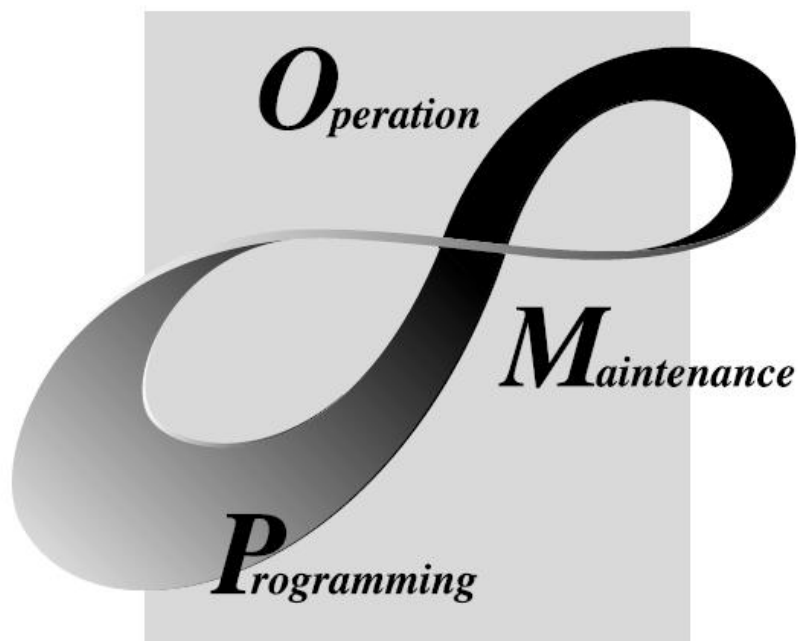


***MX OPC Configurator Version 6.05***

**Operating Manual**

**mitsubishi**



Art.no.: V0600-1LOC-E- 221608  
Version: 6.05  
March 2014

**MELSOFT**  
**Integrated FA Software**

# **MX OPC Configurator**

**(C)2014 Mitsubishi FA-EDC**

All rights reserved. No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of the publisher.

Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

Printed: March 2014

## INTRODUCTION

Thank you for choosing the Mitsubishi MELSOFT Series Integrated FA software.  
Read this manual and make sure you understand the functions and performance of MELSOFT series thoroughly in advance to ensure correct use.

## CONTENTS

<b>1 Introduction to MX OPC Server</b>	<b>9</b>
1.1 Compatibility with Former Versions.....	9
1.2 What Is OLE for Process Control.....	10
1.3 MX OPC Runtime.....	10
1.4 MX OPC Configurator.....	11
1.5 OPC client software.....	12
1.6 Installing MX OPC Server.....	13
1.6.1 System Requirements .....	13
1.6.2 Installation Procedure .....	14
1.6.3 Configuring the service .....	21
1.6.4 Rebooting the PC .....	23
1.7 MX OPC Server Quick Start.....	23
1.7.1 Starting the MX OPC Configurator .....	24
1.7.2 Setting Up the Driver .....	25
1.7.3 Creating Communication Channels .....	26
1.7.4 Configuring a New Device .....	36
1.7.5 Creating Data Tags .....	38
1.7.6 Starting the MX OPC Runtime Driver .....	41
1.7.7 Connecting to the Server Using OPC DataSpy .....	42
<b>2 Using the Configurator</b>	<b>44</b>
2.1 Starting the Configurator.....	45
2.2 Creating Configuration Databases.....	46
2.3 Toolbars .....	47
2.4 Menus.....	48
2.4.1 File Menu .....	49
2.4.1.1 MXChange database actions.....	50
2.4.1.2 MXChange project actions.....	50
2.4.1.3 iQ Works workspace actions .....	51
2.4.1.4 iQ Works project actions.....	51
2.4.2 Edit Menu .....	52
2.4.3 Rename .....	52
2.4.4 Start GX IEC developer .....	53
2.4.5 View Menu .....	54
2.4.6 Go Menu .....	55
2.4.7 Tools Menu .....	56
2.4.7.1 Options.....	56
2.4.7.1.1 General Tab.....	56
2.4.7.1.2 Advanced Tab.....	58
2.4.7.1.3 Logging Tab.....	60
2.4.7.2 Compact/Repair MS Access Database.....	62
2.4.8 Help Menu .....	63
<b>3 Configuring the Address Space</b>	<b>64</b>
3.1 Tree View.....	64

3.2 List View.....	64
3.3 Devices.....	64
3.3.1 Basic Device Properties .....	65
3.3.2 Advanced Device Properties .....	69
3.3.3 Device tuning .....	76
3.3.4 Adding a New MX Device .....	77
3.3.5 Basic Modbus Properties .....	84
3.3.6 Advanced Modbus Properties .....	86
3.3.7 Adding a New Modbus Device .....	87
3.3.8 Modbus serial port details .....	89
3.3.9 Modbus device parameters .....	90
3.3.10 Transfer setup management .....	91
3.3.10.1 Creating new transfer setups.....	92
3.3.10.2 Renaming / removing transfer setups.....	93
3.3.11 Communication Setting Wizard .....	95
3.3.12 Modem support .....	103
3.4 Data Tags.....	105
3.4.1 Basic MX Data Tag Properties .....	105
3.4.2 Addressing .....	108
3.4.3 Advanced MX Data Tag Properties .....	109
3.4.4 MX Tag Properties: Alarms .....	113
3.4.5 Multiply pages on Tag dialogs .....	114
3.4.6 Adding a New MX Data Tag .....	116
3.4.7 Basic Modbus Data Tag Properties .....	118
3.4.8 Advanced Modbus Data Tag Properties .....	119
3.4.9 Multiply pages on Modbus Tags .....	121
3.4.10 Writing to tags .....	122
3.5 Groups.....	123
3.5.1 Basic Group Properties .....	124
3.5.2 Advanced Group Properties .....	124
3.5.3 Adding a New Group .....	125
3.6 Exporting Configuration Data to a CSV File.....	127
3.7 Importing Configuration Data from a CSV File.....	128
3.8 Activating the Database.....	130

<b>4 Configuring Alarm Definitions</b>	<b>130</b>
--	------------

4.1 Limit Alarm Definitions.....	131
4.1.1 Basic Limit Alarm Properties .....	131
4.1.2 Advanced Limit Alarm Properties .....	132
4.1.3 Creating a New Limit Alarm Definition .....	133
4.2 Digital Alarm Definitions.....	134
4.2.1 Basic Digital Alarm Properties .....	135
4.2.2 Advanced Digital Alarm Properties .....	136
4.2.3 Creating a New Digital Alarm Definition .....	136

<b>5 Configuring Simulation Signals</b>	<b>138</b>
---	------------

5.1 Basic Simulation Signal Properties.....	138
5.2 Creating a New Simulation Signal.....	139

<b>6 MX OPC Server Runtime Module</b>	<b>141</b>
---------------------------------------	------------

6.1 OPC DataSpy .....	142
6.2 Activating the Database.....	145



6.3 Starting and Stopping the Driver .....	145
6.4 Statistics Mode.....	146
6.4.1 Setting the Statistics Mode Refresh Rate .....	147
6.4.2 Device Statistics .....	148
6.4.3 Device Statistics (Additional) .....	150
6.4.4 Statistical Tags in OPC Address Space .....	151
6.5 Monitor View.....	153
6.5.1 Setting the Monitor View Refresh Rate .....	154
6.5.2 Enable Monitoring .....	154
6.5.3 Disable Monitoring .....	155
6.6 Writing to Tags.....	155
6.7 Special item names.....	157
6.7.1 Accessing array items .....	157
6.7.2 Accessing bits .....	158
6.8 Dynamic Tag Support.....	159
6.8.1 Refresh Dynamic Tags .....	162
6.9 Automatic Backup.....	162
6.10 Diagnostics.....	163
6.11 Showing Register Ranges.....	164
<b>7 Configuring Conversion Definitions</b> .....	<b>165</b>
7.1 Basic Conversion Properties.....	165
7.2 Creating a New Conversion Definition.....	166
7.3 Assigning Conversion Definitions to Tags.....	168
<b>8 Configuring Poll Method Definitions</b> .....	<b>169</b>
8.1 Basic Poll Method Properties.....	170
8.2 Creating a New Poll Method Definition.....	171
8.3 Assigning Poll Method Definitions to Tags.....	172
<b>9 Using MXChange</b> .....	<b>173</b>
9.1 Connecting to MXChange.....	173
9.1.1 Connecting to a new MXChange database .....	174
9.1.2 Connecting to an existing MXChange database .....	176
9.2 Detaching from MXChange.....	179
9.3 Managing MXChange databases.....	179
9.3.1 Changing the MXChange password .....	180
9.3.2 Compacting the MXChange database .....	180
9.3.3 Save the MXChange database with a new name .....	181
9.3.4 Deleting the MXChange database .....	181
9.4 Exporting to MXChange.....	182
9.5 Importing from MXChange.....	184
9.6 Synchronising with MXChange.....	186
9.7 DUT tag support.....	188
9.7.1 Restrictions on DUT variables .....	190
9.7.2 Unallocated addresses .....	190
9.8 Back up / restore from MXChange.....	191

