 93 | 12 28 44 60 76 92 | 11 27 43 59 75 91 | 10 26 42 58 74 90 | 57 73 89 105 | 8 24 40 56 72 88 104 | 7 23 39 55 71 87 103 | 6 22 38 54 70 86 102 | 5 21 37 53 69 85 101 | 4 20 36 52 68 84 100 | 3 19 35 51 67 83 99 | 2 18 34 50 66 82 98 | 33 49 65 81 | | Us | |
| | | | | | | | Nı | umbe | ers i | h the | tabl | e inc | licat | e the | stat | ion r | numl | bers. | | | |

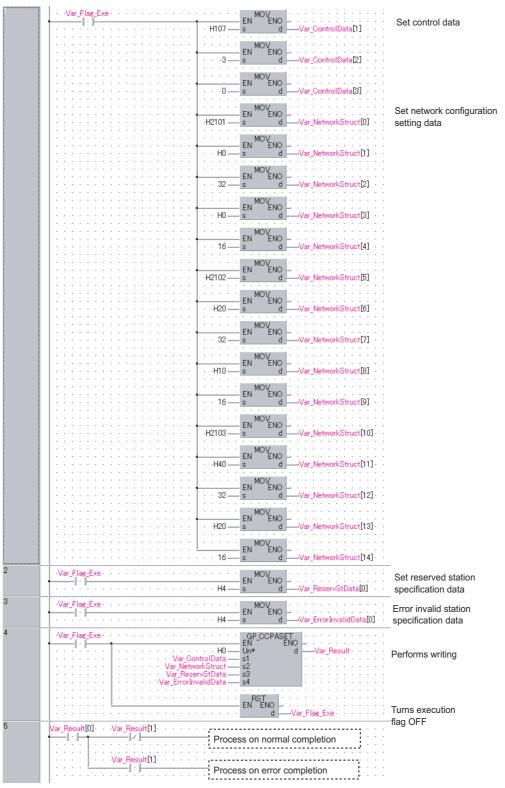
*1: Reserved station specification has a priority when an error invalid station and reserved station are specified for the same station.

Program Example

The following program sets parameters for master station of network No.1 when Var_Flag_Exe turns ON.

(Total number of slave stations is 3.)

[Structured ladder/FBD]



```
[ST]
IF( Var_Flag_Exe = TRUE ) (* Execution flag *)
    MOV(TRUE, H107, Var ControlData[1]);
                                               (* Sets control data *)
    MOV( TRUE, 3, Var_ControlData[2]);
    MOV(TRUE, 0, Var ControlData[3]);
                                       (* Sets data of network configuration setting *)
    MOV( TRUE, H2101, Var_NetworkStruct[0] );
    MOV( TRUE, H0, Var NetworkStruct[1] );
    MOV( TRUE, 32, Var_NetworkStruct[2] );
    MOV( TRUE, H0, Var_NetworkStruct[3] );
    MOV(TRUE, 16, Var NetworkStruct[4]);
    MOV( TRUE, H2102, Var NetworkStruct[5] );
    MOV( TRUE, H20, Var_NetworkStruct[6 ] );
    MOV( TRUE, 32, Var NetworkStruct[7] );
    MOV( TRUE, H10, Var_NetworkStruct[8] );
    MOV( TRUE, 16, Var NetworkStruct[9] );
    MOV(TRUE, H2103, Var NetworkStruct[10]);
    MOV( TRUE, H40, Var_NetworkStruct[11] );
    MOV( TRUE, 32, Var_NetworkStruct[12] );
    MOV( TRUE, H20, Var_NetworkStruct[13] );
    MOV( TRUE, 16, Var NetworkStruct[14] );
END IF;
IF( Var_Flag_Exe = TRUE ) (* Execution flag *)
    MOV( TRUE, H4, Var ReservStData[0] );
                                       (* Sets data of reserved station specification *)
END IF;
IF( Var_Flag_Exe = TRUE ) (* Execution flag *)
    MOV(TRUE, H4, Var ErrorInvalidData[0]);
                                      (* Sets data of error invalid station setting *)
END IF;
IF( Var_Flag_Exe = TRUE ) (* Execution flag *)
    GP CCPASET(TRUE, H0, Var ControlData, Var NetworkStruct, Var ReservStData,
                       Var_ErrorInvalidData, Var_Result);
                                                                 (* Performs writing *)
    RST( TRUE, Var Flag Exe );
                                         (* Turns execution flag OFF *)
END IF;
IF(Var Result[0]=TRUE)THEN
                                  (*Execution finished *)
    IF(Var_Result[1]=FALSE)THEN(* Normal completion *)
      (* Process on normal completion *)
    ELSE
                                   (* Error completion *)
       Process on error completion *)
    END_IF;
```

END_IF;

5.4.25 OPEN instruction

ZP_OPEN

Ether

| ZP_OPEN | | | | Exe | cuting condition | : | J |
|---------------------------------------|------------------|---|--|------------------------------|---|-----------------|---------|
| · · · · · · · · · · · · · · · · · · · | OPEN ENO d | ENO:= ZP_0 | ST PEN (EN, Un*, s1, s2, d); | ZP_C | DPEN | ollowing instru | uction. |
| Input argument | Un*: Star (00 | ecuting condition Int I/O number of the I to FE: Higher two c ee digits) | e module ligits when expressing the I/O | :Bit :String number in | | | |
| | | nnection number (1 | to 16) | :ANY1 | 6 | | |
| | | iable that stores co | ntrol data | - | of ANY16 [09] | | |
| Output argument | | ecution result | the second states of the states | :Bit | - + + + + + + + + + + + + + + + + + + + | | |
| | | | upon completion of the instru- ne time of error completion. | iction :Array | of bit [01] | | |
| | Setti | ing Internal d | evice | J\ | uiii.ciii Zn | Constant | Others |
| | data | a ^{*1} Bit | Word R, ZR Bit | Word | U\G Zn | К, Н | Others |
| | (s1) |) – | 0 | | _ | 0 | - |
| | s2 |) – | 0 | | _ | - | - |
| | d |) () | 0 | | _ | - | - |

Grant Function

This instruction establishes (opens) a connection with external device for data communication.

Control Data

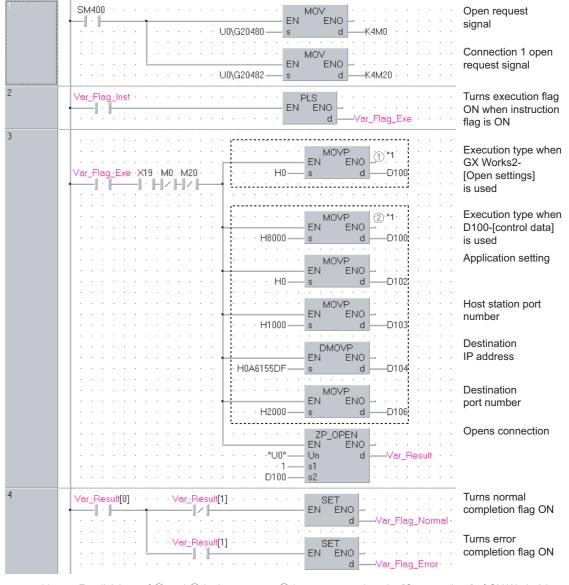
| Device | Item | Setting data | Setting range | Setting side |
|---------------|------------------------------------|--|---------------------------|--------------|
| | | Specify whether to use the parameter values set by GX Works2 or the | | |
| @[0] | Execution type/ Completion type | setting values of the following control data (@[2] to @[9]) at open processing of a connection. 0000H: Uses the parameter set in [Open settings] of GX Works2. | 0000н, 8000н | User |
| | | 8000н: Uses the settings of control data @ [2] to @ [9]. | | |
| @[1] | Completion status | The instruction application status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | - | System |
| | | Specify the application of connection. | | |
| | | b15 b14 b13 to b10 b9 b8 b7 b6 to b2 b1 b0 (6) 0 (5) (4) (3) 0 (2) (1) | | |
| | | ① Application of fixed buffer0: For sending, or fixed buffer is not used in communication | | |
| | | 1: For receiving | | |
| | | ② Check of existence of the target 0: Not checked 1: Checked | | |
| | | ③ Pairing open setting | | |
| | Application setting area | 0: No pairing open | (See the left | User |
| s2 [2] | Application setting area | 1: Pairing open | column.) | USEI |
| | | ④ Communication method (protocol)0: TCP/IP1: UDP/IP | | |
| | | © With/without procedure in fixed buffer communication | | |
| | | 0: Procedural communication | | |
| | | 1: Nonprocedural communication | l | |
| | | 6 Open system | | |
| | | 00: Active open or UDP/IP | | |
| | | 10: Unpassive open | | |
| | | 11: Fullpassive open | 401н to | |
| | | | 401н to 1387н, | |
| s2 [3] | Host station port No. | Specify the port number of the host station. | 138Bн to | User |
| | | | FFFEH | |
| 0.000 | | | 1н to | |
| <u>s</u> 2[4] | Destination IP address | Specify the IP address of the external device. | FFFFFFFH | User |
| s2 [5] | | | (FFFFFFFFн: broadcast) | |
| | | | 401H to | |
| | Destination port No | Specify the part number of the external douise | FFFFH | Lleor |
| s2 [6] | Destination port No. | Specify the port number of the external device. | (FFFFн: | User |
| | | | broadcast) | |
| s2 [7] | Destination Ethernet | | n | |
| to | address | Specify the Ethernet address of the external device. | 00000000000000H | User |
| s2 [9] | | | FFFFFFFFFFF | |

Program Example

The following program opens the connection 1 for TCP/IP communication using the Active open process.

(The I/O signals of the Ethernet module are X/Y00 to X/Y1F)

[Structured ladder/FBD]



*1:

For divisions of (1) and (2) in the program, (1) is necessary when the [Open settings] of GX Works2 is used and (2) is necessary when it is not used.

```
[ST]
IF(SM400=TRUE)THEN
                (* Always ON *)
    MOV(TRUE,U0\G20480,K4M0);
                (* Open completed signal/connection 1 open completion signal *)
    MOV(TRUE,U0\G20482,K4M20);
                (* Open request signal/connection 1 open request signal *)
END IF;
IF(Var_Flag_Inst=TRUE)THEN
                                 (* When instruction flag is ON*)
    PLS(TRUE,Var_Flag_Exe);
                                 (* Turns execution flag ON *)
END IF;
IF((Var_Flag_Exe=TRUE) AND (X19=TRUE)
                            (* Execution flag/initialization normal completion signal *)
    AND (M0=FALSE) AND (M20=FALSE))THEN
          (* Connection 1 open completion signal/connection 1 open request signal *)
               .....
(1)*1
            (*Use GX Works2-[Open settings]*)
    MOVP(TRUE,H0,D100);
            (*Execution type*)
            (2)*1
            (*Use D100-[control data]*)
    MOVP(TRUE, H8000, D100);
            (*Execution type*)
    MOVP(TRUE,H0,D102);
            (*Application setting*)
    MOVP(TRUE,H1000,D103);
            (*Host station port number*)
    DMOVP(TRUE,H0A6155DF,D104);
            (*Destination IP address*)
    MOVP(TRUE,H2000,D106);
            (*Destination port number*)
    ZP OPEN(TRUE,"U0",1,D100,Var Result); (* Opens connection *)
END IF;
IF(Var Result[0]=TRUE)THEN
                                             (* Execution finished *)
    IF(Var Result[1]=FALSE)THEN
                                             (* Normal completion *)
                                            (* Turns normal completion flag ON *)
          SET(TRUE, Var_Flag_Normal);
    END IF;
    IF(Var_Result[1]=TRUE)THEN
                                            (* Error completion *)
          SET(TRUE, Var_Flag_Error);
                                            (* Turns error completion flag ON *)
    END IF;
END_IF;
```

*1: For divisions of ① and ② in the program, ① is necessary when the [Open settings] of GX Works2 is used and ② is necessary when it is not used.

5.4.26 CLOSE instruction

ZP_CLOSE

Ether

| ZP_CLOSE | | | | | | | Executing cond | ition : | ſ | |
|-------------------------------------|--------------------|---|----------------------------|--------------------------------|--------------|-----------------|--------------------------------|----------------|------------------|----------|
| Structured EN Un* s1 s2 | d ladder/F | _ | | ST | 1, s2, d); | | ZP_CLOSE | tes the follow | wing instr | uction. |
| Input argument | EN: Un*: s1: | Executing co Start I/O nun (00 to FE: Hi three digits) Connection | nber of the gher two di | gits when exp | pressing the | e I/O number i | :Bit :String n :ANY16 | | | |
| | s2: | Variable that | | | | | :Array of ANY16 [01 | 1 | | |
| Output argument | ENO: | Execution re | | | | | :Bit | | | |
| | d: | | | ipon complet e time of erro | | | :Array of bit [01] | | | |
| | | Setting data ^{*1} | Internal de Bit | Word | R, ZR | J∭∖∭ Bit V | /ord | Zn | Constant K, H | Others |
| | Ī | <u>s1</u> | - | 0 | | | _ | | 0 | - |
| | 1 | s2 | - | 0 | | | _ | | _ | - |
| | Ť | d | 0 | 0 | | | _ | | _ | _ |
| | | Ŭ | | | *1: Local d | evices and file | e registers per program | cannot be us | sed as set | tina dat |

Grant Function

This instruction shuts off (closes) a connection with external device during data communication.

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|--------|-------------------|---|---------------|--------------|
| s2 [0] | System area | - | _ | - |
| s2[1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | - | System |

Program Example

The following program closes the connection 1.

(The I/O signals of the Ethernet module are X/Y00 to X/Y1F)

[Structured ladder/FBD]

| | Var_Flag_Open Connection 1 I · I EN EN Close timing Var_Flag_CloseTiming Close timing Close timing |
|---|---|
| 2 | Var_Flag_Close Timing Var_Flag_OpenOK Closing connection 1 from external device d Var_Flag_Close |
| 3 | Var_Flag_Inst Close instruction Image: Inst Instruction Image: Inst Inst Inst Inst Inst Inst Inst Inst |
| 4 | Var_Flag_Inst2 · Var_Flag_Open · · · · · ZP_CLOSE Closes connection |
| | Var_Flag_Close · Var_Flag_Exe |
| 5 | Var_Result[0] Var_Result[1] Turns normal Image: Set in the second se |
| | Var_Result[1] EN EN Turns error completion flag ON |
| | RST Turns execution EN ENO d Var_Flag_Exe |

[ST] (* Connection 1 open completion signal *) IF(Var_Flag_Open=TRUE)THEN PLF(TRUE,Var_Flag_CloseTiming); (* Connection 1 close timing *) END_IF; IF((Var Flag CloseTiming=TRUE) AND (Var Flag OpenOK=TRUE))THEN (* Connection 1 close timing/open instruction normal completion *) PLS(TRUE,Var_Flag_Close); (* Closing connection from external device *) END IF; IF(Var_Flag_Inst=TRUE)THEN (* Close instruction *) PLS(TRUE,Var_Flag_Inst2); (* Close instruction 1PLS *) END IF; IF(((Var_Flag_Inst2=TRUE) AND (Var_Flag_Open=TRUE)) (* Close instruction 1PLS/connection 1 open completion signal *) OR ((Var Flag Close=TRUE) AND (Var Flag Exe=FALSE)))THEN (* Closing connection 1 from external device/CLOSE instruction is in execution *) ZP CLOSE(TRUE,"U0",1,Var ControlData,Var Result); (* Closes connection *) SET(TRUE,Var_Flag_Exe); (* Turns execution flag ON *) END IF; IF(Var_Result[0]=TRUE)THEN (* Execution finished *) IF(Var_Result[1]=FALSE)THEN (* Normal completion *) SET(TRUE, Var_Flag_Normal);(* Turns normal completion flag ON *) END_IF; IF(Var Result[1]=TRUE)THEN (* Error completion *) (* Turns error completion flag ON *) SET(TRUE, Var_Flag_Error); END IF; RST(TRUE, Var Flag Exe); (* Turns execution flag OFF *) END_IF;

ZP_BUFRCV

5.4.27 BUFRCV instruction

| | | | | | | | | Eth | er |
|-------------------------------------|----------------------------------|-------------------------------|--|--|----------------------|--|---------------|------------------|-------|
| ZP_BUFRC | V | | | | | Executing condi | tion | : 🛧 | |
| Structured EN Un* s1 s2 | BUFRCV ENO d1 d2 | _ | = ZP_BUFR | ST <u>CV</u> (EN, Un*, s1, s2, | d1, d2); | ZP_BUFRCV | tes the follo | wing instru | uctio |
| Input argument | EN: Un*: | | mber of the | | ising the I/O number | :Bit :String in | | | |
| Output argument | s1: s2: ENO: d1: d2: | Variable that | number (1 t stores co esult r of the de t turns ON | | of the instruction | :ANY16 :Array of ANY16 [01] :Bit :ANY16 :Array of bit [01] | | | |
| | | Setting data ^{*1} | Internal o Bit | device R, Z Word | R Jiii\iii Bit | U\G | Zn | Constant K, H | Oth |
| | ĺ | s1 | - | 0 | | _ | | 0 | - |
| | _ | s2 | - | 0 | | - | | - | - |
| | - | (1) | - | 0 | | _ | | - | - |
| | | d2 | 0 | 0 | | - | | - | |

 * 1: Local devices and file registers per program cannot be used as setting data.

Grant Function

This instruction reads receive data from external device in fixed buffer communication. This instruction is used in a main program. AODULE DEDICATED

ZP_BUFRCV

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|--------------------------|-------------|--|---------------|--------------|
| s2 [0] | System area | - | - | - |
| | | The instruction completion status is stored. | | |
| s2 [1] Completion status | | 0 : Normal completion | - | System |
| | | Other than 0 : Error completion (error code) | | |

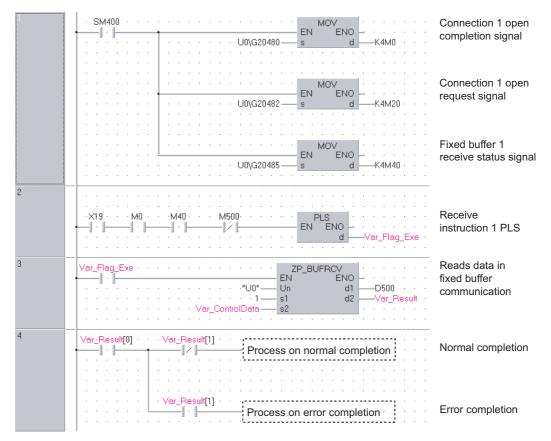
| Device | Item | Setting data | Setting range | Setting side |
|--------------------------|--------------|--|----------------------------|--------------|
| @) +0 | System area | Data length of the data read from the fixed buffer data area is stored. (Data length becomes the number of words or the number of bytes depending on the procedure used in fixed buffer communication.) With procedure (communication in binary code): The number of words With procedure (communication in ASCII code): The number of words | - 1 to 1017 1 to 508 | System |
| | | Nonprocedural communication (communication in binary code): The number of bytes | 1 to 2046 | |
| (d1) +1 to (d1) +n | Receive data | Data read from the fixed buffer data area are stored in ascending address order. | - | System |

Program Example

The following program reads out receive data from the fixed buffer of the connection 1.

(The I/O signals of the Ethernet module are X/Y00 to X/Y1F)

[Structured ladder/FBD]



5

```
[ST]
IF(SM400=TRUE)THEN
                (* Always ON *)
    MOV(TRUE,U0\G20480,K4M0);
                (* Open completion signal/connection 1 open completion signal *)
    MOV(TRUE,U0\G20482,K4M20);
                (* Open request signal/connection 1 open request signal *)
    MOV(TRUE,U0\G20485,K4M40);
                (* Fixed buffer receive status signal/fixed buffer 1 receive status signal *)
END_IF;
(* Program to receive fixed buffer number 1 (main program) *)
IF((X19=TRUE) AND (M0=TRUE) AND (M40=TRUE) AND (M500=FALSE))THEN
    (* Initialization normal completion signal/connection 1 normal open completion signal *)
    (* Fixed buffer 1 receive status signal/receive instruction completion signal *)
    PLS(TRUE,Var_Flag_Exe);
                (* Receive instruction 1PLS *)
END IF;
IF(Var_Flag_Exe=TRUE)THEN
                (* Receive instruction 1PLS *)
    ZP_BUFRCV(TRUE,"U0",1,Var_ControlData,D500,Var_Result);
                (* Reads data in fixed buffer communication *)
END IF;
IF(Var Result[0]=TRUE)THEN
                                       (* Execution finished *)
                                        (* Normal completion *)
    IF(Var_Result[1]=FALSE)THEN
             * Process on normal completion *)
                                       (* Error completion *)
    ELSE
                         (* Process on error completion *)
    END IF:
END IF;
```

5.4.28 BUFRCVS instruction

Z_BUFRCVS

Ether

Z_BUFRCVS

| Structured EN Un* s | I ladder/FE ENO - d - | | ST Z_BUFRCVS (EN, Un*, s, d); | Z_BUFRCVS | the following instruction. |
|------------------------------|-----------------------------|--|----------------------------------|------------------------------|----------------------------|
| Input argument | EN: Un*: | Executing condition Start I/O number of (00 to FE: Higher three digits) | | :Bit :String number in | |
| | s: | Connection numb | per (1 to 16) | :ANY16 | |
| Output argument | ENO: | Execution result | | :Bit | |
| | d: | Start number of th | ne device that stores read data | :ANY16 | |
| | | Setting Inter data ^{*1} Bit | R, ZR Bit | J[]]\[] Word | Zn Constant K, H Other |
| | | s – | 0 | - | 0 - |
| | | | | | |

Grant Function

This instruction reads receive data from external device in fixed buffer communication. This instruction is used in an interrupt program.

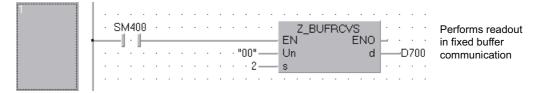
| Device | Item | Setting data | Setting range | Setting side |
|------------------------|---------------------|---|----------------------------|--------------|
| (d)+0 | Receive data length | Data length of the data read from the fixed buffer data area is stored. (Data length becomes the number of words or the number of bytes depending on the procedure used in fixed buffer communication.) With procedure (communication in binary code): The number of words With procedure (communication in ASCII code): The number of | - 1 to 1017 1 to 508 | System |
| | | words Nonprocedural communication (communication in binary code): The number of bytes | 1 to 2046 | |
| (d) +1 to (d) +n | Receive data | Data read from the fixed buffer data area are stored in ascending address order. | - | System |

Program Example

The following program reads receive data from the fixed buffer of the connection 2.

(The I/O signals of the Ethernet module are X/Y00 to X/Y1F)

[Structured ladder/FBD]

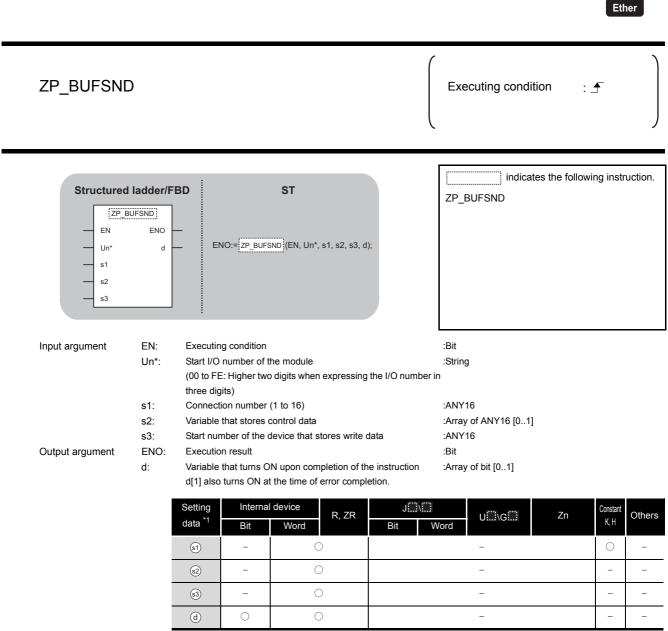




Z_BUFRCVS(SM400,"00",2,D700);(* Reads data in fixed buffer communication *)

ZP_BUFSND

5.4.29 BUFSND instruction



*1: Local devices and file registers per program cannot be used as setting data.

This instruction sends data to external device in fixed buffer communication.

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|--------|-------------------|--|---------------|--------------|
| s2 [0] | System area | - | _ | - |
| | | The instruction completion status is stored. | | |
| s2[1] | Completion status | 0 : Normal completion | - | System |
| | | Other than 0 : Error completion (error code) | | |

(1) Send data

| Device | Item | Setting data | Setting range | Setting side |
|----------------------|------------------|--|---|--------------|
| ്ര +0 | Send data length | Data length of the data read from the fixed buffer data area is stored. (Data length becomes the number of words or the number of bytes depending on the procedure used in fixed buffer communication.) With procedure (communication in binary code): The number of words With procedure (communication in ASCII code): The number of words Nonprocedural communication (communication in binary code): The number of bytes | - 1 to 1017 1 to 508 1 to 2046 | User |
| s3 +1 to s3 +n | Send data | Specify the send data. | - | User |

Program Example

The following program sends data from the fixed buffer of the connection 1.

(The I/O signals of the Ethernet module are X/Y00 to X/Y1F)

[Structured ladder/FBD]

| 1 | X19 Var_Flag_Open Send instruction Var_Flag_Open PLS Send instruction EN EN EN |
|---|--|
| | d —Var_Flag_Inst |
| 2 | Ver_Flag_Inst · · · · · · · · · · · · · · · · · · · |
| | · · · · · · · · · · · · · · · · · · · |
| | MOV Sets send data |
| | · · · · · · · · · · · · · · · · · · · |
| | EN ENO |
| | |
| | · · · · · · · · · · · · · · · · · · · |
| | · · · · · · · · · · · · · · · · · · · |
| | Var_Result communication |
| 3 | · · · · · · · · · · · · · D300 — <u>s3</u> · · · · · · · |
| Ŭ | Var_Result[0] Var_Result[1] Image: Ima |
| | Ver_Result[1] Process on error completion Error completion |

```
[ST]
IF((X19=TRUE) AND (Var_Flag_Open=TRUE))THEN
    (* Initialization normal completion signal/connection 1 open completion signal*)
    PLS(TRUE,Var_Flag_Inst);
                (* Send instruction 1PLS *)
END IF;
IF(Var_Flag_Inst=TRUE)THEN
                (* Send instruction 1PLS *)
    MOV(TRUE,3,D300);
                (* Sets data length (number of words) *)
    MOV(TRUE,1234,D301);
                (* Sets send data *)
    MOV(TRUE,5678,D302);
                (* Sets send data *)
    MOV(TRUE,8901,D303);
                (* Sets send data *)
    ZP BUFSND(TRUE,"U0",1,Var ControlData,D300,Var Result);
                (* Sends data in fixed buffer communication *)
END_IF;
IF(Var_Result[0]=TRUE)THEN
                                       (* Execution finished *)
    IF(Var_Result[1]=FALSE)THEN
                                      (* Normal completion *)
                    -----
           (* Process on normal completion *)
    ELSE
                                       (* Error completion *)
                                     -----
           (* Process on error completion *)
    END_IF;
END_IF;
```

5-234

ZP_ERRCLR

Fther

5.4.30 ERRCLR instruction

| ZP_ERRCLR | | | | | Executing condition : |
|--|---------------------------|----------|--|--|--|
| Structured la ZP_ERF EN Un* s | | ENO:= ZP | ST ERRCLR (EN, Un*, s, d); | | ZP_ERRCLR |
| Input argument | Un*: Start I/ | | the module o digits when expressing t | he I/O number | :Bit :String in |
| s: Variable that stores control data Output argument ENO: Execution result d: Variable that turns ON upon completion of the instr d[1] also turns ON at the time of error completion. | | | tion. | :Array of ANY16 [07] :Bit :Array of bit [01] | |
| | Setting data *1 (\$ | Bit | I device Word C | J\ Bit | UIII\GIII Zn Constant Others |
| | d | 0 | 0 | | - ile registers per program cannot be used as setting dat |

*1: Local devices and file registers per program cannot be used as setting data.