<b>10 Using iQ Works / MXChange2</b>	<b>193</b>
--------------------------------------	------------

10.1 Connecting to iQ Works.....	193
10.1.1 Connecting to a local workspace .....	195
10.1.2 Connecting to a remote workspace .....	198
10.2 Detaching from iQ Works.....	202
10.3 Supported labels.....	203
10.4 Exporting to iQ Works.....	203
10.5 Importing from iQ Works.....	210
10.6 Adding iQ Works labels.....	211
10.7 Restrictions on editing labels.....	216
10.8 Change notification.....	217
10.9 Synchronising with iQ Works.....	219
10.10 Unlinking devices from iQ Works.....	220
10.11 Structure support.....	221
10.12 Switching between iQ Works and MXChange.....	223
10.13 Deleting unused projects.....	223
10.14 Using MXCSV files.....	224

<b>11 Transfer Setup</b>	<b>225</b>
--------------------------	------------

11.1 Configuring Transfer Setups.....	225
11.2 Transfer Setup Wizard.....	230
11.2.1 PC Universal Serial Bus (USB) Interface .....	233
11.2.2 PC Universal Serial Bus (USB) Interface via GOT .....	233
11.2.3 PC RS232 Serial Interface .....	234
11.2.4 PC RS232 Serial Interface via GOT .....	235
11.2.5 PC Ethernet Board .....	236
11.2.6 PC Ethernet Board via GOT .....	248
11.2.7 PC Modem .....	248
11.2.7.1 Modem Line Settings.....	249
11.2.8 PC CC-Link IE Control Board .....	258
11.2.9 PC MELSECNET/H Board .....	261
11.2.10 PC MELSECNET/10 Board .....	264
11.2.11 PC CC-Link IE Field Board .....	267
11.2.12 PC CC-Link Board .....	269
11.2.13 PC Q Series Bus .....	271
11.2.14 PC GX Simulator .....	273
11.2.15 PC GX Simulator2 .....	274
11.2.16 PC MT Simulator2 .....	276
11.2.17 PC CPU Board .....	278
11.2.18 PLC CPU Interface .....	278
11.2.19 PLC C24 Interface .....	285
11.2.20 PLC FX Extended Port .....	296
11.2.21 PLC Modem .....	302
11.2.22 PLC Ethernet Interface .....	307
11.2.23 PLC G4 Module .....	321
11.2.24 PLC CC-Link IE Field Ethernet Adapter .....	327
11.2.25 Network .....	338
11.2.26 Other Station .....	341
11.2.27 GOT Side Settings .....	346
11.2.28 Enter Comment .....	350

11.3 Special Setups.....	351
11.3.1 Communication with Multi-CPU Systems .....	351
11.3.2 Restrictions with Q00, Q00J and Q01 CPUs .....	353
11.3.3 Redundant CPUs .....	354
11.3.4 Connection Types for Q24DHCCPU-V C-Controllers .....	361

<b>12 Appendix: Using OPC from Visual Basic</b>	<b>361</b>
---	------------

12.1 Prerequisites.....	361
12.2 Terminology.....	362
12.3 Adding the OPC Libraries to Your Project.....	363
12.4 OPC constants.....	364
12.5 Listing installed OPC servers.....	365
12.6 Connecting to an OPC server.....	365
12.7 Browsing the OPC server items.....	366
12.8 Reading data values from the server.....	369
12.9 Writing data values to the server.....	371
12.10 Disconnecting from an OPC server.....	372
12.11 References.....	372

<b>13 Appendix: Understanding ACT error codes</b>	<b>373</b>
---	------------

13.1 HRESULT error codes.....	374
13.2 ACT error codes.....	375
13.3 CPU, module and network board errors.....	419

<b>14 Appendix: DCOM Configuration</b>	<b>420</b>
--	------------

14.1 Introduction.....	420
14.1.1 Security considerations .....	420
14.1.2 Compatibility .....	420
14.2 Before you begin.....	421
14.2.1 User accounts .....	422
14.3 'Express' DCOM configuration on Windows XP Service Pack 2.....	424
14.4 Configuring DCOM on Windows Vista or later.....	431
14.4.1 Configuring the Windows firewall .....	432
14.4.2 Configuring DCOM security .....	438
14.4.3 Configuring services to use a specific user account .....	452
14.5 Configuring DCOM on Windows XP service pack 2.....	454
14.5.1 Configuring the Windows firewall .....	455
14.5.2 Configuring DCOM security .....	458
14.5.3 Configuring services to use a specific user account .....	471

<b>15 Appendix: Using XML-DA</b>	<b>473</b>
----------------------------------	------------

15.1 Security considerations.....	474
15.2 Before you begin.....	476
15.3 Configuring XML-DA.....	477
15.3.1 Turning off user account control .....	477
15.3.2 Installing Internet Information Server (IIS) .....	479
15.3.2.1 Installing Internet Information Server on Windows XP .....	480
15.3.2.2 Installing Internet Information Server (Vista or Windows 7).....	481

15.3.2.3 Installing Internet Information Server (Windows Server 2008).....	482
15.3.3 Installing Microsoft .NET framework v1.1 .....	488
15.3.3.1 Adding ASP.NET support to a new installation of Internet Information Server.....	489
15.3.4 Updating system components .....	490
15.3.5 Installing the XML-DA wrapper .....	490
15.3.6 Configuring the XML-DA wrapper .....	493
15.3.7 Configuring Internet Information Server (IIS) .....	494
15.3.7.1 Configuring Internet Information Server (IIS) on Windows XP .....	495
15.3.7.2 Configuring Internet Information Server (IIS) on later operating systems.....	499
15.3.8 COM configuration .....	505
15.3.8.1 COM configuration on Windows Vista or later.....	505
15.3.8.2 COM configuration on Windows XP service pack 2.....	515
15.3.9 Configuring the Windows firewall .....	523
15.3.9.1 Configuring the Windows firewall (Windows XP).....	524
15.3.9.2 Configuring the Windows firewall (Windows Vista or later).....	526
15.4 Testing XML-DA.....	528
15.5 Programming with XML-DA.....	531
15.5.1 Using XML-DA from JavaScript .....	531
15.5.1.1 Using the 'XMLHttpRequest' object to call web services.....	532
15.5.1.2 Updating parts of the web page dynamically.....	534
15.5.1.3 Sample application structure.....	535
15.5.2 Using XML-DA from Visual Studio .....	537
15.5.2.1 Adding a web service reference to a Visual Studio project.....	537
15.5.2.2 C# sample application.....	538
15.5.2.2.1 Shared code and definitions .....	539
15.5.2.2.2 Initialisation.....	540
15.5.2.2.3 GetStatus() method.....	540
15.5.2.2.4 Browse() method.....	541
15.5.2.2.5 Read() method.....	543
15.5.2.2.6 Write() method.....	545
15.5.2.2.7 Reading data using subscriptions.....	547
15.6 Troubleshooting.....	551
15.6.1 Problems and solutions .....	551
15.6.2 Using event viewer for troubleshooting .....	553

# 1 Introduction to MX OPC Server

The MX OPC Server 6.05 is a Mitsubishi I/O driver OPC Data Access (DA) and Alarm/Events (AE) server that provides the interface and communications protocol between a wide range of Mitsubishi hardware and your process control software. Mitsubishi drivers incorporate the following attributes to provide flexibility and ease-of-use:

OLE for Process Control (OPC) compliance.

The MX OPC Server consists of the following components:

- MX OPC Configurator
- MX Runtime

The MX OPC Server complies with version 3.00 of the OLE for Process Control (OPC) Data Access standard, version 1.01 of the OPC XML Data Access standard, and version 1.01 of the OPC Alarm and Events standard. Any OPC client application can access process hardware data through the I/O Server.

## 1.1 Compatibility with Former Versions

The MX OPC Server 6.05 is fully backward compatible with previous versions (4.20, etc.). The configuration databases from all previous versions can be used. The MX OPC Server 6.05 Configurator provides an automated conversion from the older configuration database into the current structure. All the database content will be preserved and converted into the new structure if needed.

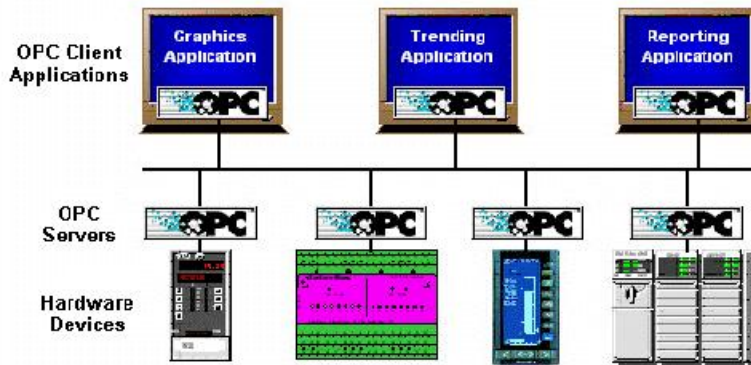
The MX OPC Server 6.05 configuration database cannot be used in the previous versions.

The MX OPC Server does not support importing of CSV configurations generated by a different version. Although this process is partially successful in some versions, the 6.05 version will not import a CSV file created before version 6.05 due to major changes in the database structure.

To preserve the configuration, the user can import the CSV file to a MDB database using the original configurator, upgrade the MDB database using the MX OPC Server 6.05 configurator and then export to CSV.

## 1.2 What Is OLE for Process Control

OLE™ for Process Control (OPC) is a standards-based approach for connecting data sources (e.g., PLCs, controllers, I/O devices, databases, etc.) with HMI client applications (graphics, trending, alarming, etc.). It enhances the interface between client and server applications by providing a universally supported and well-documented mechanism to communicate data from a data source to any client application. Included are not only a detailed guide on how to pass the data, but also specific information on other attributes to supplement those data, such as range information, data type, quality flags, and date and time information. The figure below shows the OPC Architecture, which was introduced by the OPC Foundation. By following the OPC Architecture, a device needs only one standard driver, which is an OPC-compliant server. All OPC-compliant client applications can then be connected to that device, either locally or over a network. Furthermore, connections can be made to more than one OPC server at the same time.



**OPC-Based Client-Server Architecture**

Any OPC client application can connect to any OPC server. In other words, OPC offers true Plug-and-Play capability in the fields of HMI and industrial automation. OPC server types include OPC Data Access (DA), OPC Alarm and Events (AE), and OPC Historical Data Access (HDA).

## 1.3 MX OPC Runtime

The MX OPC Runtime maintains the driver's communication channels, devices, and data tags, performs all required functions for communicating with the process hardware, and exposes the methods and properties to other applications.

In addition to performance improvements, the MX OPC Runtime provides the following functionality:

- Supports OLE for Process Control (OPC)
- Provides local configuration and control
- Supports telephone modem communication
- Provides phasing
- Enables or disables Individual devices and tags
- Provides QuickFail Logic
- Provides latched data
- Provides a time/date stamp for data and alarms
- Supports individual bit reads
- Provides advanced diagnostics.
- MX Component.

## 1.4 MX OPC Configurator

The I/O Driver Configurator is a client application to the I/O Server with a graphical user interface. The MX OPC Configurator accesses the I/O Server and lets you view and modify properties of communication channels, devices and data tags.

The MX OPC Configurator provides the following:

- The server connection - only local I/O Server.
- The tree control for an overall view of your system configuration.
- A Statistics View for displaying the statistics of your I/O driver while it is running. Statistics are provided for devices.
- A Configuration View for displaying and modifying device and tag properties.
- A Monitor View for displaying real-time data tag values during runtime mode.
- An interface to the MXComponent Communication Setting Wizard for importing or creating new configurations.
- Support for telephone modem communication.
- Templates for configuring default alarm settings.
- Simulation for tags and alarms using a built-in function pattern.

## 1.5 OPC client software

The OPC server has been tested with a range of client software. The table below shows which data types are supported by each one.

Data type	RSview	LabView 8	iFIX 3.5	InTouch v9	Genesis32 v8
BOOL	✓	✓	✓	✓	✓
WORD	Note 2	✓	Note 2	✓	✓
DWORD	Note 3	✓	Note 1	Note 1	✓
UINT	Note 2	✓	Note 2	✓	✓
UDINT	Note 3	✓	Note 3	Note 1	✓
INT	✓	✓	✓	✓	✓
DINT	✓	✓	Note 4	✓	✓
REAL	✓	✓	✓	✓	✓
STRING	✓	✓	✓	✓	✓
Bit notation .	✓	✓	✓	✓	✓
Array notation []	✓	✓	✓	✓	✓
Arrays	Note 5	✓	Note 5	Note 5	✓
n-based arrays	Note 5	Note 5	Note 5	Note 5	Note 6

### Notes

1. Supports this type only for values in signed DINT range 0..2147483647
2. Supports this type only for values in signed INT range 0..32767
3. Does not support unsigned 32-bit values
4. Does not read the correct value in all cases
5. This type of array variable is not fully supported - use of array notation [] is recommended instead.
6. When writing n-based arrays from OPC data spy, additional elements must be supplied for elements from 0 to the start of the array. For example, to write two values to an array of size 2 with lower bound 3, use '0,0,0,1,2'.

In the event of problems getting client software to read values from the OPC server, try the following:

- Select the native type of the OPC tag instead of using the default, e.g. set the requested data type to 'boolean' for BOOL tags, etc.
- Ensure that any scaling in the client package is set up correctly for the possible values of the OPC tag.
- Use array element notation (e.g. 'Item[0]') to read array elements, instead of trying to read the whole array tag.



## 1.6 Installing MX OPC Server

This section describes the steps for installing the MX OPC Server. Be sure to close any other applications before installing MX OPC Server.

There are two separate parts to the software:

- The MX OPC runtime module, which reads data from the device
- The MX OPC configurator, which defines the data to be read

The MX OPC Server runtime module can optionally be installed as a service. Services run automatically in the background whenever the computer is started; they do not rely on a user logging in to the computer to start the program manually.

### 1.6.1 System Requirements

To use MX OPC Server 6.05, your computer must meet the following minimum system requirements:

- 233 MHz processor
- 512 MB RAM
- 100 MB disk space
- 8x speed CD-ROM drive
- VGA video card (256 or more colors for best results)
- One of the following operating systems:
  - Windows® 8 (32-bit or 64-bit)
  - Windows® 7 (32-bit or 64-bit)
  - Windows® Server 2008 (R2)
  - Windows® XP with Service Pack 3
  - Windows® 2003 Server

Performance will improve with higher specification hardware (faster data updates).

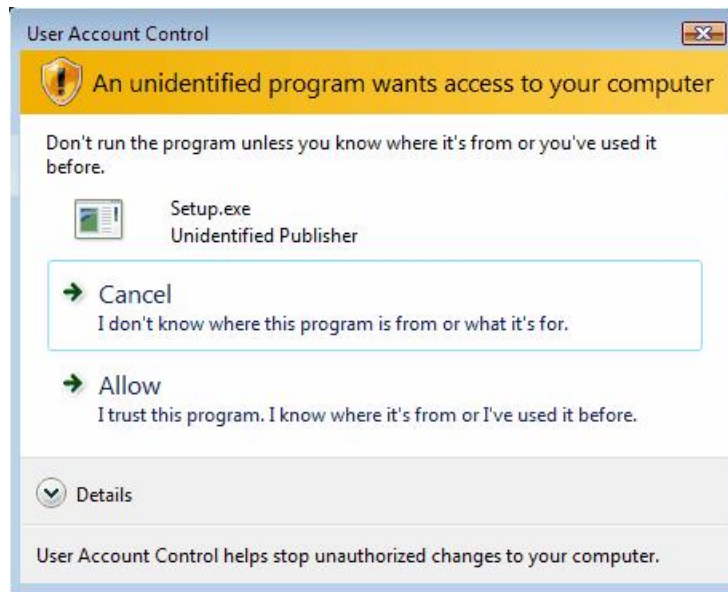
**Note:** Only versions of Microsoft Windows that accommodate Unicode character sets are supported.

## 1.6.2 Installation Procedure

The following steps detail the MX OPC Server installation:

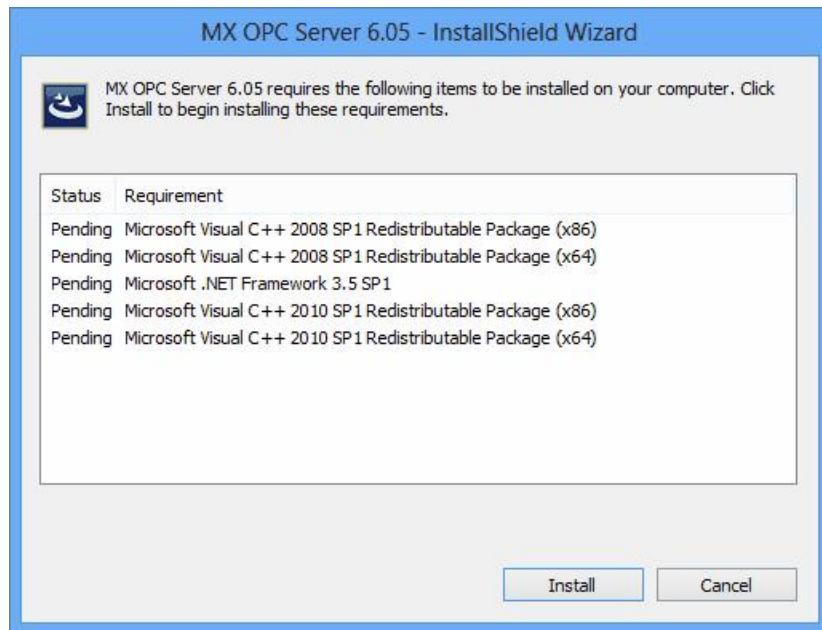
**Note:** If your operating system (e.g., Windows NT) requires a login name, you must log in with administrator capability before installing MX OPC Server.