Grant Function

This instruction turns OFF the LED on Ethernet module and clears error information stored in the buffer memory.

Control Data

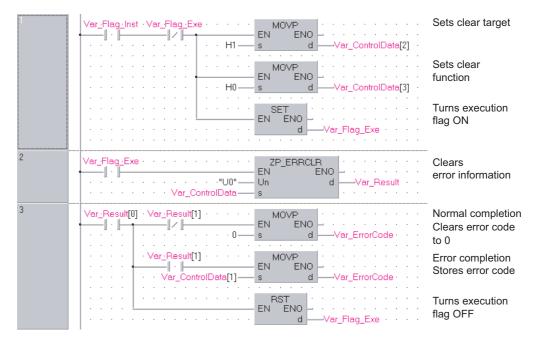
| Device | Item | Setting data | Setting range | Setting side |
|------------------------|------------------------------|--|---------------------------|--------------|
| s [0] | System area | - | - | - |
| ঙ[1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| s [2] | Clear target specification | Specify the error information to be cleared. 0000н: Initial error code 0001н to 0010н: Open error code of the corresponding connection 0100н: Error log block area 0101н: Communication status - Status by protocol 0102н: Communication status - E-mail reception status 0103н: Communication status - E-mail transmission status FFFFH: Clears all of the above. | (See the left column.) | User |
| ঙ [3] | Clear function specification | Specify the function to be cleared. 0000н: [COM.ERR] LED is turned OFF and an error code is cleared. FFFFн: Error log clear | 0000н, FFFFн | User |
| \$ [4] to \$ [7] | System area | - | - | - |

Program Example

The following program clears the open error code of the connection 1.

(The I/O signals of the Ethernet module are X/Y00 to X/Y1F)

[Structured ladder/FBD]



| <pre>[ST] IF((Var_Flag_Inst=TRUE) AND (Var_Flag_Exe=F, MOVP(TRUE,H1,Var_ControlData[2]); MOVP(TRUE,H0,Var_ControlData[3]); SET(TRUE,Var_Flag_Exe);</pre> | |
|--|-------------------------------------|
| END_IF; | |
| IF(Var_Flag_Exe=TRUE)THEN | |
| ZP_ERRCLR(TRUE,"U0",Var_ControlData,V | /ar_Result); |
| | (* Clears error information *) |
| END_IF; | |
| IF(Var_Result[0]=TRUE)THEN | (* Execution finished *) |
| IF(Var_Result[1]=FALSE)THEN | (* Normal completion *) |
| MOVP(TRUE,0,Var_ErrorCode); | (* Clears error code to 0 *) |
| END_IF; | |
| IF(Var_Result[1]=TRUE)THEN | (* Error completion *) |
| MOVP(TRUE,Var_ControlData[1],Var_ | _ErrorCode);(* Stores error code *) |
| END_IF; | |
| RST(TRUE,Var_Flag_Exe); | (* Turns execution flag OFF *) |
| END_IF; | |

5.4.31 ERRRD instruction

ZP_ERRRD

Ether

| ZP_ERRRD | | | | | | Executing condition | : _ | |
|--|---------------------------------|-------------------------------|-----------------|--|---------------------|----------------------|--------------------------|--|
| Structured EN Un* s | I ladder/F ERRRD ENO d | | NO:= <u>ZP_</u> | ST ERRRD (EN, Un*, s, d); | | ZP_ERRRD | e following instruction. | |
| Input argument | EN: Un*: | Executing c Start I/O nu | | he module | | :Bit :String | | |
| | UII . | | ligher two | o digits when expressi | ng the I/O number i | - | | |
| | s: | Variable that | at stores o | control data | | :Array of ANY16 [07] | | |
| Output argument | ENO: | Execution result | | | | :Bit | | |
| | d: | | | N upon completion of the time of error com | | :Array of bit [01] | | |
| | | Setting data ^{*1} | Internal Bit | device R, ZR Word | J∭\∭ Bit V | U::::\G:::: | Zn Constant Others | |
| | | (\$) | - | 0 | | _ | | |
| | | b | 0 | 0 | | - | | |
| *1: Local devices and file registers per program cannot be use | | | | | | | | |



This instruction reads the error information stored in the buffer memory of the Ethernet module.

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|----------------------|---------------------------------------|--|---------------|--------------|
| s [0] | System area | - | - | - |
| ©[1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | - | System |
| ঙ[2] | Read information specification | Specify the error information to be read. 0 : Initial error code 1 to 16 : Open error code of the corresponding connection | 0, 1 to 16 | User |
| sি[3] | Read target information specification | Specify the target error information to be read. 0000н: Latest error information | 0000н | User |
| ি [4] | Error information | The read error information is stored. 0000н : No error Other than 0000н : Error code | - | System |
| \$[5] to \$[7] | System area | - | _ | _ |

Program Example

The following program reads the open error code of the connection 1.

(The I/O signals of the Ethernet module are X/Y00 to X/Y1F)

[Structured ladder/FBD]

| 1 | Var_Flag_Inst Var_Flag_Exe Image: Inst Var_ControlData[2] | Sets open error code |
|---|--|---|
| | H0—s d —Var_ControlData[3] | Sets latest error information |
| | EN EN EN d | Turns execution flag ON |
| 2 | Var_Flag_Exe ZP_ERRD I EN Wor_ControlData s | Reads error information |
| 3 | Var_Result[0] Var_Result[1] I Var_ControlData[4] S d Var_Result[1] Var_ErrorInfo Var_Result[1] MOVP Var_ControlData[4] S Var_ErrorInfo Var_ErrorCode | Normal completion Stores error information Error completion Stores error code |
| | BST EN ENO d | Turns execution flag OFF |

```
[ST]
IF((Var_Flag_Inst=TRUE) AND (Var_Flag_Exe=FALSE))THEN
    MOVP(TRUE,H1,Var_ControlData[2]);
                (* Sets open error code of connection number 1 *)
    MOVP(TRUE,H0,Var_ControlData[3]);
                (* Sets latest error information *)
    SET(TRUE,Var_Flag_Exe);
                                  (* Turns execution flag ON*)
END IF;
IF(Var_Flag_Exe=TRUE)THEN
    ZP_ERRRD(TRUE,"00", Var_ControlData, Var_Result);
                                  (* Reads error information *)
END_IF;
IF(Var_Result[0]=TRUE)THEN
                                  (* Execution finished *)
    IF(Var_Result[1]=FALSE)THEN(* Normal completion *)
          MOVP(TRUE,Var_ControlData[4],Var_ErrorInfo);
                                  (* Stores error information*)
    END IF;
    IF(Var_Result[1]=TRUE)THEN (* Error completion *)
          MOVP(TRUE,Var_ControlData[1],Var_ErrorCode);
                                  (* Stores error code *)
    END IF;
    RST(TRUE,Var_Flag_Exe);
                                  (* Turns execution flag OFF *)
END_IF;
```

5-240

5.4.32 UINI instruction

Z_UINI

| | | | | | *1 | CC IE C Eth | |
|-----------------|---------------------------------|--|--|--------------------|--|------------------------|-------|
| Z(P)_UINI | | | | | P: Executing cond | dition : _ | |
| | Liadder/FBD | ENO:= Z | ST UINI (EN, Un*, s, d); | | indicate instructions. Z_UINI | es any of the followin | g |
| Input argument | Un*: Sta (00 thr s: Va | ee digits) riable that stores o | digits when expressing | g the I/O number i | :Array of ANY16 [09] | | |
| Output argument | d: Va d[1 Sett |] also turns ON at ting Internal a ^{*1} Bit | N upon completion of th the time of error compl device Word | letion. J∭∖∭ | :Bit :Array of bit [01] U:\G Vord | Zn Constant | Other |
| | | | 0 | | – – e registers per program c | | |

Grant Function

Ethernet: This instruction reinitializes the Ethernet module.

CC-Link IE Controller Network: For Universal model QCPU, this instruction sets the station number of the CC-Link IE Controller Network module on normal station (host station).

5

NULE DEDICATED

Control Data

(1) Ethernet

| Device | Item | Setting data | Setting range | Setting side |
|----------------|----------------------------|---|------------------------------|--------------|
| s [0] | System area | - | _ | _ |
| ©[1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| ©[2] | Modification specification | [When updating the address information of external devices which are held by the Ethernet module] • Specify '0H'.*1 [When modifying the host station IP address, operation settings, transmission speed, communication mode] • Specify the parameter to be modified. However, Modification specification of transmission speed, communication mode cannot be executed simultaneously with that of host station IP address, operation settings. If executed, only modification specification of host station IP address and operation settings will be set. b15 b12b11 o b2 b1 0] ① Modification specification of host station IP address Specify whether the host station IP address is modified or not. (To modify the IP address, specify the address in ④ [3] and ④ [4].) 0: Not changed 1: Changed ② Modification specification of operation setting Specify whether the operation setting is modified or not. (To modify the operation setting, specify the operation setting in ④ [5].) 0: Not changed 1: Changed ③ ③ Modification specification of transmission speed, communication mode. O: not changed 1: Changed ③ Modification specification of transmission speed, communication mode. O: not changed 1: automatic negotiation 2: 100Mbps/full-duplex communi | 0н to 5000н | User |
| ঙ [3] ঙ [4] | Host station IP address | Specify the IP address of the host station. | 00000001н to FFFFFFFEн | User |

| Device | Item | Setting data | Setting range | Setting side |
|------------------------|-------------------|--|---------------|--------------|
| ৩ [5] | Operation setting | b15 to b9 b8 b7 b6 b5 b4 b3 b2 b1 b0 ① Communication data code setting 0 ① 〕 <t< td=""><td>0 or 1</td><td>User</td></t<> | 0 or 1 | User |
| \$ [6] to \$ [9] | - | Specify 0. | 0 | User |

*1 : The Ethernet module enables data exchange to restart by clearing the address information retained in the module and by performing re-initial processing.

(The Initial normal completion signal (X19) is on.)

(2) CC-link IE Controller Network

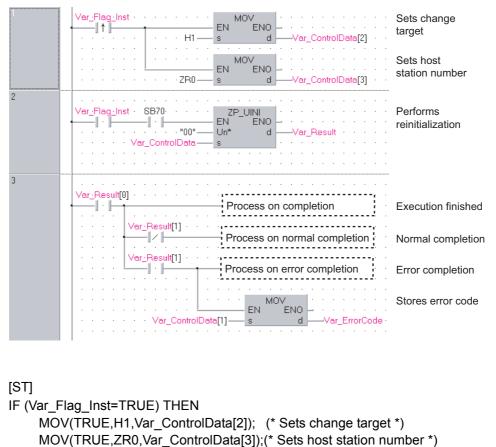
| Device | Item | Setting data | Setting range | Setting side |
|----------------------|----------------------------|---|---------------|--------------|
| <u></u> (0] | - | Specify 0. | 0 | User |
| ঙ[1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | - | System |
| ি [2] | Modification specification | Specify the change target 0001н: With station number setting | 0001н | User |
| ঙ[3] | Host station No. | Specify the station number of the host station. | 1 to 120 | User |
| \$[4] to \$[9] | - | Specify 0. | 0 | User |

The UINI instruction can be executed only once. The UINI instruction cannot be executed again after determination of station number. (It caused an error completion.)

However, in the case of the UINI instruction with the error completion, execute the UINI instruction again after taking corrective action.

Program Example

The following program sets the station number 2. The following is an example for Ethernet. [Structured ladder/FBD]



END IF;

```
IF((Var_Flag_Exe=TRUE) AND (SB70=TRUE))THEN

ZP_UINI(TRUE,"00",Var_ControlData,Var_Result);

(* Performs reinitialization *)

END_IF;

IF(Var_Result[0]=TRUE)THEN

(* Execution finished *)

IF(Var_Result[1]=FALSE)THEN

(* Normal completion *)
```

(* Process on normal completion *)

ELSE

(* Process on error completion *)

(* Process on error completion *)

MOV(TRUE, Var_ControlData[1], Var_ErrorCode);(* Stores error code *) END_IF;

(* Error completion *)

END_IF;

ZP_MRECV

5.4.33 MRECV instruction

| | | | | | | Ether |
|------------------------------|--|--|--|--|------------------|---|
| ZP_MRECV | , | | | | | Executing condition : 于 |
| Structured EN Un* s | d ladder/F MRECV ENO d1 d2 | | 2:= ZP_MF | ST IECV (EN, Un*, s, d1, d2); | Z | indicates the following instruction. |
| Input argument | EN: Un*: | Executing of Start I/O nu (00 to FE: H three digits | imber of t Higher two | he module digits when expressing the I/O n | :8 | Bit String |
| Output argument | s: ENO: d1: d2: | Variable the Execution of Start numb content of t Variable the | at stores o result er of the h the receiv at turns O | control data nost station's device that stores the ed e-mail (header + attached file N upon completion of the instruc at the time of error completion. | :E he :A) | Array of ANY16 [015] Bit ANY16 Array of bit [01] |
| | | Setting | Internal | device | J | |
| | | data *1 | Bit | Word R, ZR Bit | Wor | UIIIAGIII Zn Constant Othe |
| | | data ^{*1} | | Word Bit | | U∭\G∭ Zn Constant Othe d |

Grant Function

This instruction reads received e-mail.

ILE DEDICATED

ZP_MRECV

E Control Data

| Device | lte | em | Setting data | Setting range | Setting side |
|----------------------|---------------------------------|---------------------------------|--|--|--------------|
| © [0] | Execution/Error completion type | | b15 to b10 b9 b8 b7 to b0 0 2 0 ① 0 0 ① Error completion type (bit 7) Specify the clock data setup status at the time of error completion. 0: Clock data at the time of error completion is not set in the area starting from (§ [11]). 1: Clock data at the time of error completion is set in the area starting from (§ [11]). (2) Execution type (bit 9) *1 Specify whether to inquire about existence of mails in the server after reading received mails. 0: Not requested (not read) 1: Requested (read) | 0000н, 0080н, 0200н, 0280н | User |
| ঙ[1] | Completion status | | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| ঙ [2] | E-mail No. to be read | | Specify the number of a mail to be read when multiple mails are received. 0 : First mail 1 or more : Specified mail | 0 or more | User |
| ⑤ [3] to ⑤ [8] | System area | | _ | - | - |
| (٤) | For instruct Receive data | | Specify the data length (header + attached file) of the mail that can be stored in (1) to (1) +n. (Header: 1 to 373, attached file: 1 to 6144) 0 : Adjust data length to that of the received mail. 1 to 6517 : The number of data that can be stored in ((1) to (1) + n) | 0 to 6517 (word) * Includes the header length | User |
| | At instruction completion | At instruction completion | Data length (header + attached file) of the mail stored in ⓓ to ⓓ + n is stored. 1 to 6517: The number of receive data stored in (ⓓ to ⓓ + n) | explained below. | System |
| s়[10] | Header length | For instruction execution | Specify the header data length of the mail that can be stored in (a) to (a) + n. 0 : Adjust header data length to that of the received mail. 1 to 373 : The number of data that can be stored in ((a) to (a) + n) | 0 to 373 (word) | User |
| | | At instruction completion | Header data length of the mail stored in (d) to (d) + n is stored. 1 to 373: Number of receive data stored in ((d) to (d) + n) | | System |
| s [11] | Clock set flag | | Valid/invalid status of the data in the area starting from (§) [12] is stored. 0: Invalid 1: Valid | 0,1 | System |

| Device | Item | Setting data | Setting range | Setting side |
|------------------------|---|---|---------------|--------------|
| \$[12] to \$[15] | Clock data (set only when errors occur) | Clock data at the time of error completion are stored in BCD format. b15 to b8 b7 to b0 (s) [12] Month (01H to 12H) Year (00H to 99H) Last two digits b0 (s) [13] Hour (00H to 23H) Day (01H to 31H) b1 (s) [14] Second (00H to 59H) Minute (00H to 59H) b1 (s) [15] Year (00H to 99H) First two digits Day of week (00H to 06H) 00H (Sun.) to 06H (Sat.) | - | System |

| Device | Item | Setting data | Setting range | Setting side |
|--------|--------------|---|---------------|--------------|
| d1 +0 | | | | |
| to | Receive data | Content (header + attached file) of the received mail are stored. | - | System |
| d1 +n | | | | |

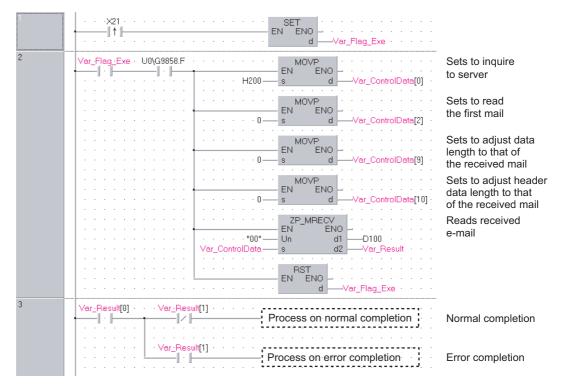
*1: The following table shows the processing that depends on the selection of the execution type after executing the MRECV instruction.

| Setting option | Processing | Advantage | Disadvantage |
|--------------------------|---|---|--|
| No request (not read) | Only e-mail read processing from the mail server is performed. Inquiry (reading) for the information of received mails remaining in the mail server is performed after the time set in the GX Works2 parameter has elapsed. | Unnecessary read processing is not performed when the mail server has no mail. | Even if mails remain in the mail server, they cannot be read immediately. Mails tend to be accumulated in the mail server. |
| Request (read) | E-mail read processing from the mail server is performed. After the execution of the MRECV instruction, inquiry (read) processing for information on the received mails remaining in the mail server is performed. (Inquiry for receiving of a mail is made immediately.) | Received mails stored in the mail server can be read in series. | Inquiries to the mail server are made more often. Internal processing of the module increases, which affects other internal processing to a certain degree. |

Program Example

The following program performs the e-mail receiving process by the receive instruction (X21). (The I/O signals of the Ethernet module are X/Y00 to X/Y1F)

[Structured ladder/FBD]



```
[ST]
IF (X21=TRUE) THEN
    SET(TRUE,Var_Flag_Exe);
END_IF;
IF((Var_Flag_Exe=TRUE) AND (U0\G9858.F=TRUE))THEN
    MOVP(TRUE,H200,Var ControlData[0]);
                (* Sets to inquire to server *)
    MOVP(TRUE,0,Var_ControlData[2]);
                (* Sets to read the first mail *)
    MOVP(TRUE,0,Var_ControlData[9]);
                (* Sets to adjust data length to that of the received mail *)
    MOVP(TRUE,0,Var_ControlData[10]);
                (* Sets to adjust header data length to that of the received mail *)
    ZP_MRECV(TRUE,"00",Var_ControlData,D100,Var_Result);
                (* Reads received e-mail *)
    RST(TRUE,Var_Flag_Exe);
END IF;
IF(Var_Result[0]=TRUE)THEN
                                        (* Execution finished *)
    IF(Var_Result[1]=FALSE)THEN
                                        (* Normal completion *)
              Process on normal completion *)
                                        (* Error completion *)
    ELSE
              Process on error completion *)
    END IF;
END_IF;
```

```
DULE DEDICATED
```

5.4.34 MSEND instruction

ZP_MSEND

Ether

| ZP_MSEND | | | | Executing condition : |
|-------------------------------------|----------------------------------|--|--|---------------------------------------|
| Structured EN Un* s1 s2 | I ladder/FB MSEND ENO d | | ST <u>2P_MSEND</u> (EN, Un*, s1, s2, d); | ZP_MSEND |
| Input argument | EN: Un*: s1: s2: | three digits) Variable that sto Start number of | | :Array of ANY16 [015] the :ANY16 |
| Output argument | ENO: d: | Execution result Variable that tur d[1] also turns C | ns ON upon completion of the instruction of the instruction. | |
| | | Setting Inte data ^{*1} Bit | R, ZR | JⅢ\Ⅲ Word UⅢ\GⅢ Zn Constant Others |
| | | (s1) – | 0 | _ |
| | | s2 - | 0 | |
| | | | | |

Grant Function

This instruction sends an e-mail.

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|------------------------|--|---|-------------------------|--------------|
| ال | Execution/Error completion type Send data format | b15 to b12 to b8 b7 to b0 0 2 1 0 1 Error completion type (bit 7) Specify the clock data setup status at the time of error completion. 0: Clock data at the time of error completion is not set in the area starting from (a) [11]. 1: Clock data at the time of error completion is set in the area starting from (a) [11]. 1: Clock data format (bit 12 to bit 8) Specify the data format of the send data. (Sending the data as an attached file) 1 1 1 0 0 0 1 0 0 1 0 0 0 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 (Sending the data as a text) Encat | (See the left column.) | User |
| s1[1] | Completion status | 0 : Normal completion Other than 0 : Error completion (error code) | - | System |
| st [2] | Transmission destination No. | Specify the external device to which e-mails are to be sent by the setting number on [Send mail address setting] of GX Works2. 1 to 16: Setting number of the external device | 1 to 16 | User |
| ৱা [3] to ৱা [8] | System area | - | - | - |
| ্রা [9] | Send data length | Specify the data length ((subject + attached file) or (subject + text)) of the mail stored in @ to @ + n. ① Sending the data as an attached file (subject: 0 to 373, attached file: 1 to 6144) 1 to 6517: Data length (word) of a mail ② Sending the data as a text (subject: 0 to 373, text: 1 to 960) 1 to 1333: Data length (word) of a mail | 1 to 6517, 1 to 1333 | User |
| গ্রা [10] | Subject length | Specify the subject data length of the mail stored in ② to ③ + n. 0 to 373: Data length (word) of subject | 0 to 373 | User |

| Device | Item | Setting data | Setting range | Setting side |
|--------------------------|---|---|---------------|--------------|
| s1 [11] | Clock set flag | Valid/invalid status of the data in the area starting from (a) [12] is stored. 0: Invalid 1: Valid | _ | System |
| ৰণ [12] to ৰণ [15] | Clock data (set only when errors occur) | Clock data at the time of error completion are stored in BCD format. b15 to b8 b7 to b0 (s) [12] Month (01H to 12H) Year (00H to 99H) Last two digits (s) [13] Hour (00H to 23H) Day (01H to 31H) (s) [14] Second (00H to 59H) Minute (00H to 59H) (s) [15] Year (00H to 99H) First two digits Day of week (00H to 06H) (0) (Sun.) to 06H (Sat.) 00v (Sun.) to 06H (Sat.) | _ | System |

(1) Send data

| Device | Item | Setting data | Setting range | Setting side |
|----------------------|-----------|--|---------------|--------------|
| €2 +0 to €2 +n | Send data | Specify the content of ((subject + attached file) or (Subject + text)) of a mail to be sent. | _ | User |

Program Example

The following program performs e-mail sending process by the send instruction (X20). (The I/O signals of the Ethernet module are X/Y00 to X/Y1F)

(1) Sending the data as an attached file

| | MOVP EN s | Sets ASCII as send data format |
|----------|---|--------------------------------------|
| | EN ENO s d | Sets transmission destination number |
| | MOVP EN ENO s d Var ControlData[9] | Sets send data |
| | | Sets subject length |
| | State | Sets subject |
| | EN ENO | Sets file to be attached |
| | | • |
| | | • |
| | · · · · · · · · · · · · · · · · · · · | Sends e-mail |
| | EN ENO Var_ControlData s1 S2 | • |
| 2 Var_Re | sult[0] Ver_Result[1] Process on normal completion | Normal completion |
| | Var_Result[1] | Error completion |

```
[ST]
IF(X20=TRUE)THEN
    MOVP(TRUE,H800,Var_ControlData[0]);
                             (* Sets ASCII as send data format *)
    MOVP(TRUE,1,Var_ControlData[2]);
                             (* Sets transmission destination number *)
    MOVP(TRUE,10,Var_ControlData[9]);
                             (* Sets send data length *)
    MOVP(TRUE,7,Var_ControlData[10]);
                             (* Sets subject length *)
    Int_Msg[0] := H6574;
                             (* te *)
                             (* st *)
    Int_Msg[1] := H7473;
    Int_Msg[2] := H616d;
                             (* ma *)
    Int_Msg[3] := H6c69;
                             (* il *)
    Int_Msg[4] := H6d20;
                             (* m *)
    Int Msg[5] := H6573;
                             (* se *)
    Int_Msg[6] := H646e;
                             (* nd *)
                             (* Sets subject *)
    MOVP(TRUE,H1234,Int_Msg[7]);
                             (* Sets file to be attached *)
    MOVP(TRUE,H5678,Int_Msg[8]);
    MOVP(TRUE,H9ABC,Int_Msg[9]);
    ZP_MSEND(TRUE,"U0", Var_ControlData, Int_Msg[0], Var_Result);
                             (* Sends e-mail *)
END IF;
IF(Var Result[0]=TRUE)THEN
                                   (* Execution finished *)
    IF(Var_Result[1]=FALSE)THEN(* Normal completion *)
                Process on normal completion *)
    ELSE
                                   (* Error completion *)
              Process on error completion *)
    END IF;
```

END_IF;

(2) Sending the data as a text [Structured ladder/FBD]

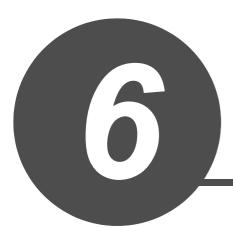
| - | | Sets text as send data format |
|---|--|--------------------------------------|
| | MOVP EN ENO s dVar_ControlData[2] | Sets transmission destination number |
| | MOVP EN ENO s dVar_ControlDeta[9] | Sets send data length |
| | MOVP EN ENO s d Var ControlData[10] | Sets subject length |
| | SMOVP EN ENO "testmail msend" s d | Sets subject |
| | | Sets text |
| | "Error Machine1 001"s D207 ZP_MSEND EN ENO | Sends e-mail |
| | Var_ControlData S1 | |
| 2 | Var_Result[0] Var_Result[1] Process on normal completion Var_Result[1] Var_Result[1] Process on error completion | Normal completion |

```
[ST]
IF(X20=TRUE)THEN
    MOVP(TRUE,H1000,Var_ControlData[0]);
                 (* Sets text as send data format *)
    MOVP(TRUE,1,Var ControlData[2]);
                 (* Sets transmission destination number *)
    MOVP(TRUE,16,Var_ControlData[9]);
                 (* Sets send data length *)
    MOVP(TRUE,7,Var_ControlData[10]);
                 (* Sets subject length *)
                             (* te *)
    Int_Msg[0] := H6574;
    Int_Msg[1] := H7473;
                             (* st *)
    Int_Msg[2] := H616d;
                             (* ma *)
    Int_Msg[3] := H6c69;
                             (* il *)
    Int_Msg[4] := H6d20;
                             (* m *)
    Int Msg[5] := H6573;
                             (* se *)
    Int_Msg[6] := H646e;
                             (* nd *)
                             (* Sets subject *)
    Int_Msg[7] := H7274;
                             (* Er *)
    Int_Msg[8] := H6f72;
                             (* ro *)
    Int_Msg[9] := H2072;
                             (* r *)
    Int_Msg[10] := H614d;
                             (* Ma *)
    Int_Msg[11] := H6863;
                             (* ch *)
    Int_Msg[12] := H6e69;
                             (* in *)
    Int_Msg[13] := H3165;
                             (* e1 *)
    Int Msg[14] := H3020;
                             (* 0 *)
    Int_Msg[15] := H3130;
                             (* 01 *)
                             (* Sets text *)
    ZP_MSEND(TRUE,"U0",Var_ControlData,Int_Msg[0],Var_Result);
                             (* Sends e-mail *)
END IF;
IF(Var_Result[0]=TRUE)THEN
                                   (* Execution finished *)
    IF(Var_Result[1]=FALSE)THEN(* Normal completion *)
              Process on normal completion *)
    ELSE
                                   (* Error completion *)
                                      Process on error completion *)
    END IF;
END_IF;
```

5-255

MEMO

| | |
|------|--|
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |



PID CONTROL INSTRUCTION

| 6.1 | PID Control Instruction (Inexact Differential) | 6-2 |
|-----|--|------|
| 6.2 | PID Control Instruction (Exact Differential) | 6-16 |

1

6.1 PID Control Instruction (Inexact Differential)

6.1.1 PIDINIT instruction

S_PIDINIT

| SP_PIDINIT | | P: Executing condition : |
|-----------------------------------|--|---|
| Structured I | | indicates any of the following instructions. S_PIDINIT SP_PIDINIT s); |
| Input argument Output argument | EN: Executing condition s: Start number of the device that stores P ENO: Execution result Setting data Internal device R, Z Sit Word State O | :Bit |

Function

This instruction enables PID control by registering the PID control data for the number of loops to be used to the CPU module in batch.