1. Before installing the software, be sure that all other applications are closed and/or disabled.
2. Insert the product CD into your CD-ROM drive. If the option to start CDs automatically is enabled in your operating system, the CD introduction starts automatically. Otherwise, browse to your CD-ROM drive and run the program "Setup.exe".
3. If you are installing on Windows Vista or later operating systems, you may see a warning prompt similar to the one below. Click **Allow** or **Yes** to continue the installation.



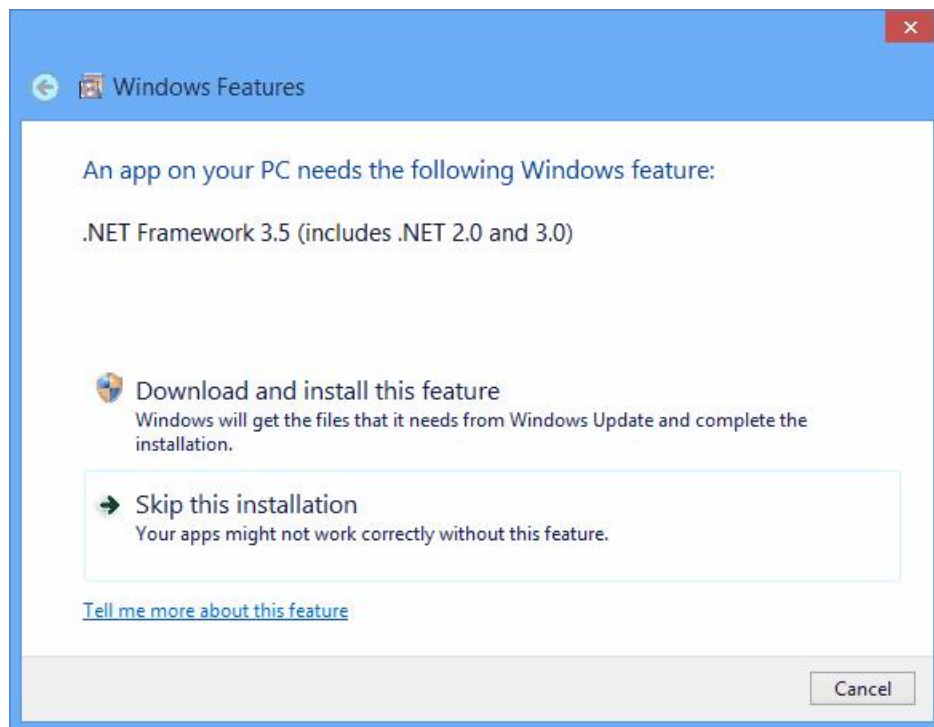
**Account control prompt**

4. You will be prompted to install some additional required software if it is not already installed on your computer. Click 'Install' to continue.



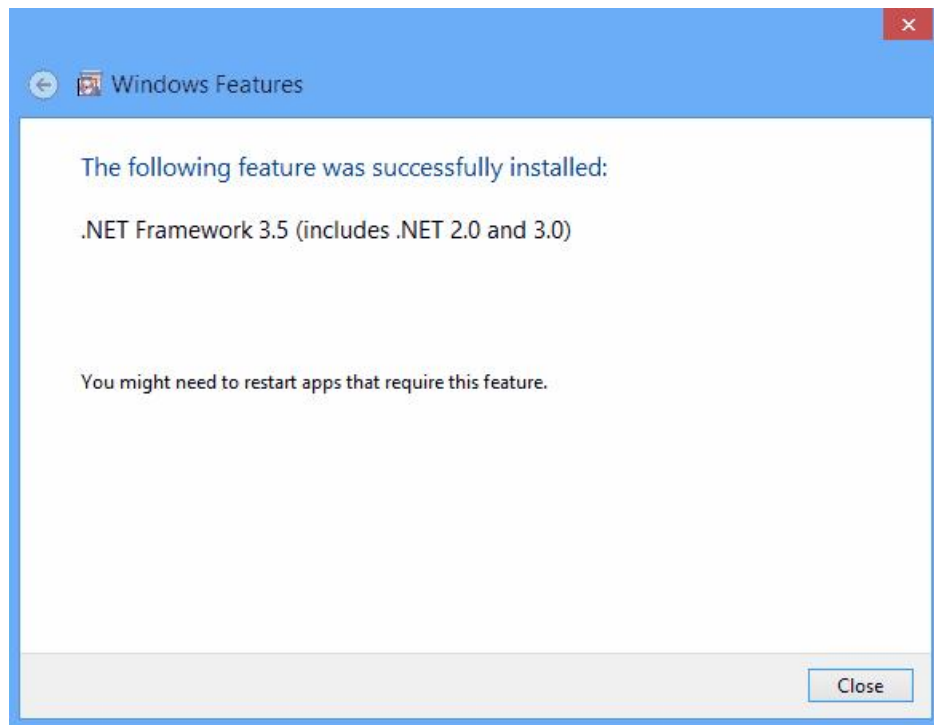
**Install additional required software**

On later operating systems (such as Windows 8) you may be prompted to install additional Windows features. If so, select 'Download and install this feature'.



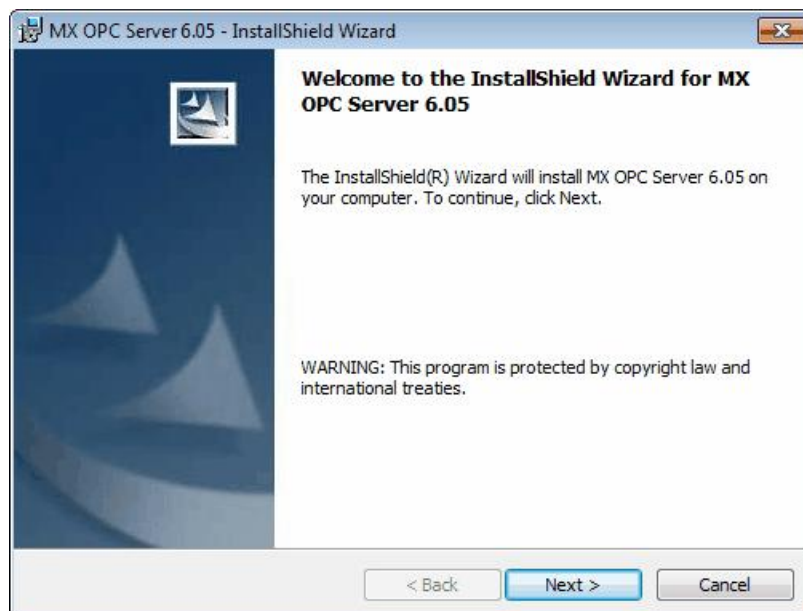
**Install Windows features**

When the features have finished installing, you will see the screen below. Click 'Close' to return to the installer.



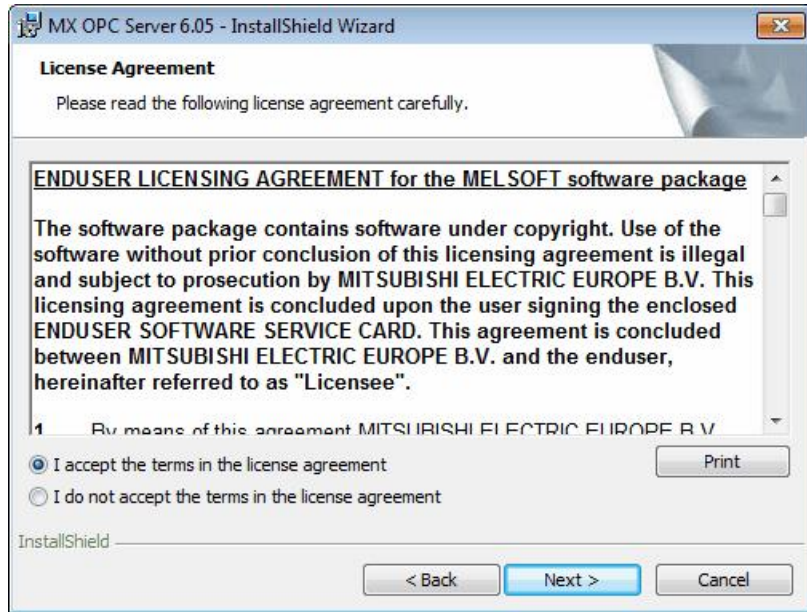
**Windows features installed**

- 5. At the Welcome screen (shown in the figure below), click the Next button to continue.



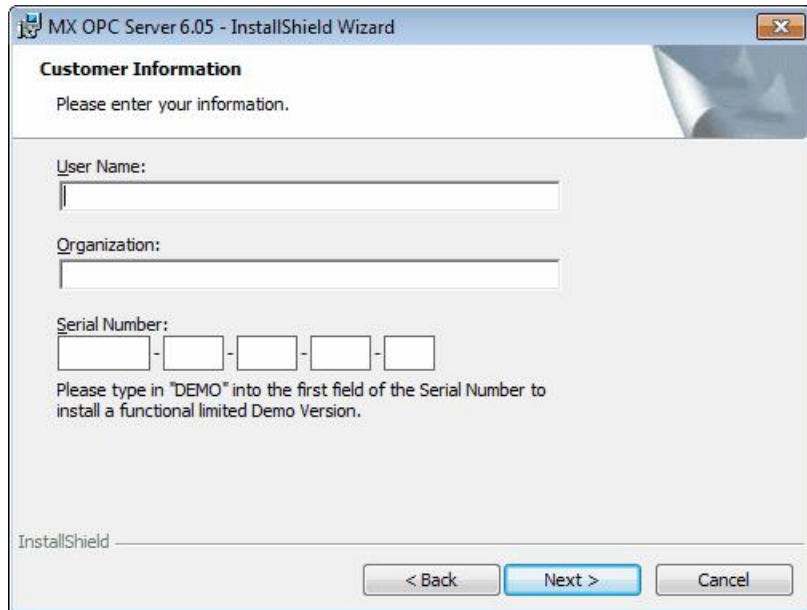
**Welcome Screen**

- 6. The **Software License Agreement** dialog box appears, as shown in the figure below. Read the License Agreement. Click **Next** if you accept the terms of the agreement.



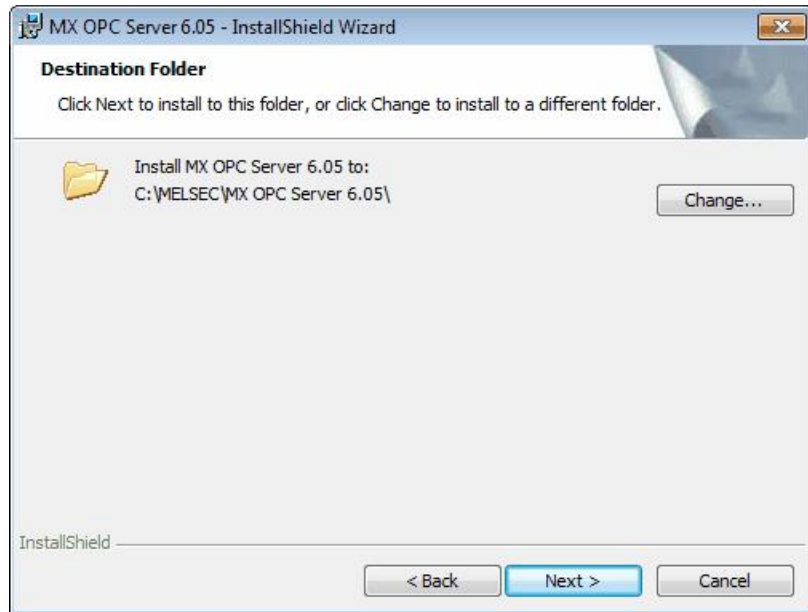
**Software License Agreement**

- 7. The **Customer Information** dialog box appears, as shown in the figure below. Enter your name, your company's name, and your product serial number. Click **Next** to continue, and confirm your registration information.



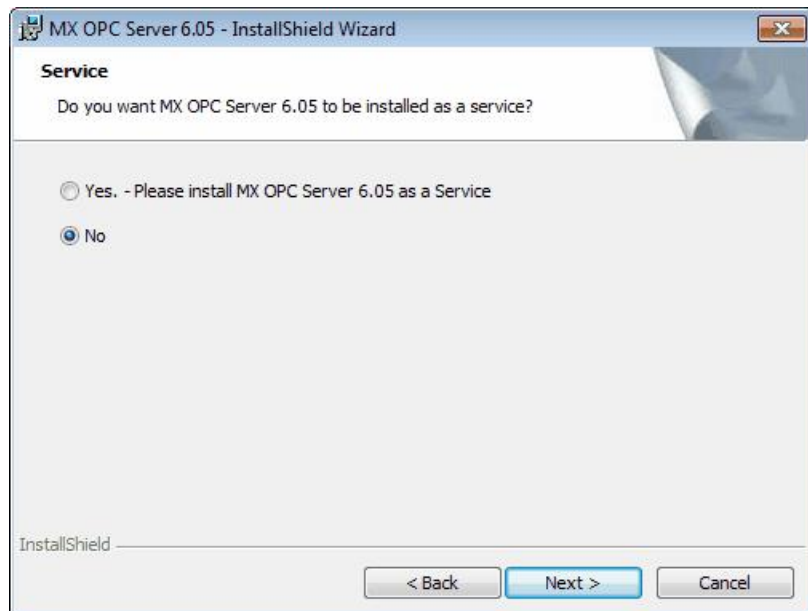
**Entering Customer Information**

- 8. The **Choose Destination Location** dialog box appears, as shown in the figure below. Choose the destination location for the software installation. If you wish to keep the default location, click the **Next** button to continue. If you prefer to select a different directory location for the installation, click the **Change** button to select a different drive or directory. Click **Next** to continue.



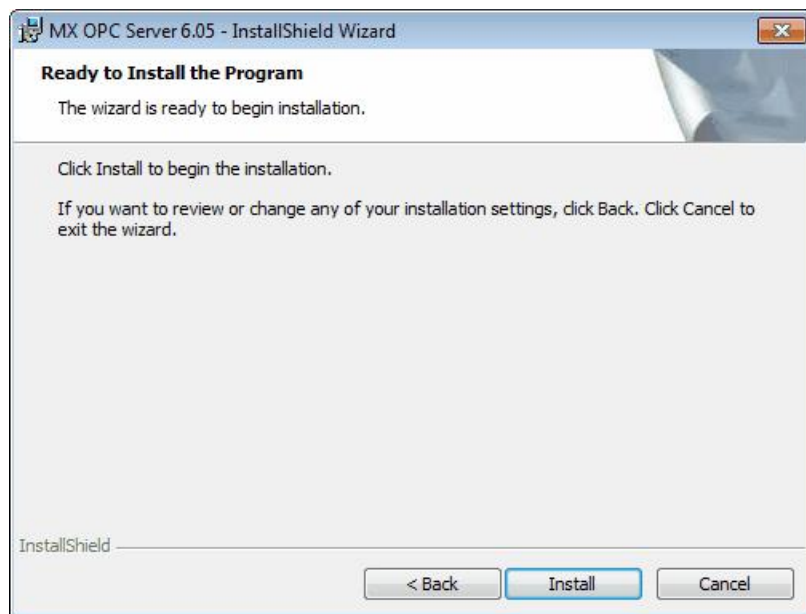
**Choosing the Destination Location**

- 9. The **Setup Type** dialog box appears, as shown in the figure below, giving you the option to install MX OPC Server as a Windows service. Select **Yes** or **No**, and then click the **Next** button to continue.



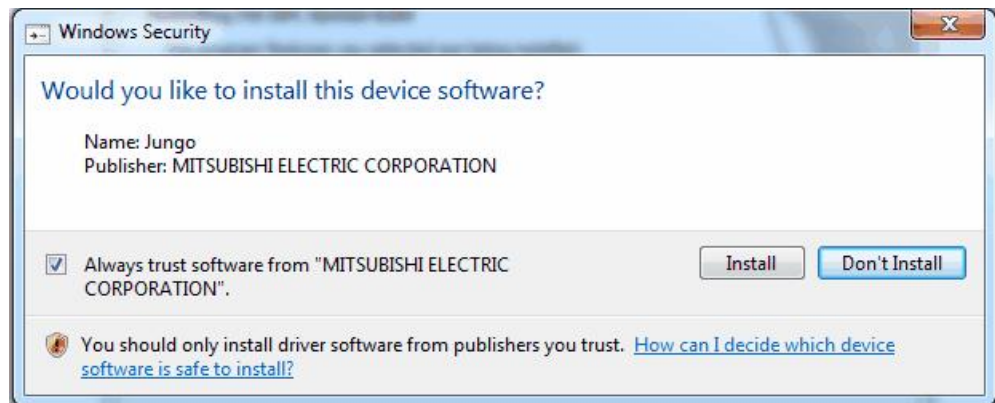
**Option to Run MX OPC Server as a Service**

10. Click **Install**, as shown in the figure below, to begin installing the files. MX Component, EasySocket and MXChange server will also be installed.  
**Note:** You can click **Cancel** to stop the installation procedure at any time.