(1) PID control data

| | | | Setting | range | 0.000 | Processing when the | |
|----------------|--|---|--|--|-----------------|---|--|
| Device | Data item | Description | With PID limits | Without PID limits | Setting side | setting data are outside the setting range | |
| Common sett | ing data (device: (s) + | 0 to (s) +1) | | | | | |
| s +0 | Number of loops | Set the number of loops for PID operation. | 1 tc | 32 | User | | |
| ঙি +1 | Number of loops in one scan | Set the number of loops for PID operation in one scan if multiple loops have reached the sampling cycle time. | 1 tc | 9 32 | User | An error occurs and the PID operation for all loops is not performed. | |
| Setting data f | or No. 1 loop (device: | s +2 to s +15) | | | | | |
| s) +2 | Operational expression selection | Select the PID operational expression. *1 | 0: Forward operation 1: Reverse operation | 0: Forward operation 1: Reverse operation | User | | |
| s) +3 | Sampling cycle (Ts) | Set the PID operation cycle. | 1 to 6000 (unit: 10ms) | 1 to 6000 (unit: 10ms) | User | | |
| s) +4 | Proportional constant (KP) | Proportional gain of PID operation | 1 to 10000 (unit: 0.01) | 1 to 10000 (unit: 0.01) | User | An error occurs and the PID operation for the corresponding loop is not performed. | |
|) +5 | Integral constant (Tı) | Constant that expresses the magnitude of the integral action (I action) effect. Increasing the integral constant slows down the manipulated value change. | 1 to 32767 (unit: 100ms) If setting value > 30000 Tı = Infinite (∞) | 1 to 32767 (unit: 100ms) If setting value > 30000 Tı = Infinite (∞) | User | | |
| © +6 | Derivative constant (Tɒ) | Constant that expresses the magnitude of the derivative action (D action) effect. Increasing the derivative constant causes a significant change in the manipulated value even with a slight change of the control target. | 0 to 30000 (unit: 10ms) | 0 to 30000 (unit: 10ms) | User | | |
| s) +7 | Filter coefficient (α) | Set the degree of filtering to be applied to the process value. The filtering effect decreases as the value gets closer to 0. | 0 to 100 | 0 to 100 | User | | |

*1 : For the PID operational expressions to be set for Operational expression selection, refer to MELSEC-Q/L/ QnA Programming Manual (PID Control Instructions).

| | | | Setting | g range | Setting | Processing when the | |
|-------------|------------------------------|---|-----------------|-----------------------|---------|---|--|
| Device | Data item | Description | With PID limits | Without PID limits | side | setting data are outside the setting range | |
| s +8 | MV lower limit (MVLL) | Set the lower limit for the manipulated value (MV) calculated in PID operation in automatic mode. If the MV is less than the set lower limit value (MVLL), the value is clipped to the MVLL. | -50 to 2050 | -32768 to 32767 | User | In the case of "With PID limits", the PID operation is performed after values are replaced as follows: • If the MVLL or MVHL is less than -50, the value is clipped to -50. • If the MVLL or MVHL is greater than 2050, the value is clipped to 2050. | |
| ্ড +9 | MV upper limit (MVHL) | Set the upper limit for the manipulated value calculated in PID operation in automatic mode. If the MV is greater than the set upper limit value (MVHL), the value is clipped to the MVHL. | -50 to 2050 | -32768 to 32767 | User | | |
| ঙ +10 | MV change rate limit (| Set the variation limit between the previous MV and the present MV. When the MV variation is greater than the limit value, bit 1 (b1) of the alarm device is set to '1'. MV variation is not limited. (Even if the MV variation exceeds the limit value, the actual MV variation is used as it is for calculating the MV.) | 0 to 2000 | -32768 to 32767 | User | In the case of "With PID limits", the PID operation is performed after values are replaced as follows: If the △ MVL value is less than 0, the value is clipped to 0. If the △ MVL value is greater than 2000, the value is clipped to 2000. | |
| s +11 | PV change rate limit (| Set the variation limit between the previous PV and the present PV. When the PV variation is greater than the limit value, bit 0 (b0) of the alarm device is set to '1'. PV variation is not limited. (Even if the PV variation exceeds the limit value, the actual PV variation is used as it is for performing the PID operation.) | 0 to 2000 | -32768 to 32767 | User | In the case of "With PID limits", the PID operation is performed after values are replaced as follows: If the △ PVL value is less than 0, the value is clipped to 0. If the △ PVL value is greater than 2000, the value is clipped to 2000. | |

| | | | Setting | range | Setting | Processing when the |
|---------------|-------------------------|---|--|---|---------|---|
| Device | Data item | item Description | With PID limits | Without PID limits | side | setting data are outside the setting range |
| s +12 | (Fixed value) | - | 0 | 0 | User | - |
| ঙ +13 | Derivative gain (Kɒ) | Set a duration (delay in action) for derivative action. As the setting value increases, the duration becomes smaller and action becomes closer to exact differential. Ideal value KD = 8.00 | 0 to 32767 (unit: 0.01) If setting value > 30000 KD = Infinite (\sigma) | 0 to 32767 (unit: 0.01) If setting value > 30000 KD = Infinite (\comega) | User | An error occurs and the PID operation for the corresponding loop is not performed. |
| s +14 | (Fixed value) | _ | 0 | 0 | User | _ |
| <u></u> জ +15 | (Fixed value) | - | 0 | 0 | User | - |

Setting data for No. 2 loop (device: (s) +16 to (s) +29)

| | Operational | |
|----------------|--------------------|---|
| s +16 | expression | |
| | selection | |
| 0 147 | Sampling cycle | |
| s +17 | (Ts) | |
| | Proportional | |
| s +18 | constant | |
| | (KP) | |
| | Integral | |
| s +19 | constant | |
| | (Tı) | |
| | Derivative | |
| s +20 | constant | |
| | (TD) | |
| (s) +21 | Filter coefficient | |
| (5) +2 I | (α) | The same as Catting data for No. 4 lash |
| (s) +22 | MV lower limit | The same as Setting data for No. 1 loop |
| 3 722 | (MVLL) | |
| 0,100 | MV upper limit | |
| s +23 | (MVHL) | |
| | MV change rate | |
| s +24 | limit | |
| | ($	riangle$ MVL) | |
| | PV change rate | |
| s +25 | limit | |
| | ($	riangle$ PVL) | |
| s +26 | (Fixed value) | |
| | Derivative gain | |
| s +27 | (KD) | |
| s +28 | (Fixed value) | |
| s) +29 | (Fixed value) | |

| | | | Setting | range | | Processing when the |
|------------------|--------------------|--------------------------|-----------------|-----------------------|-----------------|--|
| Device | Data item | Description | With PID limits | Without PID limits | Setting side | setting data are outside the setting range |
| Setting data for | or No. n loop | | | | | |
| | Operational | | | | | |
| s +(m+0) | expression | | | | | |
| | selection | | | | | |
| s +(m+1) | Sampling cycle | | | | | |
| (m+1) | (Ts) | | | | | |
| | Proportional | | | | | |
| s +(m+2) | constant | | | | | |
| | (KP) | | | | | |
| | Integral | | | | | |
| s +(m+3) | constant | | | | | |
| | (Tı) | | | | | |
| | Derivative | | | | | |
| জ +(m+4) | constant | | | | | |
| | (TD) | | | | | |
| | Filter coefficient | | | | | |
| জ +(m+5) | (α) | The same as Setting data | for No. 1 loop | | | |
| s +(m+6) | MV lower limit | The same as Setting data | | | | |
| (III+0) | (MVLL) | | | | | |
| (m + 7) | MV upper limit | | | | | |
| জ +(m+7) | (MVHL) | | | | | |
| | MV change rate | | | | | |
| s +(m+8) | limit | | | | | |
| | (🛆 MVL) | | | | | |
| | PV change rate | 1 | | | | |
| s +(m+9) | limit | | | | | |
| | (🛆 PVL) | | | | | |
| s +(m+10) | (Fixed value) | 1 | | | | |
| | Derivative gain | • | | | | |
| জ +(m+11) | (KD) | | | | | |
| s +(m+12) | (Fixed value) | | | | | |
| s +(m+13) | (Fixed value) | | | | | |

m=(n-1)×14+2 n: number of loops

Precautions

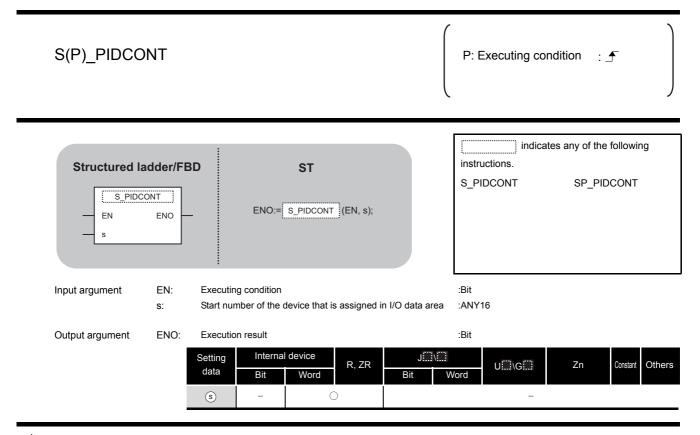
The following table shows the CPU modules applicable to the PID control instructions (inexact differential) and the PID control instructions (exact differential).

| | CPU module model | Inexact differential | Exact differential |
|------------------------|---|-------------------------|--------------------|
| Basic model QCPU | The first five digits of the serial number are '04122' or higher | 0 | 0 |
| High Performance model | The first five digits of the serial number are '05031' or lower. | × | 0 |
| QCPU | The first five digits of the serial number are '05032' or higher. | 0 | 0 |
| Universal model QCPU | | 0 | 0 |
| LCPU | | 0 | 0 |

 \bigcirc : Applicable, imes: Not applicable

6.1.2 PIDCONT instruction

S_PIDCONT



Grant Function

- (1) This instruction measures sampling cycle and performs PID operation at instruction execution.
- (2) This instruction performs PID operation based on the set value (SV) and process value (PV) in the I/O data area set to the device number specified by s or later, and stores the operation result to the automatic manipulated value (MV) area in the I/O data area.
- (3) PID operation is performed in response to the first execution of the PIDCONT instruction after the set sampling cycle time has elapsed.

(1) I/O data

| | | | Setting | range | Setting | Processing when the | |
|-----------------|--------------------------------|----------------------------|---|---------------|---------|-------------------------------|--|
| Device | Data name | Description | With PID | Without PID | side | setting data are outside | |
| | | | limits limits | | Side | the setting range | |
| | | | 0 : PID opera | tion for the | | | |
| | | | number of | f loops to be | | | |
| | | | used is ba | tch-processed | | | |
| | Initial processing flag | Processing method at | in one sca | an. | User | | |
| s +0 | miliar processing hag | the start of PID operation | Other than 0: F | PID operation | USEI | _ | |
| | | | for the number of loops to be used is processed | | | | |
| | | | | | | | |
| | | | in several | scans. | | | |
| s +1 | DID control | work area | | | | | |
| to | PID control | | - | | _ | | |
| s +9 | (reserved by | the system) | | | | | |
| I/O data area f | or No. 1 loop (device: (s) +10 | to (s) +27) | | | | | |
| | | | | | | In the case of "With PID | |
| | | | | | | limits", the PID operation is | |
| | | | | | | performed after values are | |
| | | PID control target | | -32768 to | | replaced as follows: | |

| ঙ +10 | Set value | SV | PID control target value | 0 to 2000 | -32768 to 32767 | User | replaced as follows: If SV is less than 0, the value is clipped to 0. If SV is greater than 2000, the value is clipped to 2000. |
|--------|-------------------------------------|-------|--|-------------|--------------------|--------|--|
| ঙ +11 | Process value | PV | Feedback data from the control target to the A/D conversion module | -50 to 2050 | -32768 to 32767 | User | In the case of "With PID limits", the PID operation is performed after values are replaced as follows: • If PV is less than -50, the value is clipped to -50. • If PV is greater than 2050, the value is clipped to 2050. |
| ঙ +12 | Automatic manipulated value | MV | Manipulated value obtained by PID operation The value is output from the D/A conversion module to the control target. | -50 to 2050 | -32768 to 32767 | System | _ |
| s) +13 | Process value after filtering | PVf | Process value obtained by calculation using operational expression. *1 | -50 to 2050 | -32768 to 32767 | System | - |
| ঙ +14 | Manual manipulated value | MVman | Store the data output from the D/A conversion module in manual operation. | -50 to 2050 | -32768 to 32767 | User | In the case of "With PID limits", the PID operation is performed after values are replaced as follows: • If MVMAN is less than -50, the value is clipped to -50. • If MVMAN is greater than 2050, the value is clipped to 2050. |

*1 : For Process value after filtering (PVf), the value calculated based on the process value of input data are stored.

| | | | | Setting | g range | Sotting | Processing when the |
|----------------------------|-----------------------------------|--------------|--|--------------------|--|-----------------|--|
| Device | Data nam | ne | Description | With PID limits | Without PID limits | Setting side | setting data are outside the setting range |
| s) +15 | Manual/ automatic selection | MAN/ AUTO | Select whether the output to the D/A conversion module is a manual manipulated value or an automatic manipulated value. In manual operation, the automatic manipulated value remains unchanged. | value | tic manipulated manipulated | User | When other than 0 or 1 is selected, an error occurs and the operation for the corresponding loop is not performed. |
| s +16 | Alarm | ALARM | Used to determine if the change rate of the MV (manipulated value) and the PV (process value) is within or outside the limit value range. Once set, the alarm data are maintained until the user resets it. When the MV variation is outside the limit range, bit 1 (b1) is set to '1'. When the PV variation is outside the limit range, bit 0 (b0) is set to '1'. | is | f PV variation s outside the mit range, 1' is set. f MV variation s outside the mit range, 1' is set. | User System | _ |
| (\$) +17 to (\$) +32 | | | ol work area y the system) | | - | _ | - |

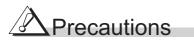
I/O data area for No. 2 loop (device: (s) +28 to (s) +45)

| I/O data area f | or No. 2 loop (de | evice: (s) +2 | 28 to (s) +45) | | | |
|----------------------|-------------------------------------|---------------|-------------------------------|----------------|---|---|
| s +33 | Set value | SV | | | | |
| s) +34 | Process value | PV | | | | |
| ঙ +35 | Automatic manipulated value | MV | | | | |
| s +36 | Process value after filtering | PVf | The same as I/O data area f | for No. 1 loop | | |
| s) +37 | Manual manipulated value | MVman | | | | |
| © +38 | Manual/ automatic selection | MAN/ AUTO | | | | |
| s +39 | Alarm | ALARM | | | | |
| s +40 to s +55 | | | ol work area y the system) | - | _ | _ |

6

| | | | | Setting | range | Setting | Processing when the |
|--|--|--------------|----------------------------------|-------------------|-----------------------|---------|--|
| Device | Data r | name | Description | With PID limits | Without PID limits | side | setting data are outside the setting range |
| I/O data area fo | or No. n loop | | | | | | |
| s +(m+0) | Set value | SV | | | | | |
| s +(m+1) | Process value | PV | | | | | |
| s +(m+2) | Automatic manipulated value | MV | | | | | |
| s +(m+3) | Process value after filtering | PVf | The same as I/O data an | ea for No. 1 loop | | | |
| s +(m+4) | Manual manipulated value | MVman | | | | | |
| জ +(m+5) | Manual/ automatic selection | MAN/ AUTO | | | | | |
| s +(m+6) | Alarm | ALARM | | | | | |
| (\$) +(m+7) to (\$) +(m+22) | | | trol work area by the system) | - | - | _ | - |

m=(n-1) \times 23+10 n: number of loops



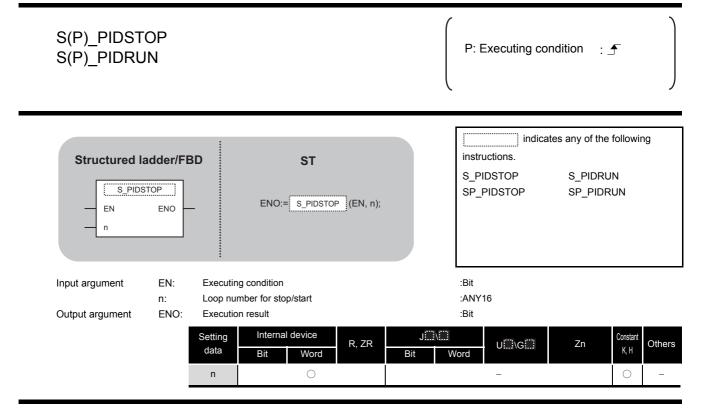
The following table shows the CPU modules applicable to the PID control instructions (inexact differential) and the PID control instructions (exact differential).

| | CPU module model | Inexact differential | Exact differential |
|------------------------|---|-------------------------|--------------------|
| Basic model QCPU | The first five digits of the serial number are '04122' or higher | 0 | 0 |
| High Performance model | The first five digits of the serial number are '05031' or lower. | × | 0 |
| QCPU | The first five digits of the serial number are '05032' or higher. | 0 | 0 |
| Universal model QCPU | | 0 | 0 |
| LCPU | | 0 | 0 |

 \bigcirc : Applicable, \times : Not applicable

6.1.3 PIDSTOP instruction and PIDRUN instruction

S_PIDSTOP, S_PIDRUN



Grant Function

- (1) S(P)_PIDSTOP This instruction stops the PID operation for the loop number specified by 'n'.
- (2) S(P) PIDRUN This instruction starts the PID operation for the loop number specified by 'n'.

Precautions

The following table shows the CPU modules applicable to the PID control instructions (inexact differential) and the PID control instructions (exact differential).

| | CPU module model | Inexact differential | Exact differential |
|------------------------|---|-------------------------|-----------------------|
| Basic model QCPU | The first five digits of the serial number are '04122' or higher | 0 | 0 |
| High Performance model | The first five digits of the serial number are '05031' or lower. | × | 0 |
| QCPU | The first five digits of the serial number are '05032' or higher. | 0 | 0 |
| Universal model QCPU | | 0 | 0 |
| LCPU | 0 | 0 | |

 \bigcirc : Applicable, \times : Not applicable

6

S_PIDSTOP, S_PIDRUN

6.1.4 PIDPRMW instruction

S_PIDPRMW

| S(P)_PIDPR | MW | | P: Executing condition : _ |
|-----------------|------|---|---|
| Structured I | | D ST ENO:= <u>S_PIDPRMW</u> (EN, n, s); | indicates any of the following instructions. S_PIDPRMW SP_PIDPRMW |
| Input argument | EN: | Executing condition | :Bit |
| | n: | Loop number to be changed | :ANY16 |
| | S: | Start number of the device that stores PID control data to be changed | :ANY16 |
| Output argument | ENO: | Execution result | :Bit |
| | | Setting data Internal device R, ZR Jiii Bit Word Bit V n O O V (s) - O V | Word U Zn Constant K, H Others - - - - - - - - - - |

☆ Function

This instruction changes the operation parameter of the loop number specified by 'n' to the PID control data stored in the devices starting from the device number specified by s.

(1) PID control data

| | | | Setting | range | Setting | Processing when the |
|---------------|--|---|--|--|---------|---|
| Device | Data item | Description | With PID limits | Without PID limits | side | setting data are outside the setting range |
| (s) +0 | Operational expression selection | Select the PID operational expression. *1 | 0: Forward operation 1: Reverse operation | 0: Forward operation 1: Reverse operation | User | An error occurs and the PID operation for the corresponding loop is not |
| (s) +1 | Sampling cycle (Ts) | Set the PID operation cycle. | 1 to 6000 (unit: 10ms) | 1 to 6000 (unit: 10ms) | User | performed. |

*1 : For the PID operational expressions set for Operational expression selection, refer to MELSEC-Q/L/QnA Programming Manual (PID Control Instructions).

| | | | Setting | g range | Cotting | Processing when the |
|--------|----------------------------------|--|--|--|-----------------|--|
| Device | Data item | Description | With PID limits | Without PID limits | Setting side | setting data are outside the setting range |
| ঙ +2 | Proportional constant (KP) | Proportional gain of PID operation | 1 to 10000 (unit: 0.01) | 1 to 10000 (unit: 0.01) | User | |
| (s) +3 | Integral constant (Tı) | Constant that expresses the magnitude of the integral action (I action) effect. Increasing the integral constant slows down the manipulated value change. | 1 to 32767 (unit: 100ms) If setting value > 30000 Ti = Infinite (∞) | 1 to 32767 (unit: 100ms) If setting value > 30000 Ti = Infinite (∞) | User | |
| ঙি +4 | Derivative constant (T⊳) | Constant that expresses the magnitude of the derivative action (D action) effect. Increasing the derivative constant causes significant changes in the manipulated value even with a slight change of the control target. | 0 to 30000 (unit: 10ms) | 0 to 30000 (unit: 10ms) | User | An error occurs and the PID operation for the corresponding loop is not performed. |
| s) +5 | Filter coefficient (α) | Set the degree of filtering to be applied to the process value. The filtering effect decreases as the value gets closer to 0. | 0 to 100 | 0 to 100 | User | |
| s) +6 | MV lower limit (MVLL) | Set the lower limit for the manipulated value (MV) calculated in PID operation in automatic mode. If the MV is less than the set lower limit value (MVLL), the value is clipped to the MVLL. | -50 to 2050 | -32768 to 32767 | User | In the case of "With PID limits", the PID operation is performed after values are replaced as follows: • If MVLL or MVHL value is less than -50, the value is clipped to -50. • If MVLL or MVHL value is greater than 2050, the value is clipped to 2050. |
| s) +7 | MV upper limit (MVHL) | Set the upper limit for the manipulated value calculated in PID operation in automatic mode. If the MV is greater than the set upper limit value (MVHL), the value is clipped to the MVHL. | -50 to 2050 | -32768 to 32767 | User | |

S_PIDPRMW

| | Data item | Description | Setting range | | Setting | Processing when the |
|-------------|--------------------------------|---|---|--|---------|---|
| Device | | | With PID limits | Without PID limits | side | setting data are outside the setting range |
| s +8 | MV change rate limit (| Set the variation limit between the previous MV and the present MV. When the MV variation is greater than the limit value, bit 1 (b1) of the alarm device is set to '1'. MV variation is not limited. (Even if the MV variation exceeds the limit value, the actual MV variation is used as it is for calculating the MV.) | 0 to 2000 | -32768 to 32767 | User | In the case of "With PID limits", the PID operation is performed after values are replaced as follows: If the △ MVL value is less than 0, the value is clipped to 0. If the △ MVL value is greater than 2000, the value is clipped to 2000. |
| ঙ +9 | PV change rate limit (| Set the variation limit between the previous PV and the present PV. When the PV variation is greater than the limit value, bit 0 (b0) of the alarm device is set to '1'. PV variation is not limited. (Even if the PV variation exceeds the limit value, the actual PV variation is used as it is for performing the PID operation.) | 0 to 2000 | -32768 to 32767 | User | In the case of "With PID limits", the PID operation is performed after values are replaced as follows: • If the △ PVL value is less than 0, the value is clipped to 0. • If the △ PVL value is greater than 2000, the value is clipped to 2000. |
| s +10 | (Fixed value) | _ | 0 | 0 | User | - |
| ঙ +11 | Derivative gain (KD) | Set a duration (delay in action) for derivative action. As the setting value increases, the duration becomes smaller and action becomes closer to exact differential. | 0 to 32767 (unit: 0.01) If setting value > 30000 KD = Infinite (\cord) | 0 to 32767 (unit: 0.01) If setting value > 30000 KD = Infinite (∞) | User | An error occurs and the PID operation for the corresponding loop is not performed. |
| () 110 | (Fixed value) | Ideal value K _D = 8.00 | 0 | 0 | User | |
| s +12 | (Fixed value) (Fixed value) | _ | 0 | 0 | User | |
| s +13 | | - | U | U | 0361 | - |

Precautions

The following table shows the CPU modules applicable to the PID control instructions (inexact differential) and the PID control instructions (exact differential).

| | CPU module model | Inexact differential | Exact differential |
|-----------------------------|---|-------------------------|-----------------------|
| Basic model QCPU | The first five digits of the serial number are '04122' or higher | 0 | 0 |
| High Performance model QCPU | The first five digits of the serial number are '05031' or lower. | × | 0 |
| | The first five digits of the serial number are '05032' or higher. | 0 | 0 |
| Universal model QCPU | | 0 | 0 |
| LCPU | | 0 | 0 |

 \bigcirc : Applicable, \times : Not applicable

6.2 PID Control Instruction (Exact Differential)

6.2.1 PIDINIT instruction

PIDINIT

| PIDINIT(P) | | P: Executing condition : |
|-----------------------------------|---|---|
| Structured I | | indicates any of the following instructions. PIDINIT PIDINITP |
| Input argument Output argument | EN: Executing condition s: Start number of the device that stores PID control ENO: Execution result Setting data Internal device Bit Word S - | Bit Word :Bit |

Function

This instruction enables PID control by registering the PID control data for the number of loops to be used to the CPU module in batch.

(1) PID control data

| | | | Setting range | | Sotting | Processing when the |
|----------------|--|--|---|--|-----------------|---|
| Device | Data item | Description | With PID limits | Without PID limits | Setting side | setting data are outside the setting range |
| Common sett | ing data (device: (s) + | 0 to ⊚ +1) | | | | |
| s +0 | Number of loops | Set the number of loops for PID operation. | | | User | |
| § +1 | Number of loops in one scan | Set the number of loops for PID operation in one scan if multiple loops have reached the sampling cycle time. | | | User | An error occurs and the PID operation for all loops is not performed. |
| Setting data f | or No. 1 loop (device: | s +2 to s +11) | | | | |
| (s) +2 | Operational expression selection | Select the PID operational expression. *1 | 0: Forward operation 1: Reverse operation | 0: Forward operation 1: Reverse operation | User | |
| s +3 | Sampling cycle (Ts) | Set the PID operation cycle. | 1 to 6000 (unit: 10ms) | 1 to 6000 (unit: 10ms) | User | |
| s) +4 | Proportional constant (KP) | Proportional gain of PID operation | 1 to 10000 (unit: 0.01) | 1 to 10000 (unit: 0.01) | User | An error occurs and the PID operation for the corresponding loop is not performed. |
| ٤) +5 | Integral constant (Tı) | Constant that expresses the magnitude of the integral action (I action) effect. Increasing the integral constant slows down the manipulated value change. | 1 to 32767 (unit: 100ms) If setting value > 30000 TI = Infinite (∞) | 1 to 32767 (unit: 100ms) If setting value > 30000 Ti = Infinite (∞) | User | |
| © +6 | Derivative constant (Tɒ) | Constant that expresses the magnitude of the derivative action (D action) effect. Increasing the derivative constant causes a significant changes in the manipulated value even with a slight change of the control target. | 0 to 30000 (unit: 10ms) | 0 to 30000 (unit: 10ms) | User | |
| ® +7 | Filter coefficient (α) | Set the degree of filtering to be applied to the process value. The filtering effect decreases as the value gets closer to 0. | 0 to 100 | 0 to 100 | User | |

*1 : For the PID operational expressions set for Operational expression selection, refer to MELSEC-Q/L/QnA Programming Manual (PID Control Instructions).

| Device | Data item | Description | Setting range | | Setting | Processing when the |
|-------------|------------------------------|---|-----------------|-----------------------|---------|--|
| | | | With PID limits | Without PID limits | side | setting data are outside the setting range |
| s +8 | MV lower limit (MVLL) | Set the lower limit for the manipulated value (MV) calculated in PID operation in automatic mode. If the MV is less than the set lower limit value (MVLL), the value is clipped to the MVLL. | -50 to 2050 | -32768 to 32767 | User | In the case of "With PID limits", the PID operation is performed after values are replaced as follows: • If MVLL or MVHL value is less than -50, the value is clipped to -50. • If MVLL or MVHL value is greater than 2050, the value is clipped to 2050. |
| (6) +9 | MV upper limit (MVHL) | Set the upper limit for the manipulated value calculated in PID operation in automatic mode. If the MV is greater than the set upper limit value (MVHL), the value is clipped to the MVHL. | -50 to 2050 | -32768 to 32767 | User | |
| s +10 | MV change rate limit (| Set the variation limit between the previous MV and the present MV. When the MV variation is greater than the limit value, bit 1 (b1) of the alarm device is set to '1'. MV variation is not limited. (Even if the MV variation exceeds the limit value, the actual MV variation is used as it is for calculating the MV.) | 0 to 2000 | -32768 to 32767 | User | In the case of "With PID limits", the PID operation is performed after values are replaced as follows: If the △ MVL value is less than 0, the value is clipped to 0. If the △ MVL value is greater than 2000, the value is clipped to 2000. |
| s) +11 | PV change rate limit (| Set the variation limit between the previous PV and the present PV. When the PV variation is greater than the limit value, bit 0 (b0) of the alarm device is set to '1'. PV variation is not limited. (Even if the PV variation exceeds the limit value, the actual PV variation is used as it is for performing the PID operation.) | 0 to 2000 | -32768 to 32767 | User | In the case of "With PID limits", the PID operation is performed after values are replaced as follows: If the △ PVL value is less than 0, the value is clipped to 0. If the △ PVL value is greater than 2000, the value is clipped to 2000. |

| Device | Data item | Description | Setting With PID limits | range Without PID limits | Setting side | Processing when the setting data are outside the setting range | | | |
|-----------------|------------------------------|----------------------------|----------------------------|--------------------------------|-----------------|--|--|--|--|
| Setting data fo | r No. 2 loop (device: | (s) +12 to (s) +21) | | | | | | | |
| | Operational | | | | | | | | |
| s +12 | expression | | | | | | | | |
| | selection | | | | | | | | |
| s +13 | Sampling cycle | | | | | | | | |
| | (Ts) | | | | | | | | |
| s +14 | Proportional constant | | | | | | | | |
| 3 +14 | (KP) | | | | | | | | |
| | Integral | | | | | | | | |
| s +15 | constant | | | | | | | | |
| U U | (Tı) | | | | | | | | |
| | Derivative | | | | | | | | |
| s +16 | constant | The same as Setting data f | or No. 1 loop | | | | | | |
| | (TD) | The same as bearing data i | | | | | | | |
| s +17 | Filter coefficient | | | | | | | | |
| | (α) MV lower limit | - | | | | | | | |
| s +18 | (MVLL) | | | | | | | | |
| | MV upper limit | - | | | | | | | |
| s +19 | (MVHL) | | | | | | | | |
| | MV change rate | | | | | | | | |
| s +20 | limit | | | | | | | | |
| U | ($	riangle$ MVL) | | | | | | | | |
| | PV change rate | | | | | | | | |
| s +21 | limit | | | | | | | | |
| | ($ ightarrow$ PVL) | | | | | | | | |
| Setting data fo | | 1 | | | | | | | |
| s +(m+0) | Operational expression | | | | | | | | |
| () +(III+0) | selection | | | | | | | | |
| | Sampling cycle | - | | | | | | | |
| s +(m+1) | (Ts) | | | | | | | | |
| | Proportional | | | | | | | | |
| s +(m+2) | constant | | | | | | | | |
| | (Kp) | | | | | | | | |
| | Integral | | | | | | | | |
| s +(m+3) | constant | | | | | | | | |
| | (Tı) Derivative | - | | | | | | | |
| s +(m+4) | constant | | | | | | | | |
| (III+4) | (TD) | The same as Setting data f | or No. 1 loop | | | | | | |
| | Filter coefficient | - | | | | | | | |
| s +(m+5) | (α) | | | | | | | | |
| () . (m · 0) | MV lower limit | 1 | | | | | | | |
| s +(m+6) | (MVLL) | | | | | | | | |
| s +(m+7) | MV upper limit | | | | | | | | |
| (III+7) | (MVHL) | ļ | | | | | | | |
| | MV change rate | | | | | | | | |
| s +(m+8) | limit | | | | | | | | |
| | (△ MVL) PV change rate | 4 | | | | | | | |
| | PV change rate | | | | | | | | |
| s +(m+9) | limit | | | | | | | | |

m=(n-1)×10+2 n: number of loops 6

INTROL JCTION

PIDINIT

Precautions

The following table shows the CPU modules applicable to the PID control instructions (inexact differential) and the PID control instructions (exact differential).

| | CPU module model | Inexact differential | Exact differential |
|------------------------|---|-------------------------|--------------------|
| Basic model QCPU | The first five digits of the serial number are '04122' or higher | 0 | 0 |
| High Performance model | The first five digits of the serial number are '05031' or lower. | × | 0 |
| QCPU | The first five digits of the serial number are '05032' or higher. | 0 | 0 |
| Universal model QCPU | | 0 | 0 |
| LCPU | | 0 | 0 |

 \bigcirc : Applicable, \times : Not applicable

6.2.2 PIDCONT instruction

PIDCONT

| PIDCONT(P | ') | P: Executing condition : |
|-----------------|--|---|
| Structured I | | indicates any of the following instructions. PIDCONT PIDCONTP |
| Input argument | EN: Executing condition s: Start number of the device that is assigned in I/ | :Bit /O data area :ANY16 |
| Output argument | Execution result Setting data Internal device Bit Word R, ZR (s) - O | Bit UIII\GIII Zn Constant Others Bit Word – |

☆ Function

- (1) This instruction measures sampling cycle and performs PID operation at instruction execution.
- (2) This instruction performs PID operation based on the set value (SV) and process value (PV) in the I/O data area set to the device number specified by (5) or later, and stores the operation result to the automatic manipulated value (MV) area in the I/O data area.
- (3) PID operation is performed in response to the first execution of the PIDCONT instruction after the set sampling cycle time has elapsed.

(1) I/O data

| | | | | Setting | range | Setting | Processing when the | |
|--------------------|-------------------------|----------------|---|---|--------------------|---------|---|--|
| Device | Data nar | ne Description | | With PID limits | Without PID limits | side | setting data are outside the setting range | |
| s +0 | Initial processing flag | | Processing method at the start of PID operation | PID operation for the number of loops to be used is batch-processed in one scan. Other than 0: PID operation for the number of loops to be used is processed in several scans. | | User | - | |
| s +1 to s +9 | | | ol work area y the system) | - | - | - | - | |
| I/O data area f | or No. 1 loop (de | vice: (s)+1 | 0 to (s) +27) | | | | | |
| (s) +10 | Set value | SV | PID control target value | 0 to 2000 | -32768 to 32767 | User | In the case of "With PID limits", the PID operation is performed after values are replaced as follows: • If SV is less than 0, the value is clipped to 0. • If SV is greater than 2000, the value is clipped to 2000. | |
| s) +11 | Process value | PV | Feedback data from the control target to the A/D conversion | -50 to 2050 | -32768 to 32767 | User | In the case of "With PID limits", the PID operation is performed after values are replaced as follows: • If PV is less than -50, the value is clipped to | |

| | | | module | | | | -50.If PV is greater than 2050, the value is clipped to 2050. |
|---------------|-------------------------------------|-----|--|-------------|--------------------|--------|--|
| s) +12 | Automatic manipulated value | MV | Manipulated value obtained by PID operation The value is output from the D/A conversion module to the control target. | -50 to 2050 | -32768 to 32767 | System | - |
| s) +13 | Process value after filtering | PVf | Process value obtained by calculation using operational expression. *1 | -50 to 2050 | -32768 to 32767 | System | - |

*1 : For process value after filtering (PVf), the value calculated based on the process value of input data are stored.

For the operational expression, refer to MELSEC-Q/L/QnA Programming Manual (PID Control Instructions).

| | | | | Setting | g range | 0 | Processing when the | |
|----------------------------|---|--------------|--|--|--|-----------------|--|--|
| Device | Data na | me | Description | With PID limits | Without PID limits | Setting side | setting data are outside the setting range | |
| \$ +14 | Manual manipulated value | MVman | Store the data output from the D/A conversion module in manual operation. | -50 to 2050 | -32768 to 32767 | User | In the case of "With PID limits", the PID operation is performed after values are replaced as follows: • If MVMAN is less than -50, the value is clipped to -50. • If MVMAN is greater than 2050, the value is clipped to 2050. | |
| \$ +15 | Manual/ automatic selection | MAN/ AUTO | Select whether the output to the D/A conversion module is a manual manipulated value or an automatic manipulated value. In manual operation, the automatic manipulated value remains unchanged. | 0: Automati value 1: Manual r value | c manipulated nanipulated | User | When other than 0 or 1 is selected, an error occurs and the operation for the corresponding loop is not performed. | |
| s) +16 | Alarm | ALARM | Used to determine if the change rate of the MV (manipulated value) and the PV (process value) is within or outside the limit value range. Once set, the alarm data are maintained until the user resets it. When the MV variation is outside the limit range, bit 1 (b1) is set to '1'. When the PV variation is outside the limit range, bit 0 (b0) is set to '1'. | | If PV variation is outside the limit range, '1' is set. If MV variation is outside the limit range, '1' is set. | User System | _ | |
| (\$) +17 to (\$) +27 | PID control work area (reserved by the system) | | | | - | - | - | |

6

PID CONTROL INSTRUCTION

PIDCONT

| Device | Data nai | me | Description | Setting With PID limits | range Without PID limits | Setting side | Processing when the setting data are outside the setting range | | |
|--|-------------------------------------|--------------|--------------------------|--|--------------------------------|--------------|--|--|--|
| I/O data area fo | or No. 2 loop (dev | /ice: (s) +2 | 8 to ⊚ +45) | linnits | linits | | the setting range | | |
| s +28 | Set value | SV | , (j), | | | | | | |
| <u>چ</u> (§) +29 | Process value | PV | | | | | | | |
| s) +30 | Automatic manipulated value | MV | | The same as I/O data area for No. 1 loop | | | | | |
| ঙ +31 | Process value after filtering | PVf | The same as I/O data are | | | | | | |
| s) +32 | Manual manipulated value | MVman | | | | | | | |
| s) +33 | Manual/ automatic selection | MAN/ AUTO | | | | | | | |
| s +34 | Alarm | ALARM | | | | | | | |
| s +35 to s +45 | | | work area the system) | - | - | _ | - | | |
| I/O data area fo | or No. n loop | | | | | | | | |
| (m+0) (জ | Set value | SV | | | | | | | |
| s +(m+1) | Process value | PV | | | | | | | |
| জ +(m+2) | Automatic manipulated value | MV | | | | | | | |
| জ +(m+3) | Process value after filtering | PVf | The same as I/O data are | ea for No. 1 loop | | | | | |
| জ +(m+4) | Manual manipulated value | MVman | | | | | | | |
| জ +(m+5) | Manual/ automatic selection | MAN/ AUTO | | | | | | | |
| s +(m+6) | Alarm | ALARM | | | | | | | |
| <pre>(\$ +(m+7) to (\$ +(m+17))</pre> | | | work area the system) | | - | _ | - | | |

 $m=(n-1) \times 18+10$ n: number of loops

Precautions

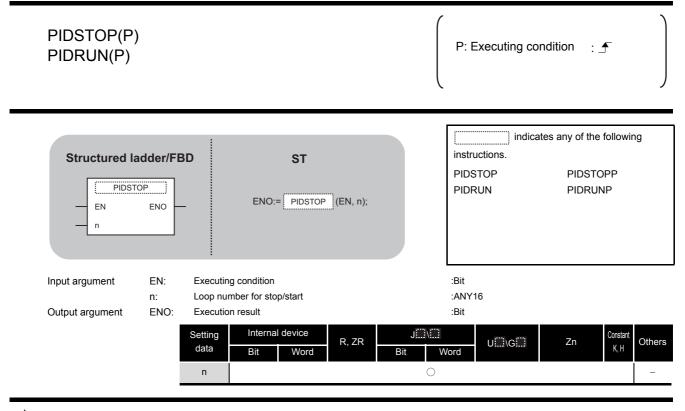
The following table shows the CPU modules applicable to the PID control instructions (inexact differential) and the PID control instructions (exact differential).

| | CPU module model | Inexact differential | Exact differential |
|------------------------|---|-------------------------|-----------------------|
| Basic model QCPU | The first five digits of the serial number are '04122' or higher | 0 | 0 |
| High Performance model | The first five digits of the serial number are '05031' or lower. | × | 0 |
| QCPU | The first five digits of the serial number are '05032' or higher. | 0 | 0 |
| Universal model QCPU | | 0 | 0 |
| LCPU | | 0 | 0 |

 \bigcirc : Applicable, \times : Not applicable

6.2.3 PIDSTOP instruction and PIDRUN instruction

PIDSTOP, PIDRUN



Grant Function

(1) PIDSTOP(P)

This instruction stops the PID operation for the loop number specified by 'n'.

(2) PIDRUN(P)This instruction starts the PID operation for the loop number specified by 'n'.

Precautions

The following table shows the CPU modules applicable to the PID control instructions (inexact differential) and the PID control instructions (exact differential).

| | CPU module model | Inexact differential | Exact differential |
|------------------------|---|-------------------------|--------------------|
| Basic model QCPU | The first five digits of the serial number are '04122' or higher | 0 | 0 |
| High Performance model | The first five digits of the serial number are '05031' or lower. | × | 0 |
| QCPU | The first five digits of the serial number are '05032' or higher. | 0 | 0 |
| Universal model QCPU | | 0 | 0 |
| LCPU | | 0 | 0 |

 \bigcirc : Applicable, \times : Not applicable

6.2.4 PIDPRMW instruction

PIDPRMW

| PIDPRMW(F | P) | | | | | | P: Exe | ecuting co | ndition : | | |
|--|-----------------|--------------------------------|---------|----------------------|--------------|-------------------|----------------------------------|------------|-------------------------|---------------------|--------|
| Structured Ia PIDPR EN n s | | 3D - | ENO:= | ST PIDPRMW | (EN, n, s |); | instructi PIDPRN | ons. | ates any of th PIDPF | ne followir RMWP | ng |
| Input argument | EN: n: s: | Loop nu Start nu changeo | t | - | tores PID co | ontrol data to be | :Bit :ANY16 :ANY16 :Bit | | | | |
| Output argument | ENO: | Executio | Interna | device | | J::::\:::: | BI | | | Constant | |
| | | Setting data | Bit | Word | R, ZR | | /ord | U\G | Zn | Constant K, H | Others |
| | 1 | n | 0 | C |) | | (| C | | 0 | _ |
| | | s | - | (|) | | | _ | | _ | _ |

Grant Function

This instruction changes the operation parameter of the loop number specified by 'n' to the PID control data stored in the devices starting from the device number specified by (s).

(1) PID control data

| | | | Setting | range | Setting | Processing when the |
|--------------|--|---|--|--|---------|---|
| Device | Data item | Description | With PID limits | Without PID limits | side | setting data are outside the setting range |
| s) +0 | Operational expression selection | Select the PID operational expression. *1 | 0: Forward operation 1: Reverse operation | 0: Forward operation 1: Reverse operation | User | An error occurs and the PID operation for the corresponding loop is not |
| s) +1 | Sampling cycle (Ts) | Set the PID operation cycle. | 1 to 6000 (unit: 10ms) | 1 to 6000 (unit: 10ms) | User | performed. |

*1 : For the PID operational expressions set for Operational expression selection, refer to MELSEC-Q/L/QnA Programming Manual (PID Control Instructions).

| | | | Setting | range | Setting | Processing when the setting data are outside the setting range | |
|--------|---|--|--|--|---------|--|--|
| Device | Data item | Description | With PID limits | Without PID limits | side | | |
| s +2 | Proportional constant (K _P) | Proportional gain of PID operation | 1 to 10000 (unit: 0.01) | 1 to 10000 (unit: 0.01) | User | | |
| s) +3 | Integral constant (Tı) | Constant that expresses the magnitude of the integral action (I action) effect. Increasing the integral constant slows down the manipulated value change. | 1 to 32767 (unit: 100ms) If setting value > 30000 Ti = Infinite (∞) | 1 to 32767 (unit: 100ms) If setting value > 30000 Tı = Infinite (∞) | User | | |
| s) +4 | Derivative constant (Tɒ) | Constant that expresses the magnitude of the derivative action (D action) effect. Increasing the derivative constant causes significant changes in the manipulated value even with a slight change of the control target. | 0 to 30000 (unit: 10ms) | 0 to 30000 (unit: 10ms) | User | An error occurs and the PID operation for the corresponding loop is not performed. | |
| s) +5 | Filter coefficient (α) | Set the degree of filtering to be applied to the process value. The filtering effect decreases as the value gets closer to 0. | 0 to 100 | 0 to 100 | User | | |
| s +6 | MV lower limit (MVLL) | Set the lower limit for the manipulated value (MV) calculated in PID operation in automatic mode. If the MV is less than the set lower limit value (MVLL), the value is clipped to the MVLL. | -50 to 2050 | -32768 to 32767 | User | In the case of "With PID limits", the PID operation is not performed after values are replaced as follows: • If MVLL or MVHL value is less than -50, the value is clipped to -50. • If MVLL or MVHL value is greater than 2050, the value is clipped to 2050. | |
| (s) +7 | MV upper limit (MVHL) | Set the upper limit for the manipulated value calculated in PID operation in automatic mode. If the MV is greater than the set upper limit value (MVHL), the value is clipped to the MVHL. | -50 to 2050 | -32768 to 32767 | User | | |

| | | | Setting | range | Setting | Processing when the |
|--------|------------------------------|---|-----------------|-----------------------|---------|---|
| Device | Data item | Description | With PID limits | Without PID limits | side | setting data are outside the setting range |
| ভ +8 | MV change rate limit (| Set the variation limit between the previous MV and the present MV. When the MV variation is greater than the limit value, bit 1 (b1) of the alarm device is set to '1'. MV variation is not limited. (Even if the MV variation exceeds the limit value, the actual MV variation is used as it is for calculating the MV.) | 0 to 2000 | -32768 to 32767 | User | In the case of "With PID limits", the PID operation is performed after values are replaced as follows: △ MVL value is less than 0, the value is clipped to 0. △ MVL value is greater than 2000, the value is clipped to 2000. |
| © +9 | PV change rate limit (| Set the variation limit between the previous PV and the present PV. When the PV variation is greater than the limit value, bit 0 (b0) of the alarm device is set to '1'. PV variation is not limited. (Even if the PV variation exceeds the limit value, the actual PV variation is used as it is for performing the PID operation.) | 0 to 2000 | -32768 to 32767 | User | In the case of "With PID limits", the PID operation is performed values are replaced as follows: If the △ PVL value is less than 0, the value is clipped to 0. If the △ PVL value is greater than 2000, the value is clipped to 2000. |

Precautions

The following table shows the CPU modules applicable to the PID control instructions (inexact differential) and the PID control instructions (exact differential).

| | CPU module model | Inexact differential | Exact differential |
|------------------------|---|-------------------------|-----------------------|
| Basic model QCPU | The first five digits of the serial number are '04122' or higher | 0 | 0 |
| High Performance model | The first five digits of the serial number are '05031' or lower. | × | 0 |
| QCPU | The first five digits of the serial number are '05032' or higher. | 0 | 0 |
| Universal model QCPU | | 0 | 0 |
| LCPU | | 0 | 0 |

 \bigcirc : Applicable, \times : Not applicable

MEMO

| |
|------|
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |

SOCKET COMMUNICATION FUNCTION INSTRUCTION

| 7.1 | SOCOPEN Instruction |
|-----|----------------------|
| 7.2 | SOCCLOSE Instruction |
| 7.3 | SOCRCV Instruction |
| 7.4 | SOCRCVS Instruction |
| 7.5 | SOCSND Instruction |
| 7.6 | SOCCINF Instruction |
| 7.7 | SOCCSET Instruction |
| 7.8 | SOCRMODE Instruction |
| 7.9 | SOCRDATA Instruction |

1

2 OVERVIEW

INSTRUCTION TABLES

3

CONFIGURATION OF INSTRUCTIONS

7.1 SOCOPEN Instruction

SP_SOCOPEN

QnUDE(H) LCPU

| data Bit Word Bit Word (st) - () - | : 1 | ondition : | Executing condition | | | | | EN | SP_SOCOP | |
|---|------------------|-----------------|---|-------------------|------------------------------|---|--|---------------------------|---|--|
| Un: Dummy ("U0") :String s1: Connection number (1 to 16) :ANY16 s2: Variable that stores control data :Array of ANY16 [09] ENO: Execution result :Bit d: Variable that turns ON during one scan upon completion of the :Array of bit [01] instruction instruction d[1] also turns ON at the time of error completion. Internal device R, ZR J(1) Setting Internal device R, ZR Bit Word Zn s: - O - - - - | owing instruct | ates the follow | · | EN, Un, s1,s2,d); | | :NO:= <u>SP_</u> | _ | ENO | — EN — Un — s1 | |
| data Bit Word R, ZR Bit Word Uiii:\Giiii Zn (st) - O - - - - | | 9] | :String :ANY16 :Array of ANY16 [09] :Bit | | control data N during one | ("U0") tion number that stores on result that turns O on | Dummy Connect Variable Executio Variable instructio | Un: s1: s2: ENO: | Un: s1: s2: Output argument ENO: | |
| | Constant K, H | Zn | Vord UIII\GIII | Bit | Word | Bit | data | | | |
| $ \begin{array}{c ccccccccccccccccccccccccccccccccccc$ | | | - | ^*1 ^*1 | | - ^ *1 | 62 | - | | |

Grant Function

This instruction establishes a connection.

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|----------------------|--------------------------------------|--|---|--------------|
| ©[0] | Execution type/ Completion type | Specify which to use the parameter values set by GX Works2 or the setting values of the following control data (<a>[2] to <a>[9]) at open processing of a connection. 0000H: Uses the parameter set in [Open settings] of GX Works2. 8000H: Uses the settings of control data <a>[2] to <a>[9] . | 0000н, 8000н | User |
| ⊚[1] | Completion status | The instruction application status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| ⊚[2] | Application setting area | b15b14b13 to b10 b9 b8 b7 to b0 c) [2] 3 0 21 0 Communication method (protocol) 0: TCP/IP 1: UDP/IP With/without procedure in socket communication function 1: Nonprocedural communication Open system 00: Active open or UDP/IP 10: Unpassive open 11: Fullpassive open | (See the left column.) | User |
| ⊚[3] | Host station port No. | Specify the port number of the host station. | 1н to 1387н, 1392н to FFFEн (400н or later is recommended) | User |
| © [4] © [5] | Destination IP address ^{*2} | Specify the IP address of the external device. | 1н to FFFFFFFFн (FFFFFFFFн: broadcast) | User |
| ⊚[6] | Destination port No. ^{*2} | Specify the port number of the external device. | 1н to FFFFн (FFFFн: broadcast) | User |
| © [7] to © [9] | - | Unavailable | _ | System |

*1 : "Destination IP address" and "Destination port No" are neglected at Unpassive open.