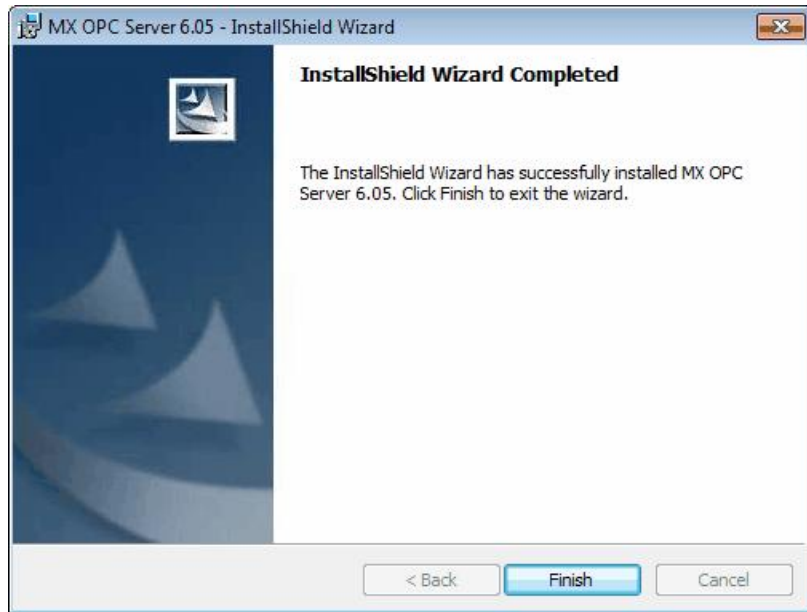
**Ready to install**

11. During the installation, depending on your operating system you may see a warning similar to the one below prompting you to install the necessary drivers. Select 'Install' or 'Yes' when prompted.



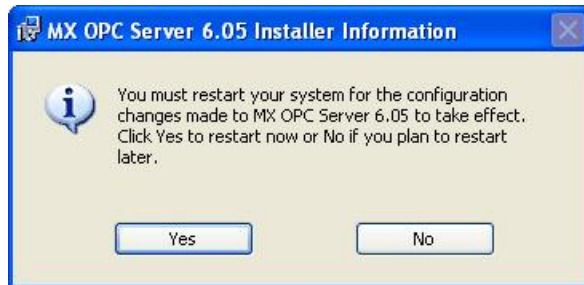
**Driver installation prompt**

- 12. After all files and components have been installed, click the **Finish** button to complete the installation, as shown in the figure below. If you installed MX OPC server as a service, you will also be given the option to configure the service immediately. After choosing whether to configure the service now or later, click the **Finish** button.



Installation Complete

- 13. You may be prompted to restart your computer before using the product. If a prompt similar to the one below is shown, either select **Yes** to restart immediately (and force all other running applications to close) or **No** to close other applications and restart manually.



Restart prompt

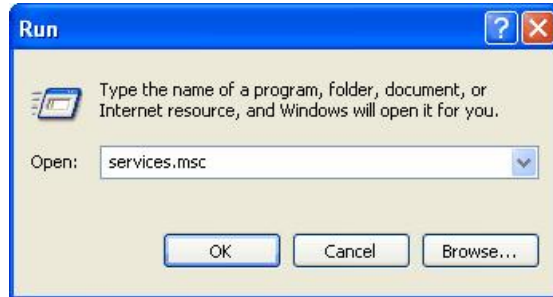


### 1.6.3 Configuring the service

When the OPC server is installed as a service, it can be useful to configure it to run using a specific user account. The 'OPC enumeration' EXE is also installed as a service. In this section, the 'OPC enumeration' EXE will be used as an example, but the same method applies when the OPC server settings are changed.

First, open the 'services' page:

**Windows XP, Windows 2000 or Windows Server 2008 R2:** select 'Start' -> 'Run', enter 'Services.msc' in the dialog that appears and click 'OK'.



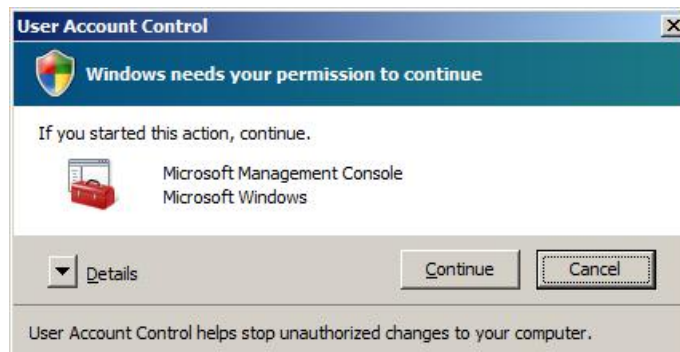
**Windows 7 or Windows Server 2008 R2:** click the Windows icon, then enter 'Services.msc' in the search box and press enter.



**Windows Vista:** click the start button, then enter 'Services.msc' in the search box and press enter.

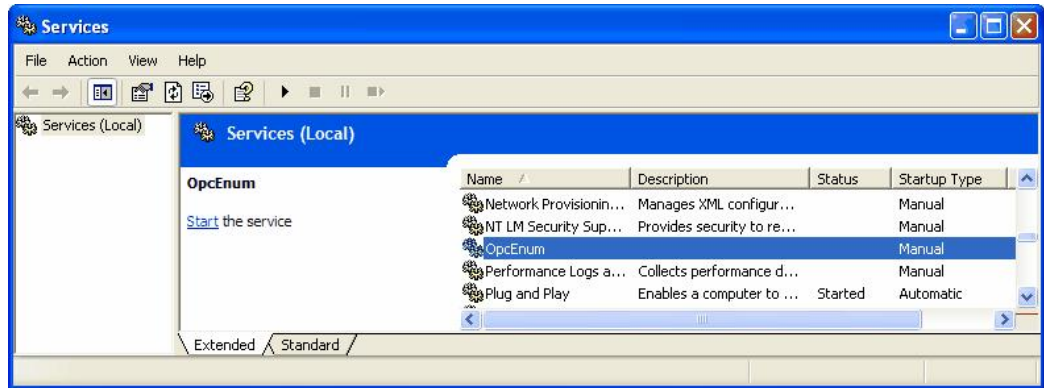


If a 'User account control' prompt like the one below is shown, click 'Continue'.

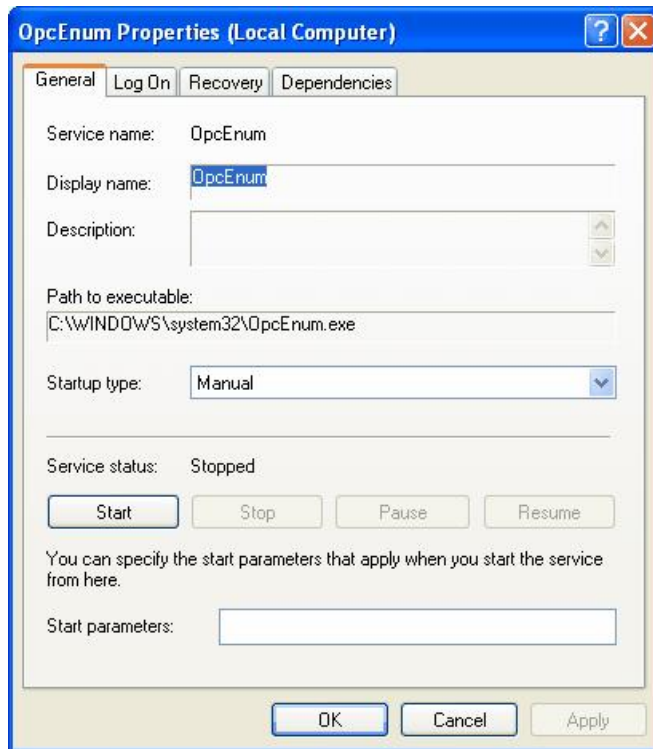


The remaining screens are similar for all operating systems.

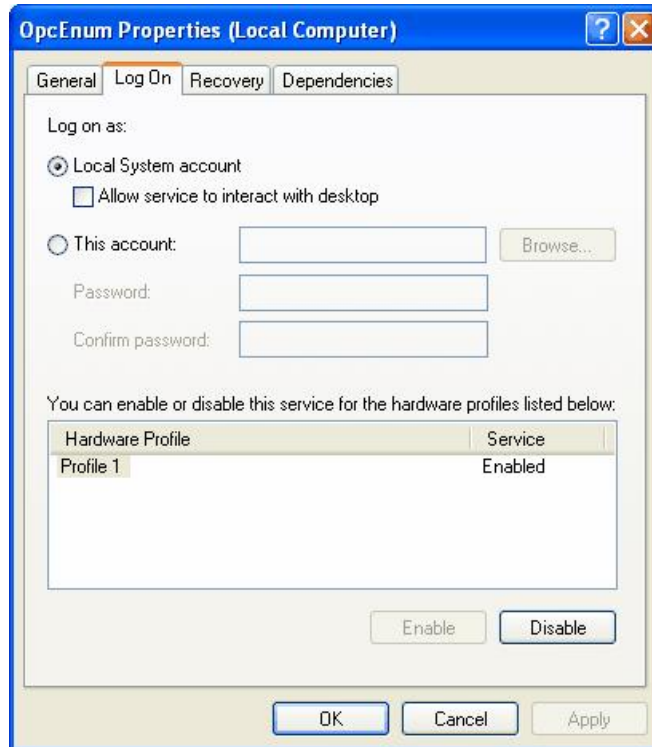
Find the service to be changed in the list. The sample screen below shows the OPC enumeration service.



Right click on the service name and select 'Properties'. The service configuration screen (below) will be shown.