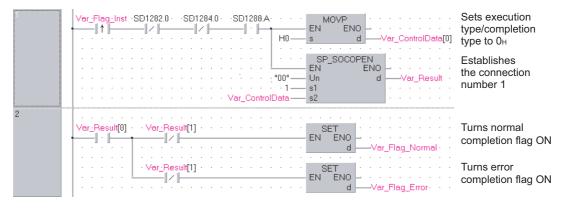
Precautions

Use the Built-in Ethernet port QCPU of which the function version is B or later and the first five digits of the serial number are '11012' or higher.

Program Example

The following program opens the connection 1.

[Structured ladder/FBD]



[ST]

IF((LDP(TRUE, Var_Flag_Inst)) &(SD1282.0=FALSE) &(SD1284.0=FALSE) &(SD1288.A=TRUE))THEN MOVP(TRUE, H0, Var_ControlData[0]); (* Sets execution type/completion type to 0H *) SP_SOCOPEN(TRUE, "00", 1, Var_ControlData, Var_Result); (* Establishes the connection number 1 *) END_IF; IF(Var_Result[0]=TRUE)THEN (* Execution finished *) IF(Var_Result[1]=FALSE)THEN (* Normal completion *) (* Turns normal completion flag ON *) SET(FALSE, Var_Flag_Normal); ELSE (* Error completion *) SET(TRUE, Var_Flag_Error); (* Turns error completion flag ON *) END_IF;

END_IF;

7.2 SOCCLOSE Instruction

SP_SOCCLOSE



| SP_SOCCL | OSE | | | | | Executing co | ondition : | : _ | |
|--|--------------------------|--|--|------------------|------------|---|----------------|------------------|-----------|
| Structured I SP_SOC EN Un s1 s2 | | _ | ST | (EN, Un, s1,s2,d | d); | SP_SOCCLOSE | tes the follow | wing instri | uction |
| Input argument | EN: Un: s1: s2: | Executing condi Dummy ("U0") Connection nun Variable that sto | nber (1 to 16) | | | :Bit :String :ANY16 :Array of ANY16 [01] | 1 | | |
| s2: Variable that stores control data Output argument ENO: Execution result d: Variable that turns ON during one scan upon completion or instruction d[1] also turns ON at the time of error completion. | | | | | | | 1 | | |
| Output argument | | Variable that tur instruction | ns ON during on | | | :Bit :Array of bit [01] | | | |
| Output argument | | Variable that tur instruction d[1] also turns (| ns ON during on DN at the time of ernal device | | n. J∭\∭ | | Zn | Constant K, H | Othe |
| Output argument | | Variable that tur instruction d[1] also turns (Setting Int | ns ON during on DN at the time of ernal device | error completio | n. J∭\∭ | :Array of bit [01] | Zn | | Othe |
| Output argument | | Variable that tur instruction d[1] also turns (Setting data | ns ON during on DN at the time of ernal device Word | error completio | n. J∭\∭ | :Array of bit [01] | Zn | K, H | Othe _ |

Grant Function

This instruction shuts off a specified connection.

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|--------|-------------------|---|---------------|--------------|
| s2 [0] | System area | - | - | - |
| ፼[1] | Completion status | The instruction completion status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | - | System |

Precautions

Use the Built-in Ethernet port QCPU of which the function version is B or later and the first five digits of the serial number are '11012' or higher.

Program Example

The following program shuts off the connection 1 when the disconnect request flag turns ON or the external device closes the connection 1.

| 2 | ·SD1282.0 · ·SD1284.0 · · · · · · · · · · · · · · · · · · · | EN ENO | Turns flag ON when open completion signal turns OFF |
|---|---|-------------|---|
| | Var_Flag_Inst ·SD1282.0 · Var_Flag_Exe 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 | EN ENO | Shuts off the connection number 1 |
| 2 | · Var_Flag. · · · · · · · · · · · · · · · · · · · | | Turns execution flag ON |
| | Var_Result[0] · · Var_Result[1] · · · · · · · · · · · · · · · · · · · | EN ENO | Turns normal completion flag ON |
| | · · · · · · · · · · · · · · · · · · · | EN ENO d | Turns error completion flag ON |
| | | EN ENO d | Turns execution flag OFF |

[Structured ladder/FBD]

```
[ST]
IF(LDF(TRUE, SD1282.0)&(SD1284.0=TRUE))THEN
                                  (* When open completion signal turns OFF *)
    PLS(TRUE, Var_Flag);
                                  (* Turns flag ON *)
END_IF;
IF(((LDP(TRUE, Var_Flag_Inst) & SD1282.0) OR Var_Flag) & (NOT Var_Flag_Exe)) THEN
    SP_SOCCLOSE(TRUE, "00", 1, Var_ControlData, Var_Result);
                                  (* Shuts off the connection number 1 *)
    SET(TRUE, Var_Flag_Exe);
                                  (* Turns execution flag ON *)
ELSE
    SP_SOCCLOSE(FALSE, "00", 1, Var_ControlData, Var_Result);
    SET(FALSE, Var_Flag_Exe);
END_IF;
IF(Var Result[0]=TRUE)THEN
                                        (* Execution finished *)
                                        (* Normal completion *)
    IF(Var_Result[1]=FALSE)THEN
          SET(FALSE, Var_Flag_Normal);(* Turns normal completion flag ON *)
    ELSE
                                        (* Error completion *)
          SET(TRUE, Var_Flag_Error);
                                       (* Turns error completion flag ON *)
    END IF;
    RST(TRUE, Var_Flag_Exe);
                                        (* Turns execution flag OFF *)
END_IF;
```

7.3 SOCRCV Instruction

SP_SOCRCV

QnUDE(H) LCPU

| SP_SOCRC | V | | | | | | Executing o | condition : _ | | |
|--|-------------------|---|--|------------------------------------|------------------------------|------------------|---|-------------------|------------------|---------|
| Structured I SP_SOU EN Un s1 s2 | | _ | IO:= | ST | N, Un, s1,s2 | d1,d2); | SP_SOCRCV | ates the followir | ng instr | uction. |
| Input argument | EN: Un: s1: | Dummy Connect | ng condition ("U0") tion number that stores | . , | | | Bit String ANY16 Array of ANY16 [0 | 11 | | |
| s2: Output argument ENO: d1: d2: | | Executio Start nur Variable instructio | on result mber of the that turns O | device that s N during on | stores receiv e scan upor | completion of th | :Anay of ANY 16 [0 :Bit :ANY16 ne :Array of bit [01] | "] | | |
| | | Setting data | Interna Bit | l device Word | R, ZR | J\ Bit | U:::\G::: Word | Zn | Constant K, H | Othe |
| | - | s1 | - | 0 | O | | | | 0 | _ |
| | - | \$2 (d1) | _ | △ ^{*1} △ ^{*1} | *1 ^*1 | | | | | |

*1: Local devices and file registers per program cannot be used as setting data.

Grant Function

This instruction reads receive data of a specified connection from the socket communication receive data area at the end process performed after the instruction execution.

△*1

 \triangle^{*1}

d2)

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|--------|-------------------|--|---------------|--------------|
| s2 [0] | System area | - | _ | - |
| | | The instruction completion status is stored. | | |
| s2 [1] | Completion status | 0 : Normal completion | - | System |
| | | Other than 0 : Error completion (error code) | | |

| Device | Item | Setting data | Setting range | Setting side |
|--------|---------------------|---|---------------|--------------|
| | | Data length of the data read from the socket communication receive | | |
| d1 +0 | Receive data length | data area is stored. | 0 to 2046 | System |
| | | (number of bytes) | | |
| d1) +1 | | Date read from the application manipulation reading date area are stand | | |
| to | Receive data | Data read from the socket communication receive data area are stored in ascending address order. | - | System |
| @1 +n | | | | |



Use the Built-in Ethernet port QCPU of which the function version is B or later and the first five digits of the serial number are '11012' or higher.

Program Example

The following program reads data received from the external device.

[Structured ladder/FBD]

| 1 | Var_Flag_inst SD1282:0 SD1286:0 Var_Result[0] SP_SOCRCV I | Reads data from the connection number 1 |
|---|---|---|
| 2 | Var_Result[0] Var_Result[1] I I </td <td></td> | |
| | ····· Var_Result[1] SET ···· I····· EN ···· I····· ···· I····· ····· I····· ····· I····· ····· I····· ····· I····· ····· I······ ····· I······ ····· I······ ····· I······ ····· ······ ····· ········ ····· ······ ····· ······ ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ····· ······ ····· ······ ····· ······ ·····< | Turns error completion flag OFF |

[ST]

IF((Var_Flag_Inst=TRUE) &(SD1282.0=TRUE) &(SD1286.0=TRUE) &(Var_Result[0]=FALSE))THEN SP_SOCRCV (TRUE, "U0", 1, Var_ControlData, D500, Var_Result);

END_IF;

```
IF(Var_Result[0]=TRUE)THEN
IF(Var_Result[1]=FALSE)THEN
SET(TRUE, Var_Flag_Normal);
ELSE
SET(TRUE, Var_Flag_Error);
END_IF;
END_IF;
```

- (* Execution finished *)
- (* Normal completion *)
- (* Turns normal completion flag ON *)

(* Reads data from the connection number 1 *)

- (* Error completion *)
- (* Turns error completion flag ON *)

7.4 SOCRCVS Instruction

S_SOCRCVS



| S_SOCRCV | S | | | | | | | Executing c | ondition : | | _ |
|--|-------------------------------------|-----------------|----------------|----------------------------|---------------|--------|--|-------------|-----------------|------------------|--------|
| Structured Ia S SOCE EN Un s | | | ENO:= | ST SOCRCVS | (EN, Un, s, (| J); | | OCRCVS | ates the follow | wing instr | uction |
| Input argument Output argument | Un: I s: 0 it argument ENO: I | | | (1 to 16) device that s | stores receiv | e data | :Bit :Strin :ANY :Bit :ANY | 16 | | | |
| | | Setting data | Interna Bit | l device Word | R, ZR | J | \ Word | - U∭\G∭ | Zn | Constant K, H | Othe |
| | | S | - | 0 | 0 | | | _ | | 0 | _ |
| | | d | - | 0 | 0 | | | _ | | - | - |

Grant Function

This instruction reads receive data of a specified connection from the socket communication receive data area.

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|------------------------|---------------------|---|---------------|--------------|
| @[0] | Receive data length | Data length of the data read from the socket communication receive data area is stored. (number of bytes) | 0 to 2046 | System |
| (d) +1 to (d) +n | Receive data | Data read from the socket communication receive data area are stored in ascending address order. | _ | System |

Precautions

Use the Built-in Ethernet port QCPU of which the function version is B or later and the first five digits of the serial number are '11012' or higher.

Program Example

The following program reads data received from the external device.

[Structured ladder/FBD]

| 1 | | |
|---|---------------------------------------|-----------------------|
| | | Reads data from |
| | | the connection |
| | dD500 d | ⁰ number 1 |
| | · · · · · · · · · · · · · · · · · · · | · |
| | | |
| | | |
| | | |

[ST]

IF((Var_Flag_Inst=TRUE) &(SD1282.0=TRUE) &(SD1286.0=TRUE))THEN S_SOCRCVS(TRUE, "U0", 1, D5000);

(* Reads data from the connection number 1 *)

END_IF;

7.5 SOCSND Instruction

SP_SOCSND



| SP_SOCSN | D | | | | | Executing condition : |
|---|--------------------------|---------------------------------------|---------------------------------|---------------------------|---|---|
| Structured Ia SP SOC EN Un s1 s2 s3 | | - | NO:= | ST SOCSND (EN | N, Un, s1,s2,s3,d) | I); |
| Input argument | EN: Un: s1: s2: | Dummy (' Connectio Variable t | ion number (that stores c | (1 to 16) control data | | :Bit :String :ANY16 :Array of ANY16 [01] |
| Output argument | s3: ENO: d: | Execution Variable the instruction | n result that turns Ol on |)N during one | stores send data le scan upon com f error completion. | :Bit mpletion of the :Array of bit [01] |
| | ļ | Setting data | Internal Bit | Il device Word | R, ZR | JIII UIII/GIII Zn Constant Bit Word Zn K,H Oth |
| | Ī | s1 | - | 0 | 0 | - 0 - |
| | 7 | s2 | - | ^*1 | *1 | |
| | | | | | ł i | |
| | | \$3 | - △*1 | 0 | | |

S Function

This instruction sends data to the external device of a specified connection.

Control Data

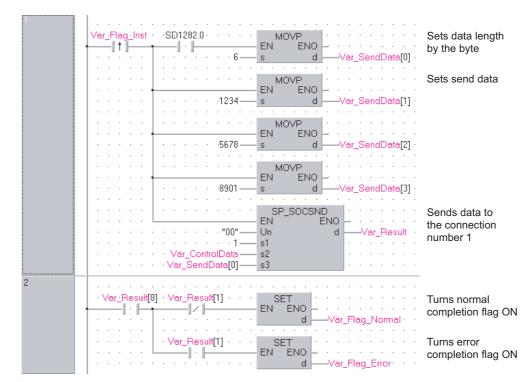
| Device | Item | Setting data | Setting range | Setting side |
|--------|-------------------|--|---------------|--------------|
| s2 [0] | System area | - | - | _ |
| | | The instruction completion status is stored. | | |
| s2 [1] | Completion status | 0 : Normal completion | - | System |
| | | Other than 0 : Error completion (error code) | | |

| Device | Item | Setting data | Setting range | Setting side |
|----------------------|------------------|--|---------------|--------------|
| s3 +0 | Send data length | Specify the data length of the send data. (number of bytes) | 0 to 2046 | User |
| €3 +1 to €3 +n | Send data | Specify the send data. | _ | User |

Use the Built-in Ethernet port QCPU of which the function version is B or later and the first five digits of the serial number are '11012' or higher.

Program Example

The following program sends data (1234, 5678, and 8901) to the external device using the socket communication function.



[Structured ladder/FBD]

[ST]

```
IF((Var_Flag_Inst=TRUE) &(SD1282.0=TRUE))THEN
    MOVP(TRUE, 6, Var_SendData[0]);
                                             (* Sets data length by the byte *)
    MOVP(TRUE, 1234, Var_SendData[1]);
                                             (* Sets send data *)
    MOVP(TRUE, 5678, Var_SendData[2]);
    MOVP(TRUE, 8901, Var SendData[3]);
    SP_SOCSND(TRUE, "00", 1, Var_ControlData, Var_SendData[0], Var_Result);
                                       (* Sends data to the connection number 1 *)
END_IF;
IF(Var Result[0]=TRUE)THEN
                                             (* Execution finished *)
    IF(Var Result[1]=FALSE)THEN
                                             (* Normal completion *)
```

SET(FALSE, Var_Flag_Normal); ELSE SET(TRUE, Var_Flag_Error); END_IF;

END IF;

- (* Turns normal completion flag ON *)
- (* Error completion *)
- (* Turns error completion flag OFF *)

7.6 SOCCINF Instruction

SP_SOCCINF

QnUDE(H) LCPU

| SP_SOCCIN | ١F | | | | | | Executing cor | ndition | : 🖍 | |
|--|--|--|-------------------------------------|-------------------|-----------------|-----|---|--------------|------------------|---------|
| Structured I SP_SOC EN Un s1 s2 | | _ | NO:= SP | ST _SOCCINF | (EN, Un, s1,s2, | d); | SP_SOCCINF | es the follo | wing instr | uction. |
| Input argument Output argument | EN: Un: s1: s2: ENO: d: | Dummy (Connecti Variable t Execution | on number hat stores n result | | | | :Bit :String :ANY16 :Array of ANY16 [01] :Bit :Array of ANY16 [04] | | | |
| | | Setting data | Interna Bit | al device Word | R, ZR | J\ | ₩ord | Zn | Constant K, H | Other |
| | | (s1) | - | 0 | 0 | | - | | 0 | _ |
| | | | | 1 | | | | | _ | |
| | | s2 | - | 0 | 0 | | - | | _ | _ |

This instruction reads connection information of a specified connection.

E Control Data

| Device | Item | Setting data | Setting range | Setting side |
|--------|-------------------|--|---------------|--------------|
| s2 [0] | System area | - | - | - |
| ፼[1] | Completion status | The instruction application status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |

| Item | Setting data | Setting range | Setting side | | | |
|------------------------------|--|--|---|--|--|--|
| | | 1н to | | | | |
| | | FFFFFFFH | | | | |
| Destination IP address | The IP address of the external device is stored | 0н : No | System | | | |
| | | destination | System | | | |
| | | (FFFFFFFFH: | | | | |
| | | broadcast) | | | | |
| (d) [2] Destination port No. | | 1н to FFFFн | | | | |
| | The port number of the external device is stored. | (FFFFH: | System | | | |
| | | broadcast) | | | | |
| d [3] Host station port No. | | 1н to 1387н, | | | | |
| | The port number of the host station is stored. | 1392н to | System | | | |
| | | FFFEH | | | | |
| Application setting area | b15b14b13 to b10 b9 b8 b7 to b0 (d)[4] (3) 0 (2)[1] 0 (1) 0 (2)[1] 0 (1) 0 (2)[1] 0 (2)[1] 0 (2)[1] 0 (2)[1] 0 (1) (1) (2) (1) (2) (1) (2) (1) (2) (1) (2) (1) (2) (1) (2) (1) (2) (1) (2) (1) (2) (1) (2) (1) (2) (1) (2) (1) (2) (1) (2) (1) (2) (1) (3) (2) (4) (1) (5) (2) (5) (2) (6) (2) (7) (2) (7) (2) (7) (2) | - | System | | | |
| | Host station port No. | Destination port No. The port number of the external device is stored. Host station port No. The port number of the host station is stored. $d[4]$ 3 0 2 10 0 0 2 10 | Destination IP address The IP address of the external device is stored. FFFFFFFH 0H: No destination (FFFFFFFH: broadcast) Destination port No. The port number of the external device is stored. 1H to FFFFH (FFFFFH: broadcast) Host station port No. The port number of the host station is stored. 1H to 1387H, 1392H to FFFFFEH Host station port No. The port number of the host station is stored. 1H to 1387H, 1392H to FFFEH Application setting area © Communication method (protocol) 0: TCP/IP 1: UDP/IP 0 @ With/without procedure in socket communication function 1: Nonprocedural communication 0: Active open or UDP/IP 10: Unpassive open - | | | |

Precautions

Use the Built-in Ethernet port QCPU of which the function version is B or later and the first five digits of the serial number are '11012' or higher.

Program Example

The following program reads connection information of the connection number 1. [Structured ladder/FBD]

| 1 | Var_Flag_inst SP_SOCCINF Image: SP_SOCCINF ENO Image: SP_SOCCINF Un Image: SP_SOCCINF Image: SP_SOCCINF Image: SP_SOCCINF Image: SP_SOCCINF <tr< th=""><th>Reads data from the connection number 1</th></tr<> | Reads data from the connection number 1 |
|---|---|---|
| | | |

[ST]

IF(Var_Flag_Inst=TRUE)THEN

SP_SOCCINF(TRUE, "U0", 1, Var_ControlData, Var_Connection);

(* Reads data from the connection number 1 *)

END_IF;

7.7 SOCCSET Instruction

SP_SOCCSET



| SP_SOCCS | ET | | | | | | | Executing co | ndition | : 🖍 | |
|---|--------------------------|--|-----------|---------------------------|--------------|-------------|-----------------------------------|-------------------|--------------|------------------|------------|
| Structured Ia SP SOC EN Un s1 s2 | | | NO:= | ST SOCCSET | (EN, Un, s1, | s2); | SP_ | indicat | es the follo | wing instr | uction. |
| Input argument | EN: Un: s1: s2: | Dummy (Connectio | on number | (1 to 16) control data | | | :Bit :String :ANY :Array | | | | |
| Output argument | ENO: | Execution Setting data (s1) (s2) | | Vord | R, ZR | J∭∖Ĭ Bit | :Bit | U[]\G[] - - | Zn | Constant K, H | Other – |

Function

This instruction changes the IP address and port number of the external device of a specified connection.

(Available only with a UDP/IP connection)

Control Data

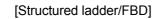
| Device | Item | Setting data | Setting range | Setting side |
|--------------|------------------------|--|---|--------------|
| s2 [0] | System area | - | - | - |
| s2 [1] | Completion status | The instruction application status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| @[2] @[3] | Destination IP address | Specify the IP address of the external device. | 1н to FFFFFFFH 0н: No destination (FFFFFFFFH: broadcast) | User |
| ⊚[4] | Destination port No. | Specify the port number of the external device. | 1н to FFFFн (FFFFн: broadcast) | User |

Precautions

Use the Built-in Ethernet port QCPU of which the function version is B or later and the first five digits of the serial number are '11012' or higher.

Program Example

The following program changes the destination (destination IP address and port number) of the connection number 1 which is being open.



| 1 | · Var_Flag_Inst · SD1282.0 · · · · · · · · · · · · · · · · · · · |
|---|--|
| | H55DE s d Var_ControlData[2] IP address to 10.97.85.222 |
| | |
| | ······································ |
| | MOVP |
| | H2001 EN ENO Var_ControlData[4] port number to 8193 |
| | |
| | SP_SOCCSET Changes the setting |
| | Of the connection |
| | Un |
| | |
| | ······································ |
| | |

[ST]

IF((LDP(TRUE, Var_Flag_Inst)) &(SD1282.0=TRUE) THEN MOVP(TRUE, H55DE, Var_ControlData[2]); MOVP(TRUE, H0A61, Var_ControlData[3]);

(* Sets destination IP address to 10.97.85.222 *)

MOVP(TRUE, H2001, Var_ControlData[4]); (* Sets destination port number to 8193 *) SP_SOCCSET(TRUE, "U0", 1, Var_ControlData); (* Changes the setting of the connection number 1 *)

END_IF;

7.8 SOCRMODE Instruction

SP_SOCRMODE

QnUDE(H) LCPU

| SP_SOCRM | IODE | | | | | E | xecuting co | ndition : | . _ | |
|---|----------------------------------|---|-------------------|-----------------|-----------|---|---------------------|---------------|------------------|---------|
| Structured I SP_SOCI EN Un s1 s2 | | | ST _SOCRMODE |)(EN, Un, s1, s | 52); | SP_SC | indicat | es the follow | wing instr | uction. |
| Input argument Output argument | EN: Un: s1: s2: ENO: | Executing condition Dummy ("U0") Connection numbe Variable that stores Execution result | er (1 to 16) | | | :Bit :String :ANY16 :Array o :Bit | f ANY16 [03] | | | |
| ouput argument | LNO. | | al device Word | R, ZR | J∰ Bit | | U:::\G::: - - | Zn | Constant K, H | Other |
| Function | | _ | | | | | | | | |

This instruction changes the TCP receive mode (unavailable for a UDP connection) and receive data size.

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|---------------|--------------------------------|--|---------------|--------------|
| s2 [0] | System area | - | - | - |
| ଛ2[1] | Completion status | The instruction application status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | _ | System |
| s2 [2] | TCP Receive Mode ^{*1} | Specify the TCP receive mode. 0 : TCP normal receive mode 1 : TCP fixed length receive mode | 0, 1 | User |
| €2[3] | Receive Data Size | Specify the receive data size of the socket communication. (number of bytes) | 1 to 2046 | User |

*1: Unavailable for a UDP connection.

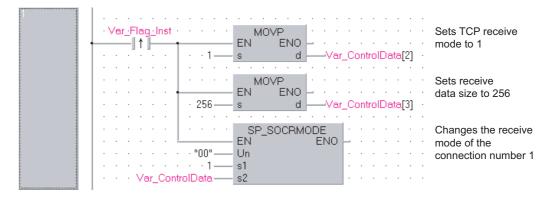
Precautions

Use the Built-in Ethernet port QCPU of which the function version is B or later and the first five digits of the serial number are '11012' or higher.

Program Example

The following program changes the receive mode of the connection number1 to TCP fixed length receive mode and changes its receive data length to 256 bytes.

After instruction execution, the connection number 1 turns the receive status signal ON when the length of receive data reaches 256 bytes.



[Structured ladder/FBD]

[ST]

IF (Var_Flag_Inst=TRUE) THEN

MOVP(TRUE, 1, Var_ControlData[2]); (* Sets TCP receive mode to 1 *) MOVP(TRUE, 256, Var_ControlData[3]); (* Sets receive data size to 256 *) SP_SOCRMODE(TRUE, "00", 1, Var_ControlData); (*Changes the receive mode of the connection number 1 *)

END_IF;

7.9 SOCRDATA Instruction

S_SOCRDATA

QnUDE(H) LCPU

| S(P)_SOCR | DATA | | | | | | P:Executing | condition | : 🖍 | |
|--|----------------------------------|---|---------------------------|------------------|---------------|------------------------|---|---------------|------------------|--------|
| Structure EN Un s1 s2 n | ed ladder OCRDATA ENO d | /FBD | ENO:= | S_SOCRD4 | ST (EN, Un | , s1, s2, n, d); | S_SOCRDATA | tes the follo | wing instr | |
| Input argument | EN: Un: s1: s2: | Dummy Connect Variable | ion number that stores | control data | | | :Bit :String :ANY16 :Array of ANY16 [01] | | | |
| Output argument | n: ENO: d: | Number of read data (1 to 1024 words) Execution result Variable that stores read data | | | | ANY16 :Bit ANY16 | | | | |
| | | Setting data | Interna Bit | l device Word | R, ZR | J∭\III Bit V | Vord | Zn | Constant K, H | Others |
| | Ī | (s1) | _ | 0 | 0 | | - | | 0 | - |
| | | <u>s2</u> | - | 0 | 0 | | - | | - | - |
| | | n | _ | 0 | 0 | | _ | | 0 | |

Grant Function

This instruction reads data for the specified number of words from the socket communication receive data area of a specified connection, and stores it.

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|--------|-------------------|--|---------------|--------------|
| s2 [0] | System area | - | - | - |
| ፼[1] | Completion status | The instruction application status is stored. 0 : Normal completion Other than 0 : Error completion (error code) | - | System |

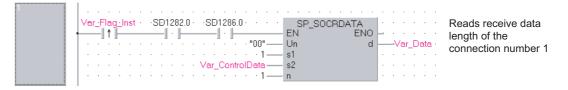
Precautions

Use the Built-in Ethernet port QCPU of which the function version is B or later and the first five digits of the serial number are '11012' or higher.

Program Example

The following program reads the receive data length of the connection number 1.