Switch to the 'Log On' tab, select 'This account' and enter the username and password of the user to run as.



Now select 'OK' and close the services window.

#### 1.6.4 Rebooting the PC

In the situation where MX OPC Server is running on a PC, which needs to be rebooted or switched off, we suggest following the instructions below:

1. Start the MXConfigurator application and stop the MX OPC server
2. Close the MXConfigurator application and disconnect all clients from the MX OPC Server runtime application

This procedure needs to be followed due to the fact that the MX OPC Server runtime accesses the MSACCESS runtime database functions. In the situation where the PC is switched off while the MX OPC server is simultaneously accessing the database, there can be damage or data may not be accessible.

### 1.7 MX OPC Server Quick Start

The objectives of this Quick Start section are to:

1. Start the MX OPC Configurator.
2. Set up the driver.
3. Set up a communication channel.
4. Create and configure a new device.
5. Create and configure a new data tag.
6. Start the I/O Server driver.
7. Connect to the server using OPC DataSpy.

### 1.7.1 Starting the MX OPC Configurator

Once you have installed MX OPC Server, start the MX OPC Configurator:

1. From the Windows Start menu, select **[All] Programs > MELSOFT Application > MX OPC Server 6.05 > MX OPC Configurator**.  
**Note:** The name of the program folder may vary depending on your local settings (e.g. language settings).
2. This opens the Configurator for the MX OPC Server, as shown in the figure below. The screen consists of a split window with a tree control view in the left-hand pane and a configuration view in the right-hand pane. The Configurator provides a default standard configuration database called MXConfigurator.mdb, which contains a sample configuration project.



**Configurator Screen**

The MX OPC Configurator is a client of the MX OPC Runtime that accesses the I/O Server and lets you view and modify properties for devices and data tags. The Address Space tree control in the Configurator sets the properties and connection parameters for the following hierarchy of server objects:

- **Devices:** A device is a hardware device or station that uses the I/O driver to communicate with a client PC. The device item contains the properties and methods that govern the behavior of a device. A device is visible to the OPC client.
- **Data tags:** A data tag is an object that makes device data accessible to OPC clients. Data tags can be logically organized into groups (folders).

## 1.7.2 Setting Up the Driver

The table below provides a general overview of the steps necessary for setting up your I/O driver.

### 1 Know your process hardware:

- What device does the driver communicate with?
- What type of cable are you using?
- What is the station number for each device?
- What addresses do you want to access and what data do you want to retrieve?

### 2 Know the hardware parameters:

Example for serial communication

- COM Port
- Baud Rate
- Stop Bits
- Data Bits
- Parity

### 3 Choose the method of configuration:

Use the MX OPC Configurator

### 4 Configure the I/O Driver:

Configuring Communication Channels

Configuring devices

Setting default values for driver, channels, devices and tags

Configuring tags

### 5 Check the I/O Driver Status:

Using monitoring mode

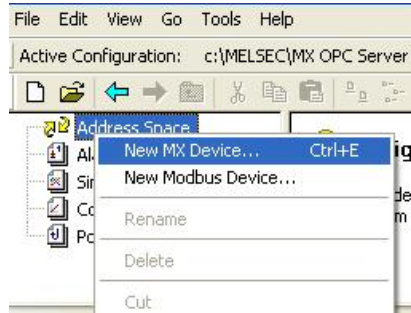
### 1.7.3 Creating Communication Channels

In the following examples, we will create and configure new channels for both the Serial and Ethernet communication paths.

#### Serial Communication Channel

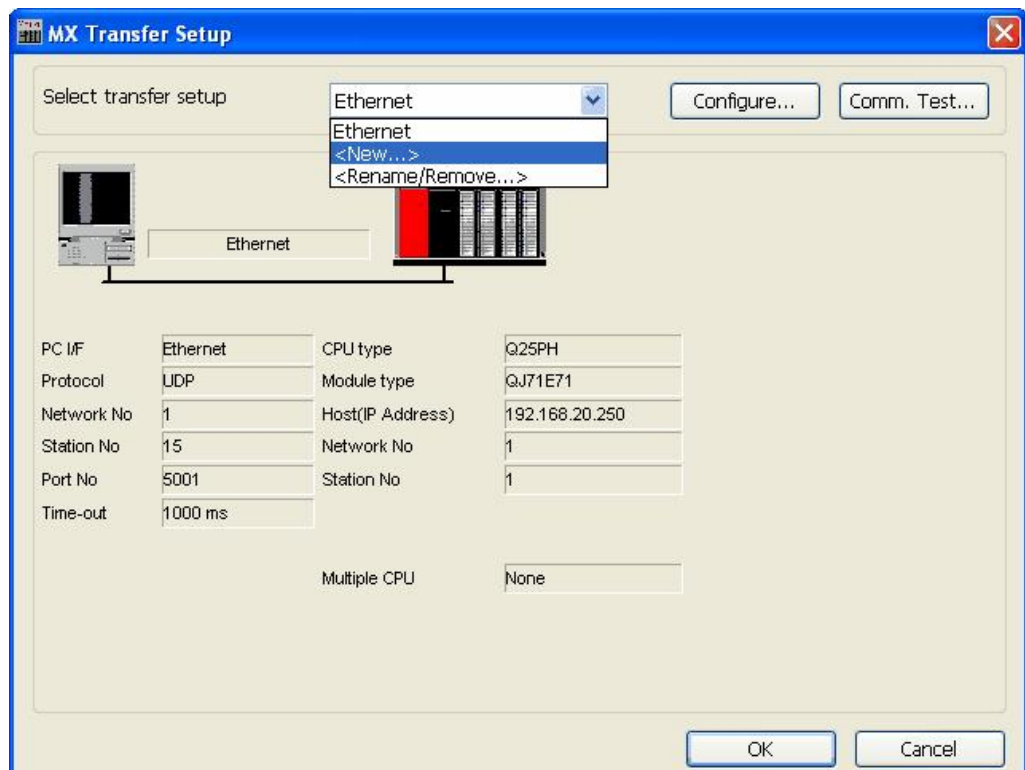
To create a new Serial communication path:

1. Right-click the Address Space tree control of the Configurator screen and select **New MX Device** from the pop-up menu, as shown in the figure below.



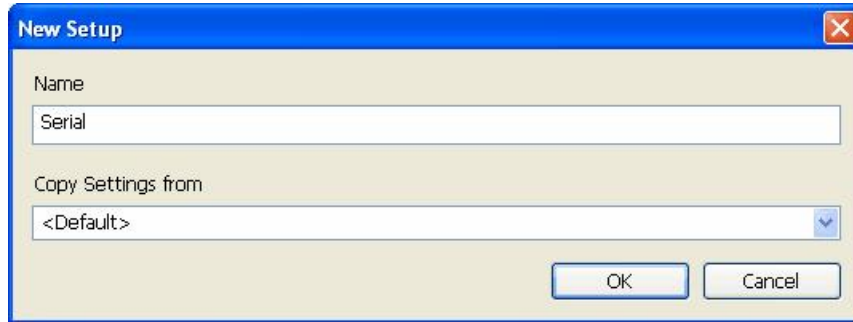
Adding a New MX Device

2. The **MX Transfer Setup dialog** will be shown. You can either select an existing transfer setup, or (as in the example below) select <New...> to create a new transfer setup.



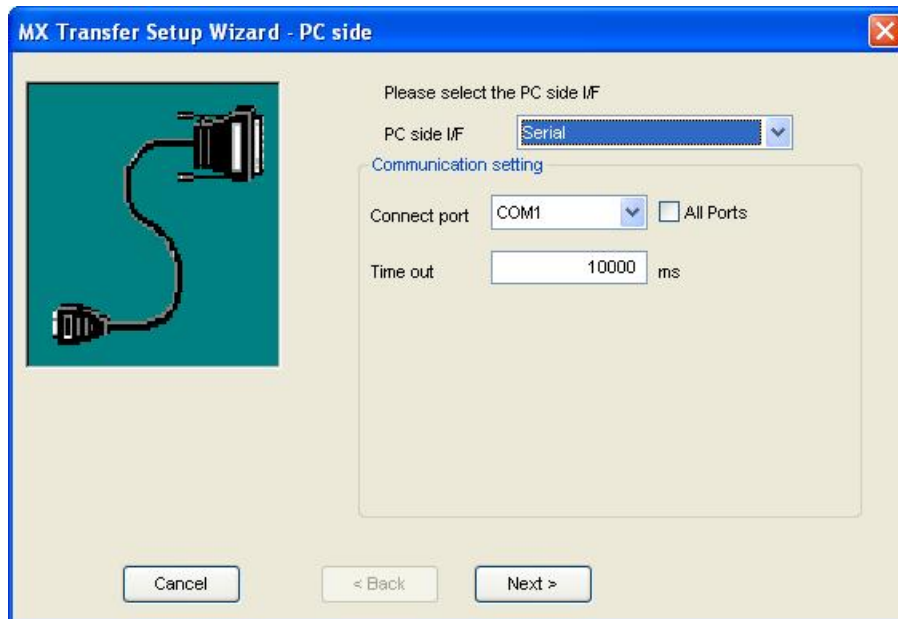
MX Transfer Setup dialog

- 3. The **New Transfer Setup dialog**<sup>[93]</sup> will appear. Enter a name for the new transfer setup (*Serial* in the picture below) and select *<Default...>* as the template, then click **OK**.