[Structured ladder/FBD]

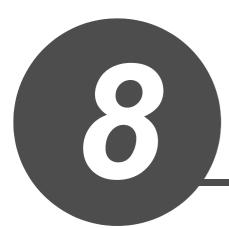


[ST]

IF((Var_Flag_Inst=TRUE) & (SD1282.0=TRUE) &(SD1286.0=TRUE))THEN SP_SOCRDATA(TRUE, "00", 1, Var_ControlData, 1, Var_Data); (* Reads receive data length of connection number 1 *)

END_IF;

MEMO



BUILT-IN I/O FUNCTION INSTRUCTION

| 8.1 | Positioning Function Dedicated Instruction | 8-2 |
|-----|--|------|
| 8.2 | Counter Function Dedicated Instruction | 8-18 |

8.1 Positioning Function Dedicated Instruction

8.1.1 IPPSTRT instruction

IPPSTRT1, IPPSTRT2

LCPU

| IPPSTRT1(F IPPSTRT2(F | | P: Executing condition : |
|--|--|---|
| Structured I — IPPST — EN — n | | indicates any of the following instructions. IPPSTRT1 IPPSTRT1P IPPSTRT2 IPPSTRT2P |
| Input argument Output argument | EN:Executing conditionn:Positioning data number (Setting range: 1 to 10)ENO:Execution result | :Bit :ANY16 :Bit |
| | Setting data Internal device R, ZR Ji n - O O | Word UIII\GIII Zn Constant Others |

☆ Function

This instruction specifies a data number to be executed for 'n' from the positioning data No. 1 to No. 10 which are previously set in GX Works2, and starts the specified axis (refer to the following).

- IPPSTRT1(P): Axis 1
- IPPSTRT2(P): Axis 2

Program Example

The following program starts the "Positioning Data" No. 1 of the Axis 1 when M0 turns ON.



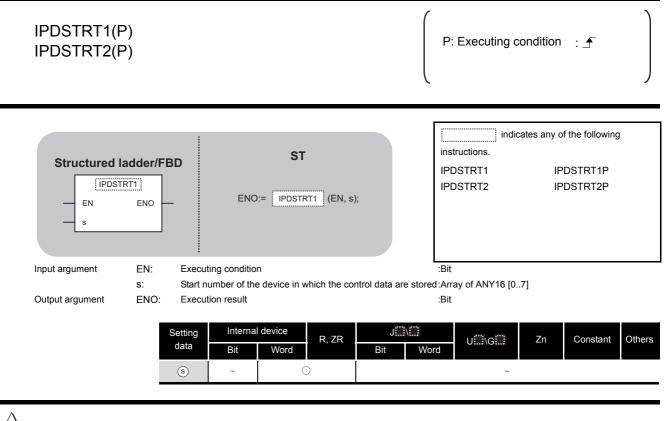
[ST] IPPSTRT1(M0, 1);

NULT-IN I/O FUNCTION

8.1.2 IPDSTRT instruction

IPDSTRT1, IPDSTRT2

LCPU



Grant Function

Regardless of "Positioning Data" No. 1 to No. 10 which are previously set in GX Works2, this instruction starts the positioning of the specified axis (refer to the following) using the data stored in the devices starting from \odot .

- IPDSTRT1(P): Axis 1
- IPDSTRT2(P): Axis 2

Control Data

| Device | Item | Setting data | Setting range | Setting side | | | |
|--------|-------------------------------------|-------------------------------------|-------------------------|--------------|--|--|--|
| | | 1: Positioning control (ABS) | | | | | |
| | | 2: Positioning control (INC) | | | | | |
| | | 3: Speed/position switching control | | | | | |
| | | (forward RUN) | | | | | |
| s [0] | Control system | 4: Speed/position switching control | 1 to 7 | | | | |
| | | (reverse RUN) | | | | | |
| | | 5: Current value change | | | | | |
| | | 6: Speed control (forward RUN) | | | | | |
| | | 7: Speed control (reverse RUN) | | | | | |
| ঙ [1] | Acceleration/deceleration time | 0 to 32767 (ms) | User | | | | |
| s [2] | Deceleration stop time | eceleration stop time – | | | | | |
| ঙ [3] | Dwell time | 0 to 65535 (ms) ^{*1} | | | | | |
| s [4] | Command speed | | 0 to 200000 | | | | |
| s [5] | | _ | (pulse/s) ^{*2} | | | | |
| § [6] | | | -2147483648 to | 1 | | | |
| | Positioning address/movement amount | - | 2147483647 | | | | |
| s [7] | | | (pulse) | | | | |

*1: Enter the setting value to the program as described below.

1 to 32767: Enter in decimal

- 32768 to 65535: Enter after converting it to hexadecimal
- *2: The restricted speed value may be applied when the set value of the command speed is not within 0 to 200000.

Program Example

The following program sets the following positioning data and starts the axis 1 when M0 turns ON.

| Device | Item | Setting data | | | |
|--------|--|---------------------------|--|--|--|
| D0 | Control system | Positioning control (ABS) | | | |
| D1 | Acceleration/deceleration time | 1000 (ms) | | | |
| D2 | Deceleration stop time | 1000 (ms) | | | |
| D3 | Dwell time | 0 (ms) | | | |
| D4, D5 | Command speed | 20000 (pulse/s) | | | |
| D6, D7 | Positioning address/movement amount | 100000 (pulse) | | | |

[Structured ladder/FBD]

| | · M0 | | | | | | | | | | MOV |
|-----|------|-----|---|---|---|---|----|---|-----|---------------------------------|---------------------|
| · | - | | | | | | | | | · ·1— | EN ENO s d |
| | • • | • • | · | • | • | • | · | • | | | |
| | | | | | | | Ľ | | | | |
| | · · | • • | | | · | | · | | · | ·1000 | s d |
| · | • • | • • | · | | | • | · | · | 1 | | |
| | • • | | : | ÷ | Ì | : | Ŀ. | • | • | | EN ENO |
| | · · | • • | | | | | . | | | ·1000 | s d |
| · · | • • | • • | · | • | · | · | · | · | · | | |
| | · · | | : | ÷ | ÷ | : | | • | • | | |
| | | • • | | | | | . | | | · · 0— | s d |
| | • • | • • | • | 1 | | | · | | | | · · · · · · · |
| | · · | | : | ÷ | ÷ | : | · | • | • | | |
| | | | | | | | . | | | 20000 — | s d |
| · · | • • | • • | · | · | · | · | · | · | · | | |
| | • • | • • | ÷ | ÷ | ÷ | : | ŀ | • | • | | |
| | | | | | | | | | - 1 | 00000 | s d |
| - · | · · | · · | · | | | | · | ÷ | ÷ | \cdot \cdot \cdot \cdot | |
| · · | • • | • • | • | ÷ | • | : | · | • | · | | IPDSTRT1P EN ENC |
| | | | | | | | | | | · D0 | - S |

[ST]

MOV(M0, 1, D0); MOV(M0, 1000, D1); MOV(M0, 1000, D2); MOV(M0, 0, D3); DMOV(M0, 20000, D4); DMOV(M0, 100000, D6); IPDSTRT1P(M0, D0);

8.1.3 IPSIMUL instruction

IPSIMUL

LCPU

| IPSIMUL(P) | | P: Executing condition : |
|-----------------------------------|--|---|
| Structured EN n1 n2 | ENO ENO:= IPSIMUL (EN, n1, n2); | indicates any of the following instructions. IPSIMUL IPSIMULP |
| Input argument Output argument | EN:Executing conditionn1:Axis 1 positioning data numbern2:Axis 2 positioning data numberENO:Execution result | :Bit :ANY16 :ANY16 :Bit |
| | Setting dataInternal device R, ZRBitWordR, ZRn1-On2-O | UIGI Zn Constant Others |

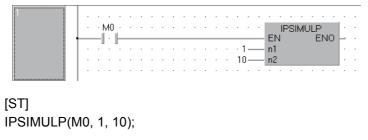
Grant Function

This instruction simultaneously starts the positioning of the axis 1 positioning data number specified by n1 and the axis 2 positioning data number specified by n2.

Program Example

The following program simultaneously starts the axis 1 positioning data No. 1 and the axis 2 positioning data No. 10 when M0 turns ON.

[Structured ladder/FBD]



LCPU

8.1.4 IPOPR instruction

IPOPR1, IPOPR2

8

| IPOPR1(P) IPOPR2(P) | | P: Executing condition : |
|-----------------------------------|--|---|
| Structured I | | indicates any of the following instructions. IPOPR1 IPOPR1P IPOPR2 IPOPR2P |
| Input argument Output argument | EN: Executing conditions: Start number of the device in which the control dENO: Execution result | :Bit ata are stored:Array of ANY16 [02] :Bit |
| | Setting data Internal device R, ZR Bit Word Bit | JINI UII∖GII Zn Constant Others Bit Word – |

Grant Function

This instruction starts the OPR of which type is specified by \odot on the specified axis (refer to the following).

- IPOPR1(P): Axis 1
- IPOPR2(P): Axis 2

Control Data

| Device | Item | Setting data | Setting range | Setting side | |
|---------|---|-------------------------------|--------------------------|--------------|--|
| | | 1: Machine OPR | | | |
| s [0] | OPR type | 2: Fast OPR (OP address) | 1 to 3 | | |
| | | 3: Fast OPR (standby address) | | | |
| s [1] | Standby address | | -2147483648 to | User | |
| 0.1 | , , , , , , , , , , , , , , , , , , , | | 2147483647 (pulse) | | |
| s [2] | (Set only when Fast OPR (standby address (3)) is set for the OPR type) | - | (Ignored when other than | | |
| (I=1 () | (3)) is set tor the OFK type) | | standby address (3)) | | |

Program Example

The following program starts the machine OPR of the axis 1 when M0 turns ON.

| Device | Item | Setting data | | | | |
|--------|-----------------|--------------|--|--|--|--|
| D0 | OPR type | Machine OPR | | | | |
| D1, D2 | Standby address | 0 (Ignored) | | | | |

[Structured ladder/FBD]

| 1 | - 1 | | | | | | | | | | | | | | | | | | |
|---|-----|---|---|----|----|---|---|---|---|---|----------|---|---|---|---|---|-----|---|------------|
| | - 1 | | • | | | | | | • | | | | | • | | | • | | |
| | | • | | M0 | i. | | • | • | • | • | Ì | • | • | • | • | • | • | • | |
| | | | • | | | | | | | | Ι | | | | | | 1— | | |
| | - 1 | | • | | | | | | • | | | | | | | | • | | |
| | | | | · | | | | | | | | | · | · | · | · | | · | DIMOV |
| | - 1 | | • | | | • | | | • | | t | _ | _ | _ | _ | _ | _ | _ | EN ENO |
| | | | | | · | · | · | · | | ÷ | | | · | · | | · | 0 — | | s d — D1 · |
| | - 1 | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | IPOPR1P |
| | - 1 | | | | | | | | | | <u> </u> | | | | | | | _ | EN ENOL··· |
| | | | | | | | | | | | | | | | | D | 0 — | | S |
| | | | · | · | · | · | · | · | · | ÷ | · | · | · | · | · | · | · | · | |

[ST] MOV(M0, 1, D0); DMOV(M0, 0, D1); IPOPR1P(M0, D0);

8.1.5 IPJOG instruction

IPJOG1, IPJOG2

LCPU

8

IPJOG1 IPJOG2

| Structured la PJOC EN s1 s2 | 31 | ENO:= | ST IPJOG1 | (EN, s1, s | 52); | ins IPJ | indic tructions. IOG1 IOG2 | cates any c | of the following | 3 |
|---|-------------------|--|------------------|------------|-----------|------------|-------------------------------------|-------------|------------------|--------|
| Input argument | EN: s1: s2: | Executing conditio Start number of the Direction specifica 0: Forward RUN 1: Reverse RUN | e device in v | | | :Bit | | .3] | | |
| Output argument | | Execution result Setting data Bit (s) - (s2 O | l device Word | R, ZR | Ji Bit | :Bit | U[]]\G[]] - - | Zn | Constant | Others |

Grant Function

This instruction starts the JOG operation of the specified axis (refer to the following).

- IPJOG1: Axis 1
- IPJOG2: Axis 2

The JOG operation is executed in the direction specified by 0, using the JOG speed, JOG acceleration/deceleration time stored in the devices starting from 0.

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|--------|-----------------------|--------------|-------------------------|--------------|
| s1 [0] | JOG speed | | 0 to 200000 | |
| s1 [1] | | | (pulse/s) ^{*1} | User |
| s1 [2] | JOG acceleration time | - | 0 to 32767 (ms) | 0301 |
| s1 [3] | JOG deceleration time | - | 0.0002707 (113) | |

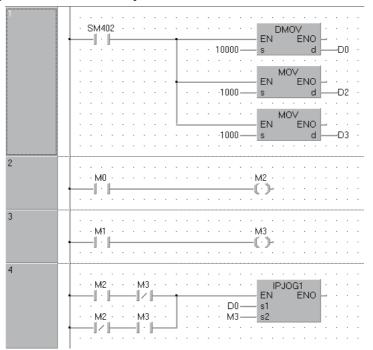
*1: The restricted speed value may be applied when the set value of the JOG speed is not within 0 to 200000.

Program Example

The following program starts the forward JOG operation when M0 turns ON, and starts the reverse JOG operation when M1 turns ON.

| Device | Item | Setting data |
|--------|-----------------------|-----------------|
| D0, D1 | JOG speed | 10000 (pulse/s) |
| D2 | JOG acceleration time | 1000 (ms) |
| D3 | JOG deceleration time | 1000 (113) |

[Structured ladder/FBD]



[ST] DMOV(SM402, 10000, D0); MOV(SM402, 1000, D2); MOV(SM402, 1000, D3); OUT(M0, M2); OUT(M1, M3); IPJOG1(M2 AND NOT M3 OR NOT M2 AND M3, D0, M3);

BUILT-IN I/O FUNCTION INSTRUCTION

8.1.6 IPABRST instruction

IPABRST1, IPABRST2



| PABRST1 PABRST2 | | | Executing condition : |
|------------------------------|------------|---|---|
| Structured PAB EN s | ladder/FBD | ST ENO:= IPABRST1 (EN, s, d) | indicates any of the following instructions. IPABRST1 IPABRST2 |
| nput argument | EN: E | Executing condition | :Bit |
| | | Start number of the device for input | :Array of bit [02] |
| Output argument | | Execution result | :Bit |
| | d: Set | Start number of the device for output ting Internal device R, ZR tta Bit Word | :Array of bit [02] J∭\[] Bit Word U[]\G[] Zn Constant |

Grant Function

This instruction executes the absolute position restoration of the specified axis (refer to the following) by communicating with the servo amplifier using the input device specified by s and output device specified by s.

- IPABRST1: Axis 1
- IPABRST2: Axis 2

Control Data

(1) Signals imported from servo amplifier

| Device | Item | Setting data | Setting range | Setting side | |
|--------|---------------------------------------|---------------------|---------------|--------------|--|
| s [0] | | ABS send data bit0 | | | |
| s [1] | Signals imported from servo amplifier | ABS send data bit1 | 0, 1 | User | |
| s [2] | | ABS send data ready | | | |

(2) Signals exported to servo amplifier

| Device | Item | Setting data | Setting range | Setting side |
|---------|-------------------------------------|-------------------|---------------|--------------|
| (0) (0) | | Servo ON | | |
| d [1] | Signals exported to servo amplifier | ABS transfer mode | - | System |
| d [2] | * | ABS request flag | | |

Program Example

This instruction executes the absolute position restoration of the axis 1 when M0 turns ON.

- X20 to X22: Signals imported from the servo amplifier
- Y30 to Y32: Signals exported to the servo amplifier

[Structured ladder/FBD]

| 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|----|-----|---|---|---|---|---|---|---|---|---|-----|-----|---|---|---------|-----|----|----|------|-----------|---|---|----|----|---|---|
| | · | | MO | - A | ÷ | · | · | ÷ | · | · | ÷ | · | · | ÷ | · | · | | | IP. | AB | RS | ST 1 | } | | • | · | · | ÷ | · |
| | | _ | Ľ | | | | | | | | | | | . ; | ×20 |) | | EN s | 1 | | | E | INU: (| | _ | _Y | 30 | | : |
| | · | | · | | | | | | · | | · | · | | | | | · | • | • | • | • | • | • | · | | | | | |

[ST] IPABRST1(M0, X20, Y30);

LCPU

8.1.7 IPSTOP instruction

IPSTOP1, IPSTOP2

8

IPSTOP1, IPSTOP2

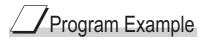
IPSTOP1 IPSTOP2

| Structured ladder/FBD | ST ENO:= IPSTOP (EN); | instructions. IPSTOP1 IPSTOP2 | cates any c | of the following |] |
|-----------------------|--------------------------|-------------------------------------|-------------|------------------|--------|
| | | Bit Bit ∪∭\G∭ | Zn | Constant | Others |

Grant Function

This instruction stops the positioning of the specified axis (refer to the following).

- IPSTOP1: Axis 1
- IPSTOP2: Axis 2



The following program stops the axis 1 when M0 turns ON.

[Structured ladder/FBD]

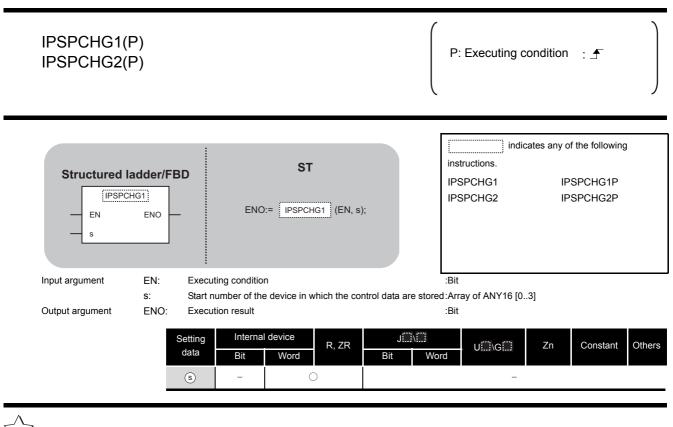




8.1.8 IPSPCHG instruction

IPSPCHG1, IPSPCHG2

LCPU



Grant Function

This instruction changes the speed of the specified axis (refer to the following) using the acceleration/deceleration time at speed change, deceleration stop time at speed change, and new speed value stored in the devices starting from (s).

- IPSPCHG1(P): Axis 1
- IPSPCHG2(P): Axis 2

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|--------|---|--------------|-------------------------|--------------|
| s [0] | Acceleration/deceleration time at speed | - | | |
| | change | | 0 to 32767 (ms) | |
| ঙ [1] | Deceleration stop time at speed change | - | | User |
| s [2] | New speed value | _ | 0 to 200000 | |
| s [3] | | | (pulse/s) ^{*1} | |

*1: The restricted speed value may be applied when the set value of the new speed is not within 0 to 200000.

Program Example

The following program changes the speed of the axis 1 when M0 turns ON.

| Device | Item | Setting data |
|--------|---|------------------|
| D0 | Acceleration/deceleration time at speed change | 2000 (ms) |
| D1 | Deceleration stop time at speed change | 1000 (ms) |
| D2, D3 | New speed value | 200000 (pulse/s) |

[Structured ladder/FBD]

| 1 | · · · · · · · · · · · · · · · · · · · |
|---|--|
| | |
| | |
| | · · · · · · · · · · · · · · · · · · |
| | MOV |
| | EN ENO |
| | · · · · · · · · · · · · 1000 — s d — D1 · |
| | · · · · · · · · · · · · · · · · · · · |
| | |
| | EN ENO |
| | · · · · · · · · · · · · 20000 — s d — D2 · |
| | |
| | PSPCHG1P |
| | EN ENO |
| | · · · · · · · · · · · · D0 — s |
| | |
| | |

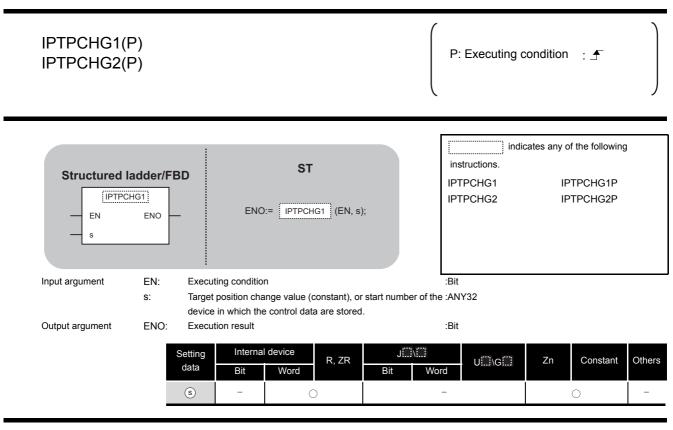
[ST]

MOV(M0, 2000, D0); MOV(M0, 1000, D1); DMOV(M0, 20000, D2); IPSPCHG1P(M0, D0); 8

8.1.9 IPTPCHG instruction

IPTPCHG1, IPTPCHG2

LCPU



This instruction changes the position of the specified axis (refer to the following) to the new target position specified by (s).

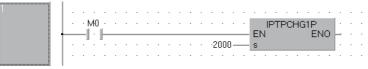
- IPTPCHG1(P): Axis 1
- IPTPCHG2(P): Axis 2

Control Data

| Device | Item | Setting data | Setting range | Setting side |
|--------|------------------------------|--------------|----------------|--------------|
| s +0 | | | -2147483648 to | |
| - | Target position change value | - | 2147483647 | User |
| s +1 | | | (pulse/s) | |

Program Example

The following program changes the target position of the axis 1 to 2000 when M0 turns ON. [Structured ladder/FBD]



[ST] IPTPCHG1P(M0, 2000);

8.2 Counter Function Dedicated Instruction

8.2.1 ICCNTRD instruction

ICCNTRD1, ICCNTRD2

LCPU

| ICCNTRD1(P) ICCNTRD2(P) | | P: Executing o | condition : 于 | |
|----------------------------|--|---|--|----------|
| Structured ladder/FBD | ST ENO:= ICCNTRD1 (EN); | indi instructions. ICCNTRD1 ICCNTRD2 | icates any of the follow ICCNTRD1F ICCNTRD2F | 5 |
| | euting condition eution result Internal device Bit Word | :Bit Bit J U∭\G∭ | Zn Constan | t Others |

Function

This instruction stores a value at the time of instruction execution to the current value of the specified CH (refer to the following).

| Instruction | CH | Device in which the current value is stored |
|-------------|-----|---|
| ICCNTRD1(P) | CH1 | SD1880, SD1881 |
| ICCNTRD2(P) | CH2 | SD1900, SD1901 |

Program Example

The following program stores the most recent value to the CH 1 current value (SD1880, SD1881) when M0 turns ON.

[Structured ladder/FBD]



[ST] ICCNTRD1(M0);

8.2 Counter Function Dedicated Instruction 8.2.1 ICCNTRD instruction

LCPU

8.2.2 ICRNGWR instruction

ICRNGWR1, ICRNGWR2

| ICRNGWR ICRNGWR | . , | P: Executing conditi | on : |
|--------------------|---|--|--|
| | Iadder/FBD ST GWR1 ENO ENO ENO:= ICRNGWR1 (EN, s1, | instructions. | y of the following ICRNGWR1P ICRNGWR2P |
| Input argument | EN: Executing condition s1: Ring counter lower limit value (constant), the device that stores the ring counter low • Constant: Settings which is within the ra | ver limit value ange of -2147483648 | |
| | to 2147483647 and ((s), (s) +1) ≦ • Device: Within the range of specified d s2: Ring counter upper limit value (constant), the device that stores the ring counter up • Constant: Settings which is within the range to 2147483647 and ((s), (s) +1) ≦ | evice or start number of :ANY32 per limit value ange of -2147483648 | |
| Output argument | • Device: Within the range of specified de ENO: Execution result | | |
| | Setting data Bit Word R, ZR | JIII\\III Bit Word UIII\GIII Zn − | Constant Othe |

Grant Function

This instruction sets the ring counter lower limit value and the ring counter upper limit value of the specified CH (refer to the following).

_

• ICRNGWR1(P): CH1

s2)

_

• ICRNGWR2(P): CH2

 \bigcirc

Program Example

The following program sets -100000 for the ring counter lower limit value and 100000 for the ring counter upper limit value of CH 1 when M0 turns ON.

[Structured ladder/FBD]

| 1 | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|----|-----|---|---|---|---|---|---|---|---|----|----|----|-----|---|-------|----|----|---|---|---|---|
| | | | ٠Ņ | vł0 | ÷ | | · | | | | | | | | | | | ICRNO | WR | 1P | | | | |
| | - | _ | | | ⊢ | | | _ | _ | _ | _ | _ | _ | | _ | _ | _ | EN | | ΕN | 0 | _ | ÷ | · |
| | | | | | | | | | | | | | | 00 | | - | | s1 | | | | · | · | · |
| | | | · | ÷ | ÷ | · | ÷ | · | ÷ | · | · | ÷ | ÷1 | 00 | 00 | 0 — | _ | s2 | | | | | · | · |
| | | | · | ÷ | ÷ | ÷ | ÷ | ÷ | ÷ | ÷ | · | ÷ | · | ÷ | ÷ | ÷ | · | | | | | | · | · |

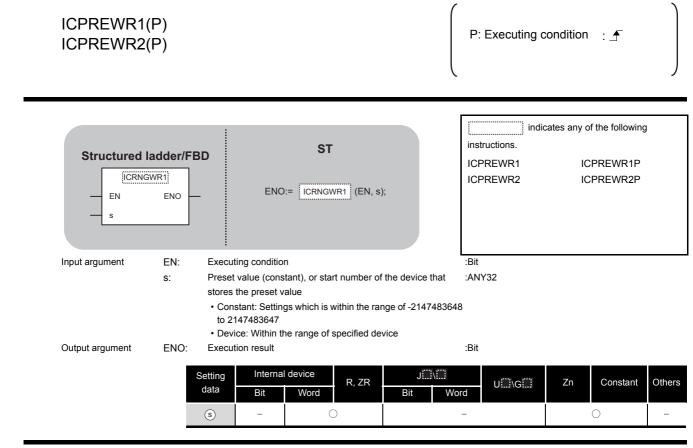
[ST] ICRNGWR1P(M0, -100000, 100000);

LCPU

IUILT-IN I/O FUNCTION

8.2.3 ICPREWR instruction

ICPREWR1, ICPREWR2



Grantion

This instruction sets a preset value of the specified CH (refer to the following).

- ICPREWR1(P): CH1
- ICPREWR2(P): CH2

Program Example

The following program sets 10000 for the preset value of CH 1 when M0 turns ON.

[Structured ladder/FBD]

| 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|----|---|---|----|----|---|---|---|---|---|---|---|---|---|---|----|----|----|---|---|-----|-----|----|-----|----|-----|---|---|
| | | | | M | 0_ | | | | | | | | | | | | | | | | 101 | PRI | E٧ | /R1 | | Ŀ | | |
| | -t | _ | - | 1: | ŀ | - | - | _ | _ | _ | | | | | _ | | | | _ | E | N. | | | E١ | 10 | F. | | · |
| | | · | | | | • | • | · | ÷ | ÷ | · | · | · | · | · | 10 | 00 | 0- | _ | s | | | | | | Ŀ., | · | · |
| | | • | | | | • | | · | • | | · | | · | · | • | · | | · | · | 1 | · | | • | • • | • | | • | • |

[ST] ICPREWR1(M0, 10000);

8.2.4 ICLTHRD instruction

ICLTHRD1, ICLTHRD2

LCPU

| ICLTHRD1(I ICLTHRD2(I | | | | | P: Executing c | condition | : 🖍 | |
|--------------------------------|------------|---|--------------------------|-------------------|--|-----------|---------------------------------------|------|
| Structured ICLTF EN n | | | ST ICLTHRD1 (EN, n, r | d); | indic instructions. ICLTHRD1 ICLTHRD2 | IC | f the following LTHRD1P LTHRD2P |) |
| Input argument | EN: | Executing condition | | | :Bit | | | |
| | n: | Latch count value (1 | ,2) | | :ANY16 | | | |
| Output argument | ENO: d: | Execution result Start number of the stored | device in which the | latch count value | :Bit is:ANY32 | | | |
| | | Setting Internal of data Bit | levice R, ZR Word | J\ Bit Wo | u | Zn | Constant | Othe |
| | | n – | 0 | | - | | 0 | - |
| | | - b | 0 | | | 0 | | |

Grant Function

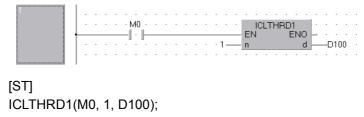
This instruction stores a latch count value n of the specified CH (refer to the following) to .

- ICLTHRD1(P): CH1
- ICLTHRD2(P): CH2

Program Example

The following program stores the latch count value 1 of CH 1 to D100 and D101 when M0 turns ON.

[Structured ladder/FBD]



8.2.5 ICSMPRD instruction

ICSMPRD1, ICSMPRD2

8

| ICSMPRD1(ICSMPRD2(| | P: Executing condition : |
|-----------------------------------|--|---|
| Structured ICSM EN | | indicates any of the following instructions. ICSMPRD1 ICSMPRD1P ICSMPRD2 ICSMPRD2P |
| Input argument Output argument | EN: Executing condition ENO: Execution result d: Start number of the device in which the sam is stored Setting Internal device R, ZR Bit Word R, ZR | :Bit :Bit pling count value;ANY32 JU::\G Bit Word U::\G |

Grant Function

This instruction stores a sampling count value of the specified CH (refer to the following) to .

- ICSMPRD1(P): CH1
- ICSMPRD2(P): CH2

Program Example

The following program stores the sampling count value of CH 1 to D100 and D101 when M0 turns ON.

[Structured ladder/FBD]

| 1 | . | | | | | | | | | | | | | | | | | | |
|---|---|---|--------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|-------|--|
| | | | м0 | | • | • | • | • | • | • | • | • | • | • | • | • | • | | |
| | | | | | | | | | | | | | | | | | | dD100 | |
| | · | • | | · | | ÷ | · | ÷ | ÷ | ÷ | 1 | · | | ÷ | · | 1 | ÷ | | |

[ST] ICSMPRD1(M0, D100);

8.2.6 ICCOVWR instruction

ICCOVWR1, ICCOVWR2

LCPU

| ICCOVWR1 ICCOVWR2 | · , | | | | P: Executing c | ondition : 于 | |
|-----------------------|-----------------|---|--------------------------|---|--|---|---------|
| Structured EN s | Iadder/FB | | ST ICCOVWR1 (EN, n, s | \$); | indic instructions. ICCOVWR1 ICCOVWR2 | ates any of the following ICCOVWR1P ICCOVWR2P | 3 |
| Input argument | EN: n: s: | the device in which • Constant: Setting to 2147483647 | | n point is stored ige of -2147483648 | | | |
| Output argument | ENO: | Execution result | Vord R, ZR | JINI Bit Wo | :Bit rd U::::\G::: - | Zn Constant | Oth |

Grant Function

This instruction stores a coincidence output No. n point of the specified CH (refer to the following).

- ICCOVWR1(P): CH1
- ICCOVWR2(P): CH2

Program Example

The following program sets the value of D100 and D101 to the coincidence output No. 2 point of CH 1 when M0 turns ON.

[Structured ladder/FBD]

[ST] ICCOVWR1(M0, 2, D100);

8.2.7 ICFCNT instruction

ICFCNT1, ICFCNT2

8

IN I/O FU

ICFCNT1 ICFCNT2

| Structured I | | ENO: | ST = ICFCNT1 (EN, d); | i, ir 10 | indic instructions. CFCNT1 CFCNT2 | cates any c | of the following | 3 |
|-----------------------------------|----------------|---|---------------------------------|---------------------------------|--|-------------|------------------|--------|
| Input argument Output argument | ENO: E d: S | Executing condition Execution result Start number of the value | n e device that stores the r | :B B neasured frequency:A | it | | | |
| | Sett da | | l device R, ZR Word | J∷i∖ii Bit Word | U∭\G∭ | Zn | Constant | Others |
| | d |) – | 0 | - | | 0 | - | |

Grant Function

This instruction measures a frequency of the specified CH (refer to the following) according to the settings such as the frequency measurement unit time setting.

- · ICFCNT1: CH1
- ICFCNT2: CH2

The measured value is stored to (a) at the ICFCNT instruction execution. The measurement starts at the rising pulse of the ICFCNT instruction execution command, and ends at the falling pulse.

Program Example

The following program executes the frequency measurement of CH 1 while M0 is ON.

[Structured ladder/FBD]

| 1 | . | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|-----|------|---|---|-----|---|---|---|---|---|---|---|---|---|---|------|----|---|----------|---|---------|----|---|---|
| | | - ÷ | ···- | | | • • | | | • | • | • | • | • | • | • | F | 101 | FC | | 1 INC | | | · | • | : |
| | | | - 1 | | | | | | | | | | | | | | IN . | | L | C | | D10 | 00 | | |
| | · | · | · | · | • | | · | · | · | · | · | · | 1 | · | ÷ | | ÷ | | • | • | • | ÷ | · | 1 | · |

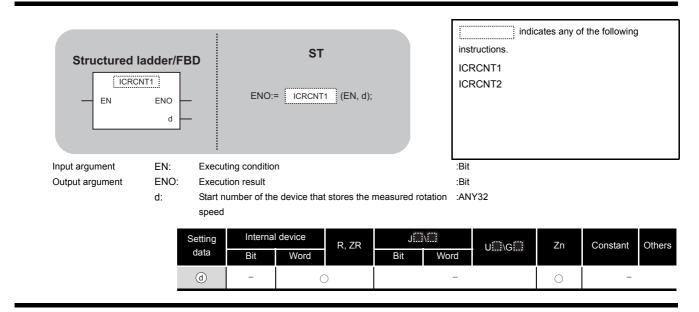
[ST] ICFCNT1(M0, D100);

8.2.8 ICRCNT instruction

ICRCNT1, ICRCNT2

LCPU

ICRCNT1 ICRCNT2



☆ Function

This instruction measures a rotation speed of the specified CH (refer to the following) according to the settings such as the rotation speed measurement unit time setting.

- ICRCNT1: CH1
- ICRCNT2: CH2

The measured value is stored to (a) at the ICRCNT instruction execution. The measurement starts at the rising pulse of the ICRCNT instruction execution command, and ends at the falling pulse.

Program Example

The following program stores the rotation speed measurement value of CH 1 to D100 and D101 while M0 is ON.

[Structured ladder/FBD]



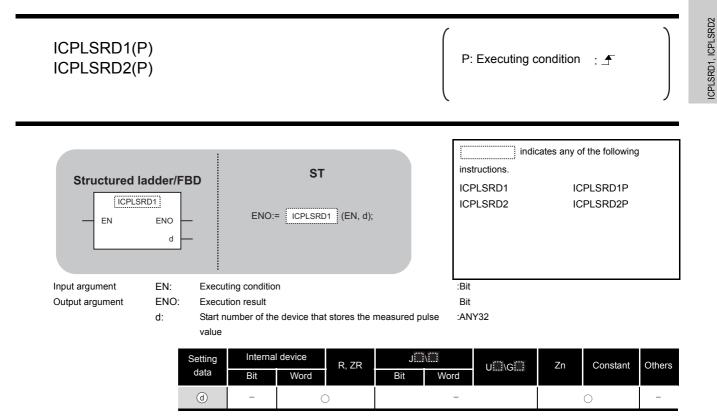
[ST] ICRCNT1(M0, D100);

LCPU

8.2.9 ICPLSRD instruction

ICPLSRD1, ICPLSRD2

8





This instruction stores a measured pulse value of the specified CH (refer to the following) to .

- · ICPLSRD1(P): CH1
- ICPLSRD2(P): CH2

Program Example

The following program stores the measured pulse value of CH 1 to D100 and D101 when M0 turns ON.

8.2 Counter Function Dedicated Instruction

8.2.9 ICPLSRD instruction

[Structured ladder/FBD]



[ST] ICPLSRD1(M0, D100);

8.2.10 ICPWM instruction

ICPWM1, ICPWM2

LCPU

ICPWM1 ICPWM2

| Structured EN s1 s2 | | | ST ICPWM1 (EN, s1, s | ICF | indic ructions. WM1 WM2 | cates any of t | he following | 1 |
|------------------------------|------|---|--------------------------------|----------------------------------|----------------------------------|----------------|--------------|------|
| Input argument | EN: | Executing condition | n | :Bit | | | | |
| | s1: | PWM output ON tin | me setting value (consta | ant), or start number :AN | (32 | | | |
| | | of the device that s | stores the PWM output 0 | ON time setting | | | | |
| | | value | | | | | | |
| | | Constant: Setting | gs which is 0 or within th | e range of 10 to 10 ⁷ | | | | |
| | | (0.1µs) and (🗊 | , ⑸ +1) ≦ (⑸ , ⑸ |) +1) | | | | |
| | | Device: Within the second | ne range of specified de | vice | | | | |
| | s2: | PWM output cycle | time setting value (cons | stant), or start :AN | (32 | | | |
| | | | ice that stores the PWM | output cycle time | | | | |
| | | setting value | | | | | | |
| | | Constant: Setting | gs which is 0 or within th | e range of 50 to 10 ⁷ | | | | |
| | | (0.1µs) and (🗊 | , (s1) +1) \leq ((s2) , (s2) |) +1) | | | | |
| | | Device: Within the second | ne range of specified de | vice | | | | |
| Output argument | ENO: | Execution result | | :Bit | | | | |
| | 9 | etting Internal | l device | JIII\III | | | | |
| | | data Bit | R, ZR | Bit Word | U\G | Zn | Constant | Othe |
| | _ | Ві | vvora | Bit Word | | | | 1 |
| | | s1 – | 0 | - | | C |) | - |
| | | | | | | | | |

Grant Function

This instruction outputs a PWM waveform of the specified CH (refer to the following).

- ICPWM1: CH1
- ICPWM2: CH2

The PWM waveform with the ON time ($_{\odot}$) and the cycle time ($_{\odot}$) is output from the coincidence output No.1 signal during the ICPWM instruction execution. The output of the PWM waveform starts from OFF.

Program Example

The following program outputs the PWM waveform with 1μ s ON time and 5μ s cycle time from CH 1 while M0 is ON.

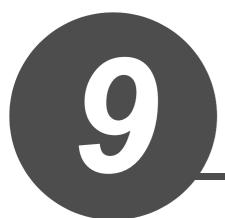
[Structured ladder/FBD]



[ST] ICPWM1(M0, 10, 50);

MEMO

| | |
|------|--|
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |



DATA LOGGING FUNCTION INSTRUCTION

| 9.1 | LOGTRG Instruction, LOGTRGR Instruction | 9-2 |
|-----|---|-----|
| | | |

9.1 LOGTRG Instruction, LOGTRGR Instruction

| | | LCPU |
|-----------------------------------|---|--|
| LOGTRG LOGTRGR | | Executing condition : _ |
| Structured I EN n | RG | indicates any of the following instructions. LOGTRG LOGTRGR |
| Input argument Output argument | EN:Executing conditionn:Data logging configuration numberENO:Execution result | :Bit :ANY16 :Bit |
| | Setting dataInternal deviceR,BitWordR, | JIIII UIIIGII Zn Constant Oth Bit Word UIIIGII Zn K, H Oth |

Grant Function

LOGTRG

- (1) The LOGTRG instruction generates a trigger in the trigger logging of the data logging configuration number specified by 'n'.
- (2) A value from 1 to 10 is set for 'n'.
- (3) When the LOGTRG instruction is executed, the special relay (data logging trigger) of the data logging configuration number specified by 'n' turns ON. After executing the trigger logging for the number of times set for "Number of records", the instruction latches the data and stops the trigger logging.
- (4) Validated when "When trigger instruction executed" is selected as the trigger condition.
- (5) No processing is performed with the following condition.
 - Specifying a data logging configuration number for which other than "When trigger instruction executed" is specified as the trigger condition.
 - Specifying a data logging configuration number which is not configured.
 - Specifying a data logging configuration number which is currently used for continuous logging.
 - Executing the LOGTRG instruction again without executing the LOGTRGR instruction after the LOGTRG instruction.

LOGTRGR

- (1) The LOGTRGR instruction resets the LOGTRG instruction of the specified data logging configuration number.
- (2) When the LOGTRGR instruction is executed, the special relay (data logging trigger, trigger logging complete) of the data logging configuration number specified by 'n' turns OFF.
- (3) When the instruction is executed while transferring data in the buffer memory to the SD memory card, the instruction process is held until data transfer is complete.

Operation Error

In the following case, an operation error occurs, the error flag (SM0) is turned ON, and the corresponding error code is stored to SD0.

• The value for n is outside the range of 1 to 10 (Error code: 4100)

Program Example

The following program executes the LOGTRG instruction on the data logging configuration No. 1 when X0 turns ON, and resets the trigger condition with the LOGTRGR instruction when X1 turns ON.

[Structured ladder/FBD]

| 1 | . | • | ×0 | | • | • | | | | | | • | | | | • | | | GTI | RG | | • | • | • |
|---|---|---|-----|----|---|---|---|---|---|---|---|---|---|---|---|-----|---|---------|-----|----------|---|---|---|---|
| | | | | | • | | • | • | • | • | • | • | • | • | • | 1 – | | EN n | | EN. | | | | • |
| 2 | | | | | | | | | | | | | | | | | | | | | • | | | • |
| | | | ×1. | ė. | • | • | • | • | • | • | • | • | • | • | • | • | • | LOG | TF | GP EN | | | ÷ | : |
| | | | ۰. | | | | | | | | | | | | | 1 – | | n | | | | | | |
| | · | | | | | | | | | | | | | | | · . | | | | | | | | |

[ST] LOGTRG(X0,1); LOGTRGR(X1,1);



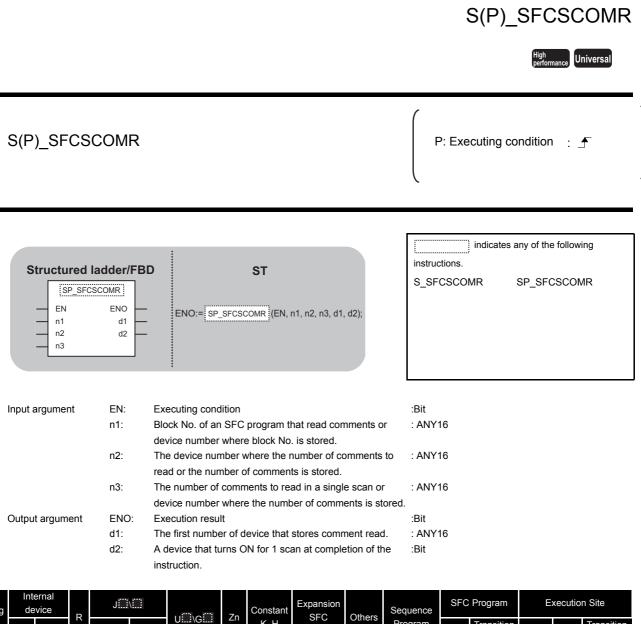
| | | |
|------|-------------------------|---------|
| 10.1 | SFC Control Instruction |)-2 |

10

INDEX

10.1 SFC Control Instruction

10.1.1 SFCSCOMR instruction



| Setting data | Internal device | | | | \ | 213 213 | Zn | Constant | t Expansion SFC | Others | Sequence | SFC Program | | Execution Site | | |
|-----------------|----------------------------------|------|---|-----|------|---------|-----|----------|--------------------|---------|----------|-------------|-------------------------|----------------|------|-------------------------|
| | Bit | Word | R | Bit | Word | U\G | 211 | К, Н | K, H BLm\Sn | Outlets | Program | Sten | Transition Condition | Block | Step | Transition Condition |
| n1 | - | 0 | | | | - | | 0 | - | | | | - | | - | _ |
| n2 | - | 0 | | | | _ | | 0 | - | | | - | | - | - | |
| n3 | - | 0 | | - | | _ | | 0 | - | | 0 | | - | 0 | - | - |
| d1 | - | O*′ | 1 | | | _ | | - | - | | | | - | | - | - |
| d2 | O*1 | _ | | | | _ | | - | - | | | | - | | - | - |
| | *1: Local device cannot be used. | | | | | | | | | | | | | | | |

☆ Function

This function reads step comments being activated in the SFC block specified at , by the number of comment specified at , and stores those to the device number of after specified at .

Caution

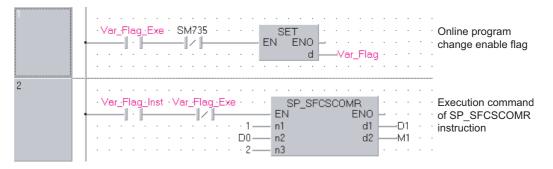
For High Performance model QCPU, use the function version is B or later and the first five digits of the serial number are '07012' or higher.

Program Example

This program reads 2 comments being activated at the SFC block No.1 when X1 is turned ON, and stores those to the storage device after D0.

(The number of comment to be read in a single scan is also set in 2.)

[Structured ladder/FBD]



[ST]

IF((Var_Flag_Exe=TRUE) & (SM735=FALSE))THEN

(*Online program change execution command*) SET(TRUE, Var_Flag); (*Online program change enable flag*) END_IF; IF((Var_Flag_Inst=TRUE) & (Var_Flag=FALSE))THEN (*Execution command of SP_SFCSCOMR instruction*) SP_SFCSCOMR(TRUE, 1, D0, 2, D1, M1); END_IF;

10.1.2 SFCTCOMR instruction

S(P)_SFCTCOMR

^h formance Universal

S(P)_SFCTCOMR P: Executing condition : indicates any of the following instructions. Structured ladder/FBD ST S_SFCTCOMR SP_SFCTCOMR SP_SFCTCOMR ΕN ENO ENO:= SP_SFCTCOMR (EN, n1, n2, n3, d1, d2); n1 d1 n2 d2 n3 Input argument EN: Executing condition :Bit n1: Block No. of an SFC program that read comments or : ANY16 device number where block No. is stored. n2: The device number where the number of comments to : ANY16 read or the number of comments is stored. n3: : ANY16 The number of comments to read in a single scan or device number where the number of comments is stored. Output argument ENO: Execution result :Bit d1: The first number of device that stores comment read. : ANY16 d2: A device that turns ON for 1 scan at completion of the :Bit instruction.

| Setting | | Internal device | | J | | . Jacob - Jacob | 7 | Constant | nt Expansion | Others | Sequence | SFC Program | | Execution Site | | n Site |
|---------|----------------------------------|-----------------|---|-----|------|-----------------|----|----------|---------------|--------|----------|-------------|-------------------------|----------------|------|-------------------------|
| data | Bit | Word | R | Bit | Word | U\G | Zn | К, Н | SFC BLm\Sn | Others | Program | Step | Transition Condition | Block | Step | Transition Condition |
| n1 | - | 0 | 0 | | _ | | 0 | - | | | - | | | _ | - | |
| n2 | - | 0 | | | | - | | 0 | - | | | | - | | - | - |
| n3 | - | 0 | | | | _ | | 0 | _ | | 0 | | - | 0 | - | - |
| d1 | - | 0*1 | 1 | | | _ | | | - | | | | - | | - | - |
| d2) | O*1 | - | | | | _ | | - | _ | | | | - | | - | _ |
| | *1: Local device cannot be used. | | | | | | | | | | | | | | | |

Grant Function

This function reads comments of the transition condition 1 associated with steps activated in the SFC block specified at with the number of comments specified at , and stores those to the device number of after specified at .

Caution

For High Performance model QCPU, use the function version is B or later and the first five digits of the serial number are '07012' or higher.

Program Example

This program reads 2 comments being activated at the SFC block No.1 when X1 is turned ON, and stores those to the storage device after D0.

(The number of comment to be read
in a single scan is also set in 2.)

[Structured ladder/FBD]

| 1 | ·Var_Flag_Exe SM735 SET Change enable flag □ □ □ □ □ □ □ □ □ |
|---|---|
| 2 | · Var_Flag.Inst · Var_Flag.···· SP_SFCTCOMR Execution command · · · · · · · · · · · · · · · · · · · |

[ST]

IF((Var_Flag_Exe=TRUE) & (SM735=FALSE))THEN

(*Online program change execution command*) (*Online program change enable flag*)

END_IF;

SET(TRUE, Var_Flag);

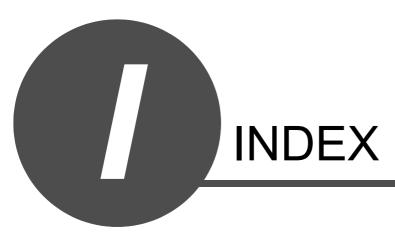
IF((Var_Flag_Inst=TRUE) & (Var_Flag=FALSE))THEN

(*Execution command of SP_SFCTCOMR instruction*) SP_SFCTCOMR(TRUE, 1, D0, 2, D1, M1); END_IF; S(P)_SFCTCOMR

10-5

MEMO

| |
|------|
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |





[A]

| ABRST (absolute position restoration) | 5-51 |
|---------------------------------------|------|
| Analog instruction | |
| Analog module | 1-8 |
| | |

[B]

| BIDIN (receiving data using bidirectional protocol |
|---|
| communication) |
| BIDOUT (sending data using bidirectional protocol |
| communication)5-74 |
| BUFRCV (receiving data in fixed buffer |
| communication) 5-225 |
| BUFRCVS (receiving data with interrupt |
| program)5-83,5-229 |
| BUFSND (sending data in fixed buffer communication) |
| |

[C]

| CC-Link IE Controller Network module 1-8,1-9 |
|---|
| CC-Link system master/local module1-8 |
| CLOSE (connection closing) |
| Configuration of Instructions 3-2 |
| CPRTCL (Pre-defined protocol communication) |
| |
| CSET (initial setting) |
| CSET (programmable controller CPU monitoring) |
| |
| CSET (receive data clear)5-80 |
| וחז |

[D]

| Destination | 3-3 |
|-------------|-----|
| [E] | |

| 6-1 | |
|-------------------------------------|-------|
| ERRCLR (clearing error information) | 5-235 |
| ERRRD (reading error information) | 5-238 |
| Ethernet interface module | 1-9 |
| | |

[G]

| GETE (user frame reading) | 5-104 |
|--|-------|
| [H] | |
| HOW TO READ INSTRUCTIONS | 4-1 |
| [1] | |
| I/O number | 3-2 |
| I/O number of module | 3-2 |
| ICCNTRD (current value read) | 8-18 |
| ICCOVWR (coincidence output point write) | 8-24 |
| ICFCNT (frequency measurement) | 8-25 |
| ICLTHRD (latch counter value read) | 8-22 |
| ICPLSRD (pulse measurement read) | 8-27 |

| CRNGWR (ring counter upper/lower limit value write) | |
|---|---|
| | 9 |
| CSMPRD (sampling counter value read) | |
| NPUT (receiving data using nonprocedural protocol | |
| communication)5-7 | 1 |
| NSTRUCTION TABLES 2- | 1 |
| PABRST (absolute position restoration) | 1 |
| PDSTRT (positioning start)8-3 | 3 |
| PJOG (JOG start)8-9 | 9 |
| POPR (OPR start)8- | 7 |
| PPSTRT (table start)8-2 | 2 |
| PSIMUL (2 axes simultaneous start)8-6 | 6 |
| PSPCHG (speed change) 8-14 | 4 |
| PSTOP (axis stop)8-13 | 3 |
| PTPCHG (target position change)8-16 | 6 |

[L]

| LOGTRG (trigger logging set) | 9-2 |
|---------------------------------|-----|
| LOGTRGR (trigger logging reset) | 9-2 |

[M]

| MELSECNET/H network module | 1-9 |
|---|-------|
| Modules and Versions Applicable to Instructions | 1-8 |
| MRECV (receiving e-mail) | 5-245 |
| MSEND (sending e-mail) | 5-250 |

[N]

| 2 |
|---|
| - |

[0]

| OFFGAN (mode switching) | 5-2 |
|--|--------|
| OGLOAD (reading offset/gain setting value) | 5-4 |
| OGSTOR (restoring offset/gain setting value) | 5-27 |
| ONDEMAND (sending data using the on-demar | nd |
| function) | 5-64 |
| OPEN (connection opening) | 5-218 |
| OUTPUT (sending data using nonprocedural pr | otocol |
| communication) | 5-68 |
| | |

[P]

| PFWRT (writing data to flash ROM) | 5-60 |
|--|-----------|
| PID control instruction | |
| PIDCONT (PID operation) | 6-7,6-21 |
| PIDINIT (setting data to be used for PID | |
| operation) | 6-2,6-16 |
| PIDPRMW (changing parameter for PID | |
| operation) | 6-12,6-27 |
| PIDRUN (PID operation start) | 6-11,6-26 |
| PIDSTOP (PID operation stop) | 6-11,6-26 |
| PINIT (setting data initialization) | 5-62 |
| Positioning Instruction | 5-51 |
| PRR (data transmission/reception) | 5-85 |
| PSTRT (positioning start) | 5-55 |
| | |

| PUTE (user frame registration) | 5-101 |
|--------------------------------|-------|
|--------------------------------|-------|

[R]

| F3 |
|---|
| READ (reading data from a word device) 5-144 |
| RECV (receiving data) 5-173 |
| RECVS (receiving data) 5-178 |
| Related manuals A-10 |
| REMFR (reading data from buffer memory) 5-208 |
| REMTO (writing data to buffer memory) 5-210 |
| REQ (transient request) 5-181 |
| RIFR (reading data from auto-refresh buffer memory) |
| |
| RIRCV (reading data from buffer memory) 5-125 |
| RIRD (reading data) 5-115 |
| RISEND (writing data to buffer memory) 5-129 |
| RITO (writing data to auto-refresh buffer memory) |
| |
| RIWT (writing data) 5-120 |
| RLPASET (parameter setting) 5-137 |
| RRUN (remote RUN) 5-197 |
| RSTOP (remote STOP) 5-200 |
| RTMRD (reading clock data) 5-203 |
| RTMWR (writing clock data) 5-205 |
| |

[S]

| SEND (sending data) | 5-165 |
|--|-------|
| Serial communication instruction | 2-5 |
| Serial communication module | 1-8 |
| SFCSCOMR instruction | 10-2 |
| SFCTCOMR instruction | 10-4 |
| SOCCINF (reading connection information) | 7-16 |
| SOCCLOSE (shutting off a connection) | 7-5 |
| SOCCSET (changing connection target) | 7-19 |
| SOCOPEN (opening a connection) | 7-2 |
| SOCRCV (reading receive data) | 7-8 |
| SOCRCVS (reading receive data) | 7-11 |
| SOCRDATA (reading data from the socket | |
| communication receive data area) | 7-24 |
| SOCRMODE (changing receive mode) | 7-22 |
| SOCSND (sending data) | 7-13 |
| Source | 3-3 |
| SPBUSY (communication status check) | 5-79 |
| SREAD (reading data from a word device) | 5-150 |
| SWRITE (writing data to a word device) | 5-161 |
| [T] | |
| TEACH (teaching) | 5-57 |
| | |

[U]

| UINI (re-initialization) | 5-241 |
|--|--------|
| UINI (switching the mode, transmission specifica | ation, |
| and host station number) | 5-107 |

[W]

| WRITE (writing data to a word device)5-154 |
|--|
| [Z] |
| ZNWR 5-193 |

MEMO

| |
|------|
| |
| |

<u>WARRANTY</u>

Please confirm the following product warranty details before using this product.

1. Gratis Warranty Term and Gratis Warranty Range

If any faults or defects (hereinafter "Failure") found to be the responsibility of Mitsubishi occurs during use of the product within the gratis warranty term, the product shall be repaired at no cost via the sales representative or Mitsubishi Service Company. However, if repairs are required onsite at domestic or overseas location, expenses to send an engineer will be solely at the customer's discretion. Mitsubishi shall not be held responsible for any re-commissioning, maintenance, or testing onsite that involves replacement of the failed module.

[Gratis Warranty Term]

The gratis warranty term of the product shall be for one year after the date of purchase or delivery to a designated place. Note that after manufacture and shipment from Mitsubishi, the maximum distribution period shall be six (6) months, and the longest gratis warranty term after manufacturing shall be eighteen (18) months. The gratis warranty term of repair parts shall not exceed the gratis warranty term before repairs.

[Gratis Warranty Range]

- (1) The range shall be limited to normal use within the usage state, usage methods and usage environment, etc., which follow the conditions and precautions, etc., given in the instruction manual, user's manual and caution labels on the product.
- (2) Even within the gratis warranty term, repairs shall be charged for in the following cases.
 1. Failure occurring from inappropriate storage or handling, carelessness or negligence by the user. Failure caused by the user's hardware or software design.
 - 2. Failure caused by unapproved modifications, etc., to the product by the user.
 - 3. When the Mitsubishi product is assembled into a user's device, Failure that could have been avoided if functions or structures, judged as necessary in the legal safety measures the user's device is subject to or as necessary by industry standards, had been provided.
 - 4. Failure that could have been avoided if consumable parts (battery, backlight, fuse, etc.) designated in the instruction manual had been correctly serviced or replaced.
 - 5. Failure caused by external irresistible forces such as fires or abnormal voltages, and Failure caused by force majeure such as earthquakes, lightning, wind and water damage.
 - 6. Failure caused by reasons unpredictable by scientific technology standards at time of shipment from Mitsubishi.
 - 7. Any other failure found not to be the responsibility of Mitsubishi or that admitted not to be so by the user.

2. Onerous repair term after discontinuation of production

- (1) Mitsubishi shall accept onerous product repairs for seven (7) years after production of the product is discontinued. Discontinuation of production shall be notified with Mitsubishi Technical Bulletins, etc.
- (2) Product supply (including repair parts) is not available after production is discontinued.

3. Overseas service

Overseas, repairs shall be accepted by Mitsubishi's local overseas FA Center. Note that the repair conditions at each FA Center may differ.

4. Exclusion of loss in opportunity and secondary loss from warranty liability

Regardless of the gratis warranty term, Mitsubishi shall not be liable for compensation to damages caused by any cause found not to be the responsibility of Mitsubishi, loss in opportunity, lost profits incurred to the user by Failures of Mitsubishi products, special damages and secondary damages whether foreseeable or not, compensation for accidents, and compensation for damages to products other than Mitsubishi products, replacement by the user, maintenance of on-site equipment, start-up test run and other tasks.

5. Changes in product specifications

The specifications given in the catalogs, manuals or technical documents are subject to change without prior notice.

Microsoft, Windows are registered trademarks of Microsoft Corporation in the United States and other countries.

Ethernet is a trademark of Xerox Corporation.

All other company names and product names used in this manual are trademarks or registered trademarks of their respective companies.

MELSEC-Q/L Structured Programming Manual

Special Instructions

Q-KP-TM-E

MODEL

MODEL CODE

13JW09

SH(NA)-080785ENG-I(1107)KWIX

MITSUBISHI ELECTRIC CORPORATION

HEAD OFFICE : TOKYO BUILDING, 2-7-3 MARUNOUCHI, CHIYODA-KU, TOKYO 100-8310, JAPAN NAGOYA WORKS : 1-14 , YADA-MINAMI 5-CHOME , HIGASHI-KU, NAGOYA , JAPAN

When exported from Japan, this manual does not require application to the Ministry of Economy, Trade and Industry for service transaction permission.



| HEADQUARTERS | |
|--|-----------|
| MITSUBISHI ELECTRIC EUROPE B.V. German Branch Gothaer Straße 8 | EUROPE |
| D-40880 Ratingen Phone: +49 (0)2102 / 486-0 Fax: +49 (0)2102 / 486-1120 | |
| MITSUBISHI ELECTRIC EUROPE B.Vorg.sl. | ZECH REP. |
| Czech Branch Avenir Business Park, Radlická 714/113a CZ-158 00 Praha 5 Phone: +420 - 251 551 470 Fax: +420 - 251-551-471 | |
| MITSUBISHI ELECTRIC EUROPE B.V. French Branch 25, Boulevard des Bouvets F-92741 Nanterre Cedex Phone: +33 (0)1 / 55 68 55 68 | FRANCE |
| Finite: +33 (0)1 / 55 68 57 57 | |
| MITSUBISHI ELECTRIC EUROPE B.V. Irish Branch Westgate Business Park, Ballymount IRL-Dublin 24 Phone: +353 (0)1 4198800 Fax: +353 (0)1 4198890 | IRELAND |
| MITSUBISHI ELECTRIC EUROPE B.V. Italian Branch Viale Colleoni 7 I-20041 Agrate Brianza (MB) Phone: +39 039 / 60 53 1 Fax: +39 039 / 60 53 312 | ITALY |
| MITSUBISHI ELECTRIC EUROPE B.V. Poland Branch Krakowska 50 PL-32-083 Balice Phone: +48 (0)12 / 630 47 00 Fax: +48 (0)12 / 630 47 01 | POLAND |
| MITSUBISHI ELECTRIC EUROPE B.V. 52, bld. 3 Kosmodamianskaya nab 8 floor RU-115054 Moscow Phone: +7 495 721-2070 Fax: +7 495 721-2071 | RUSSIA |
| MITSUBISHI ELECTRIC EUROPE B.V. | SPAIN |
| Spanish Branch Carretera de Rubí 76-80 E-08190 Sant Cugat del Vallés (Barce Phone: 902 131121 // +34 935653131 Fax: +34 935891579 | lona) |
| MITSUBISHI ELECTRIC EUROPE B.V. UK Branch Travellers Lane UK-Hatfield, Herts. AL10 8XB Phone: +44 (0)1707 / 27 61 00 | UK |
| Fax: +44 (0) 1707 / 27 86 95 MITSUBISHI ELECTRIC CORPORATION Office Tower "Z" 14 F | JAPAN |
| 8-12,1 chome, Harumi Chuo-Ku Tokyo 104-6212 Phone: +81 3 622 160 60 | |
| FaX: +813 622 160 75 | |
| Fax: +81 3 622 160 75 MITSUBISHI ELECTRIC AUTOMATION, Inc. 500 Corporate Woods Parkway Vernon Hills, IL 60061 Phone: +1 847 478 21 00 | USA |

| EUROPEAN REPRESENT | ATIVES |
|--|---|
| GEVA | AUSTRIA |
| Wiener Straße 89 AT-2500 Baden | |
| Phone: +43 (0)2252 / 85 55 20 | |
| Fax: +43 (0)2252 / 488 60 | |
| TECHNIKON | BELARUS |
| Oktyabrskaya 19, Off. 705 | |
| BY-220030 Minsk | |
| Phone: +375 (0)17 / 210 46 26 | |
| Fax: +375 (0)17 / 210 46 26 | |
| ESCO DRIVES & AUTOMATION Culliganlaan 3 | BELGIUM |
| BE-1831 Diegem | |
| Phone: +32 (0)2 / 717 64 30 | |
| Fax: +32 (0)2 / 717 64 31 | |
| Koning & Hartman b.v. | BELGIUM |
| Woluwelaan 31 | |
| BE-1800 Vilvoorde | |
| Phone: +32 (0)2 / 257 02 40 Fax: +32 (0)2 / 257 02 49 | |
| INEA RBT d.o.o. BOSNIA AND H | |
| Aleja Lipa 56 | ENZEGUVINA |
| BA-71000 Sarajevo | |
| Phone: +387 (0)33 / 921 164 | |
| Fax: +387 (0)33/ 524 539 | |
| AKHNATON | BULGARIA |
| 4, Andrei Ljapchev Blvd., PO Box 21 | |
| BG-1756 Sofia | |
| Phone: +359 (0)2 / 817 6000 Fax: +359 (0)2 / 97 44 06 1 | |
| | CDOATIA |
| INEA RBT d.o.o. Losinjska 4 a | CROATIA |
| HR-10000 Zagreb | |
| Phone: +385 (0)1/36940-01/-02/-03 | |
| Fax: +385 (0)1 / 36 940 - 03 | |
| | CH REPUBLIC |
| Technologická 374/6 | |
| CZ-708 00 Ostrava-Pustkovec Phone: +420 595 691 150 | |
| | |
| | |
| Fax: +420 595 691 199 | DENMARK |
| Fax: +420 595 691 199 Beijer Electronics A/S | DENMARK |
| Fax: +420 595 691 199 | DENMARK |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 | DENMARK |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 56 26 | |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 56 26 Beijer Electronics Eesti OÜ | DENMARK |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 56 26 Beijer Electronics Eesti OÜ Pärnu mnt.160i | |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/75 76 66 Fax: +45 (0)46/75 56 26 Beijer Electronics Eesti OÜ Pärnu mnt.160i EE-11317 Tallinn | |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 56 26 Beijer Electronics Eesti OÜ Pärnu mnt.160i EE-11317 Tallinn Phone: +372 (0)6/ 51 81 40 | |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 56 26 Beijer Electronics Eesti OÜ Pärnu mnt.160i EE-11317 Tallinn Phone: +372 (0)6/ 51 81 40 Fax: +372 (0)6/ 51 81 49 | |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 56 26 Beijer Electronics Eesti OÜ Pärnu mnt.160i EE-11317 Tallinn Phone: +372 (0)6/ 51 81 40 | ESTONIA |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/75 76 66 Fax: +457 (0)46/75 76 66 Fax: +472 (0)46/75 81 40 Fax: +372 (0)6/51 81 40 Fax: +372 (0)6/51 81 49 Beijer Electronics OY Peltoie 37 FIN-28400 Ulvila | ESTONIA |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 56 26 Beijer Electronics Eesti OÜ Pärnu mnt.160i EE-11317 Tallinn Phone: +372 (0)6 / 51 81 40 Fax: +372 (0)6 / 51 81 49 Beijer Electronics OY Peltoie 37 FIN-28400 Ulvila Phone: +358 (0)207 / 463 540 | ESTONIA |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 76 66 Beijer Electronics Eesti OÜ Pärnu mnt.160i EE-11317 Tallinn Phone: +372 (0)6 / 51 81 40 Fax: +372 (0)6 / 51 81 49 Beijer Electronics OY Peltoie 37 FIN-28400 Ulvila Phone: +358 (0)207 / 463 540 Fax: +358 (0)207 / 463 541 | ESTONIA |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 56 26 Beijer Electronics Eesti OÜ Pärnu mnt.160i EE-11317 Tallinn Phone: +372 (0)6/ 51 81 40 Fax: +372 (0)6/ 51 81 49 Beijer Electronics OY Peltoie 37 FIN-28400 Ulvila Phone: +358 (0)207 / 463 540 Fax: +358 (0)207 / 463 541 UTEC0 | ESTONIA |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/75 76 66 Fax: +45 (0)46/75 56 26 Beijer Electronics Eesti OÜ Pärnu mnt.160i EE-11317 Tallinn Phone: +372 (0)6/51 81 40 Fax: +372 (0)6/51 81 49 Beijer Electronics OY Peltoie 37 FIN-28400 Ulvila Phone: +358 (0)207 / 463 540 Fax: +358 (0)207 / 463 541 UTECO 5, Mavrogenous Str. | ESTONIA |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 56 26 Beijer Electronics Eesti OÜ Pärnu mnt.160i EE-11317 Tallinn Phone: +372 (0)6/ 51 81 40 Fax: +372 (0)6/ 51 81 49 Beijer Electronics OY Peltoie 37 FIN-28400 Ulvila Phone: +358 (0)207 / 463 540 Fax: +358 (0)207 / 463 541 UTEC0 | ESTONIA |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 56 26 Beijer Electronics Eesti OÜ Pärnu mnt.160i EE-11317 Tallinn Phone: +372 (0)6 / 51 81 40 Fax: +372 (0)6 / 51 81 40 Fax: +372 (0)6 / 51 81 49 Beijer Electronics OY Peltoia 37 FIN-28400 Ulvila Phone: +358 (0)207 / 463 540 Fax: +358 (0)207 / 463 541 UTECO 5, Mavrogenous Str. GR-18542 Piraeus Phone: +30 211 / 1206 900 | ESTONIA |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 Fax: +477 70 75 76 66 Fax: +477 70 75 76 66 Fax: +477 76 76 76 76 76 76 76 76 76 76 76 76 7 | ESTONIA |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/75 76 66 Fax: +45 (0)46/75 76 66 Fax: +45 (0)46/75 56 26 Beijer Electronics Eesti 0Ü Pärnu mnt.160i EE-11317 Tallinn Phone: +372 (0)6/51 81 40 Fax: +372 (0)6/51 81 49 Beijer Electronics OY Peltoie 37 FIN-28400 Ulvila Phone: +358 (0)207 / 463 540 Fax: +358 (0)207 / 463 541 UTEC0 5, Mavrogenous Str. GR-18542 Piraeus Phone: +30 211 / 1206 900 Fax: +30 211 / 1206 999 MELTRADE Kft. Fertő utca 14. | ESTONIA FINLAND GREECE |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 78 60 Fax: +45 (0)46/ 75 78 60 Fax: +372 (0)6/ 51 81 40 Fax: +372 (0) | ESTONIA FINLAND GREECE |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 76 66 Beijer Electronics Eesti OÜ Pärnu mnt.160i EE-11317 Tallinn Phone: +372 (0)6 / 51 81 40 Fax: +372 (0)6 / 51 81 40 Fax: +372 (0)6 / 51 81 49 Beijer Electronics OY Peltoia 37 FIN-28400 Ulvila Phone: +358 (0)207 / 463 540 Fax: +358 (0)207 / 463 541 UTECO 5, Mavrogenous Str. GR-18542 Piraeus Phone: +30 211 / 1206 900 Fax: +30 211 / 1206 999 MELTRADE Kft. Fertő utca 14. HU-1107 Budapest Phone: +36 (0)1 / 431-9726 | ESTONIA FINLAND GREECE |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 76 66 Beijer Electronics Eesti OÜ Pärnu mnt.160i EE-11317 Tallinn Phone: +372 (0)6 / 51 81 40 Fax: +372 (0)6 / 51 81 49 Beijer Electronics OY Peltois 37 FIN-28400 Ulvila Phone: +358 (0)207 / 463 540 Fax: +358 (0)207 / 463 541 UTECO S, Mavrogenous Str. GR-18542 Piraeus Phone: +30 211 / 1206 900 Fax: +30 211 / 1206 909 MELTRADE Kft. Fertő utca 14. HU-1107 Budapest Phone: +36 (0)1 / 431-9726 Fax: +36 (0)1 / 431-9727 | ESTONIA FINLAND GREECE HUNGARY |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 76 66 Beijer Electronics Eesti OÜ Pärnu mnt.160i EE-11317 Tallinn Phone: +372 (0)6/ 51 81 40 Fax: +372 (0)6/ 51 81 40 Fax: +372 (0)6/ 51 81 49 Beijer Electronics OY Peltoie 37 FIN-28400 Ulvila Phone: +358 (0)207 / 463 540 Fax: +358 (0)207 / 463 541 UTECO 5, Mavrogenous Str. GR-18542 Piraeus Phone: +30 211 / 1206 900 Fax: +30 211 / 1206 999 MELTRADE Kft. Fertő utca 14. HU-107 Budapest Phone: +36 (0)1 / 431-9726 Fax: +36 (0)1 / 431-9727 Beijer Electronics SIA | ESTONIA FINLAND GREECE |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 76 66 Beijer Electronics Eesti OÜ Pärnu mnt.160i EE-11317 Tallinn Phone: +372 (0)6/ 51 81 40 Fax: +338 (0)207 / 463 540 Fax: +338 (0)207 / 463 541 UTECO 5, Mavrogenous Str. GR-18542 Piraeus Phone: +30 211 / 1206 900 Fax: +30 (0) 1 / 431-9726 Fax: +36 (0) 1 / 431-9727 Beijer Electronics SIA Ritausmas iela 23 | ESTONIA FINLAND GREECE HUNGARY |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 76 66 Beijer Electronics Eesti OÜ Parnu mnt.160i EE-11317 Tallinn Phone: +372 (0)6 / 51 81 40 Fax: +372 (0)7 / 463 540 Fax: +358 (0)207 / 463 540 Fax: +358 (0)207 / 463 541 UTECO 5, Mavrogenous Str. GR-18542 Piraeus Phone: +30 211 / 1206 900 Fax: +30 211 / 1206 900 Fax: +30 211 / 1206 999 MELTRADE Kft. Fertő utca 14. HU-1107 Budapest Phone: +35 (0)1 / 431-9727 Beijer Electronics SIA Ritausmas iela 23 LV-1058 Riga Phone: +371 (0)784 / 2280 | ESTONIA FINLAND GREECE HUNGARY |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 76 66 Eax: +45 (0)46/ 75 76 66 Eax: +45 (0)46/ 75 76 66 Eijer Electronics Eesti OÜ Parnu mnt.1601 EE-11317 Tallinn Phone: +372 (0)6 / 51 81 40 Fax: +38 (0)207 / 463 540 Fax: +38 (0)207 / 463 540 Fax: +38 (0)207 / 463 541 UTECO 5, Mavrogenous Str. GR-18542 Piraeus Phone: +30 211 / 1206 900 Fax: +30 211 / 1206 910 MELTRADE Kft. Fertő utca 14. HU-1107 Budapest Phone: +36 (0)1 / 431-9727 Beijer Electronics SIA Ritausmas iela 23 LV-1058 Riga | ESTONIA FINLAND GREECE HUNGARY |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 78 64 Fax: +372 (0)6/ 51 81 40 Fax: +372 (0)7 / 463 540 Fax: +358 (0)207 / 463 540 Fax: +36 (0)27 / 463 541 UTECO 5, Mavrogenous Str. GR-18542 Piraeus Phone: +35 (0)1 / 431-9727 Beijer Electronics SIA Ritausmas iela 23 LV-1058 Riga Phone: +371 (0)784 / 2280 | ESTONIA FINLAND GREECE HUNGARY |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/75 76 66 Fax: +45 (0)46/75 78 140 Fax: +372 (0)6/51 81 40 Fax: +372 (0)6/7 463 540 Fax: +386 (0)207 / 463 540 Fax: +386 (0)207 / 463 540 Fax: +386 (0)207 / 463 541 UTECO 5, Mavrogenous Str. GR-18542 Piraeus Phone: +30 (2017 / 403 541 UTECO Fax: +30 (0)1 / 431-9726 Fax: +36 (0)1 / 431-9727 Beijer Electronics SIA Ritausmas iela 23 LV-1058 Riga Phone: +371 (0)784 / 2280 Fax: +371 (0)7 | ESTONIA FINLAND GREECE HUNGARY LATVIA |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 76 66 Fax: +45 (0)46/ 75 76 66 Fax: +372 (0)6 / 51 81 40 Fax: +372 (0)7 / 463 540 Fax: +388 (0)207 / 463 540 Fax: +388 (0)207 / 463 541 UTECO 5, Mavrogenous Str. GR-18542 Piraeus Phone: +30 (2)1 / 1206 900 Fax: +30 211 / 1206 900 Fax: +30 201 / 431-9727 Fax: | ESTONIA FINLAND GREECE HUNGARY LATVIA |
| Fax: +420 595 691 199 Beijer Electronics A/S Lykkegårdsvej 17 DK-4000 Roskilde Phone: +45 (0)46/75 76 66 Fax: +477 Tallinn Phone: +372 (0)6/51 81 40 Fax: +372 (0)207 / 463 540 Fax: +386 (0)207 / 463 540 Fax: +386 (0)207 / 463 541 UTECO 5, Mavrogenous Str. GR-18542 Piraeus Phone: +30 (2017 / 463 541 UTECO 5, Mavrogenous Str. GR-18542 Piraeus Phone: +30 (2017 / 463 541 UTECO 5, Mavrogenous Str. GR-18542 Piraeus Phone: +30 (2017 / 431-9727 Beijer Electronics SIA Ritausmas iela 23 LV-1058 Riga Phone: +371 (0)784 / 2280 Fax: +371 (0)784 / 2280 Fax | ESTONIA FINLAND GREECE HUNGARY LATVIA |

EUROPEAN REPRESENTATIVES ALFATRADE Ltd. MALTA 99, Paola Hill Malta- Paola PLA 1702 Phone: +356 (0)21 / 697 816 Fax: +356 (0)21 / 697 817 INTEHSIS srl MOLDOVA bld. Traian 23/1 MD-2060 Kishinev Phone: +373 (0)22 / 66 4242 Fax: +373 (0)22 / 66 4280 HIFLEX AUTOM.TECHNIEK B.V. NETHERLANDS Wolweverstraat 22 NL-2984 CD Ridderkerk Phone: +31 (0)180 - 46 60 04 Fax: +31 (0)180 - 44 23 55 Koning & Hartman b.v. NETHERLANDS Haarlerbergweg 21-23 NL-1101 CH Amsterdam Phone: +31 (0)20 / 587 76 00 Fax: +31 (0)20 / 587 76 05 **Beijer Electronics AS** NORWAY Postboks 487 NO-3002 Drammen Phone: +47 (0)32 / 24 30 00 Fax: +47 (0)32 / 84 85 77 Fonseca S.A. PORTUGAL R. João Francisco do Casal 87/89 **PT - 3801-997 Aveiro, Esgueira** Phone: +351 (0)234 / 303 900 Fax: +351 (0)234 / 303 910 Sirius Trading & Services srl Aleea Lacul Morii Nr. 3 ROMANIA R0-060841 Bucuresti, Sector 6 Phone: +40 (0)21 / 430 40 06 Fax: +40 (0)21 / 430 40 02 INEA RBT d.o.o. SERBIA Izletnicka 10 SER-113000 Smederevo Phone: +381 (0)26 / 615 401 Fax: +381 (0)26 / 615 401 SIMAP s.r.o. **SLOVAKIA** Jána Derku 1671 SK-911 01 Trencín Phone: +421 (0)32 743 04 72 Fax: +421 (0)32 743 75 20 PROCONT, spol. s r.o. Prešov SLOVAKIA Kúpelná 1/Å SK-080 01 Prešov Phone: +421 (0)51 7580 611 Fax: +421 (0)51 7580 650 INEA RBT d.o.o. SLOVENIA Stegne 11 **SI-1000 Ljubljana** Phone: +386 (0)1 / 513 8116 Fax: +386 (0)1 / 513 8170 Beijer Electronics AB SWEDEN Box 426 **SE-20124 Malmö** Phone: +46 (0)40 / 35 86 00 Fax: +46 (0)40 / 93 23 01 Omni Ray AG SWITZERLAND lm Schörli 5 CH-8600 Dübendorf Phone: +41 (0)44 / 802 28 80 Fax: +41 (0)44 / 802 28 28 GTS TURKEY Bayraktar Bulvari Nutuk Sok. No:5 TR-34775 Yukarı Dudullu-Ümraniye-İSTANBUL Phone: +90 (0)216 526 39 90 Fax: +90 (0)216 526 3995 CSC Automation Ltd. UKRAINE 4-B, M. Raskovoyi St. UA-02660 Kiev Phone: +380 (0)44 / 494 33 55 Fax: +380 (0)44 / 494-33-66 UKRAINF Systemgroup 2 M. Krivonosa St. UA-03680 Kiev Phone: +380 (0)44 / 490 92 29 Fax: +380 (0)44 / 248 88 68

EURASIAN REPRESENTATIVES TOO Kazpromavtomatika KAZAKHSTAN UI. Zhambyla 28 KAZ-100017 Karaganda Phone: +7 7212 / 50 10 00 Fax: +7 7212 / 50 11 50 MIDDLE EAST REPRESENTATIVES ILAN & GAVISH Ltd. ISRAEL 24 Shenkar St Kirvat Arie IL-49001 Petah-Tiqva Phone: +972 (0)3 / 922 18 24 Fax: +972 (0)3 / 924 0761 **GIRIT CELADON LTD** ISRAEL 12 H'aomanut Street **IL-42505 Netanya** Phone: +972 (0)9 / 863 39 80 Fax: +972 (0)9 / 885 24 30 **CEG INTERNATIONAL** LEBANON Cebaco Center/Block A Autostrade DORA Lebanon - Beirut Phone: +961 (0)1 / 240 430 Fax: +961 (0)1 / 240 438

AFRICAN REPRESENTATIVE SOUTH AFRICA Private Bag 2016 SOUTH AFRICA ZA-1600 Isando Phone: + 27 (0)11 / 977 0770 Fraction (1) / 977 0770 Fax: + 27 (0)11 / 977 0770 Fraction (2) / 977 0770 Fraction (2) / 977 0770