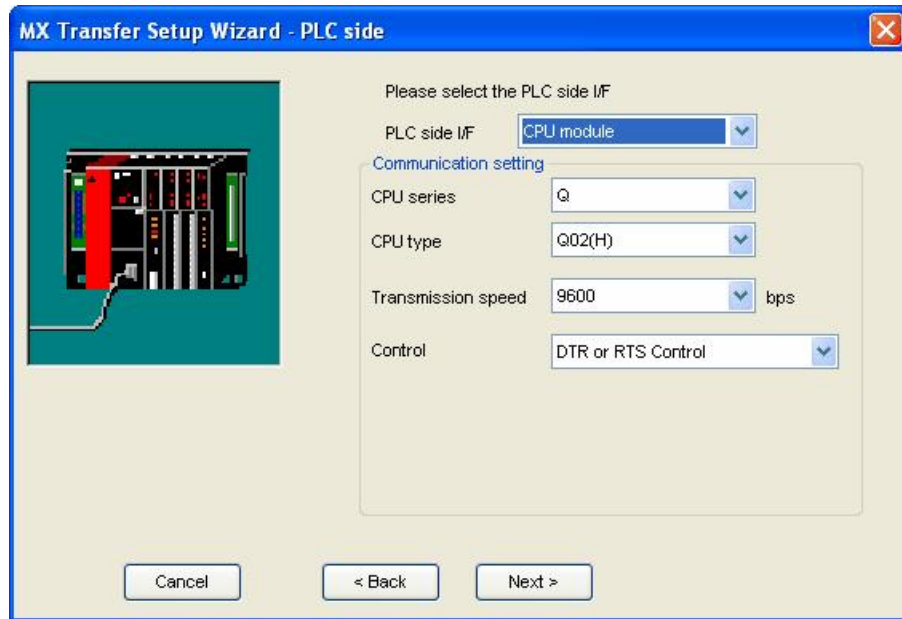
Entering a transfer setup name

- 4. The new transfer setup can now be edited using the **Communication Setting Wizard**, as shown in the figure below. Specify the communication type to use on the PC side, as shown in the figure below. In the **PC Side I/F** field, select the **Serial** connection type from the drop-down list. By default, you will only see the physical COM ports that are actually available on your computer. If you are preparing a configuration to be used on another computer which has different COM ports, check the **All ports** box to show a full list of possible serial ports. After entering the details, click **Next** to continue.



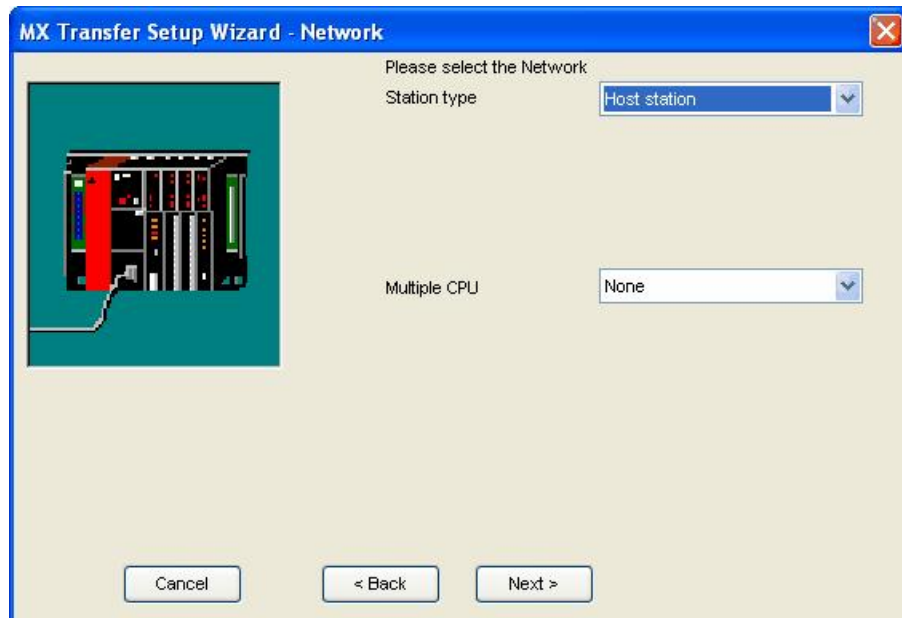
Setting up a Connection for the PC Side

- Now you must specify the communication type to use on the PLC (device) side, as shown in the figure below. In the **PLC Side I/F** field, select a communication type from the drop-down list as the connection type to use for the selected channel. Click **Next** to continue.



**Setting up the Connection for the PLC Side**

- Select the network **Station Type** and **CPU Type** (if applicable) from the respective drop-down lists, as shown in the figure below. Click **Next** to continue.



**Configuring Network Settings**

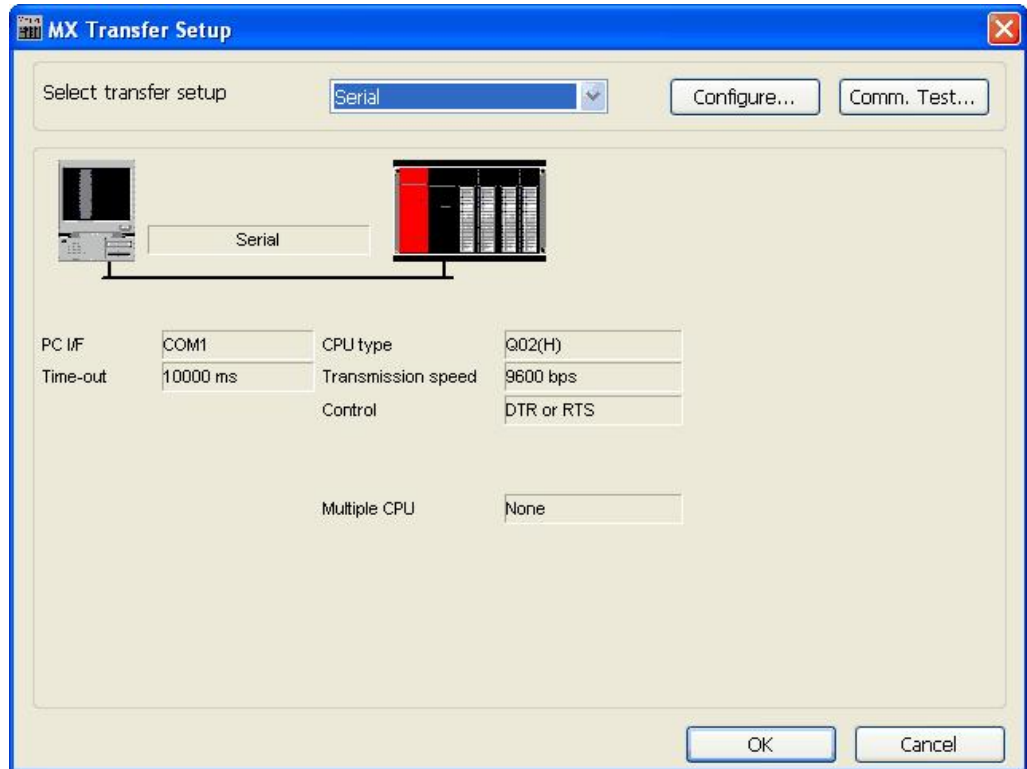


7. Click the **Finish** button to complete the communication channel setup, as shown in the figure below.



**Completing Channel Setup**

8. The **MX Transfer Setup** dialog box appears with the new transfer setup selected, as shown in the figure below. Each network node is shown graphically, with the node properties shown under the graphic. The left-hand side of the dialog lists the configuration properties for the PC side. You can double-click the image of a node (or double-click the properties list) to edit the configuration properties. The right-hand side of the dialog lists the configuration properties for the PLC device side. You can double-click the image of the PLC device (or double-click the configuration properties list) to edit the PLC device side configuration properties. You can also click the **Configure** button to return to the **Communication Setting Wizard**. Click the **OK** button to save the changes.



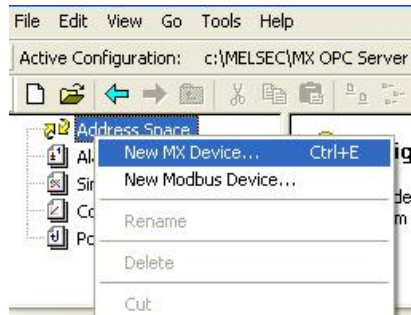
**MX Transfer Setup dialog (configured)**

**Note:** For complete information about channel configuration properties and the **Communication Setting Wizard**, see "[Configuring the address space](#)"<sup>[64]</sup>.

### Ethernet Communication Channel

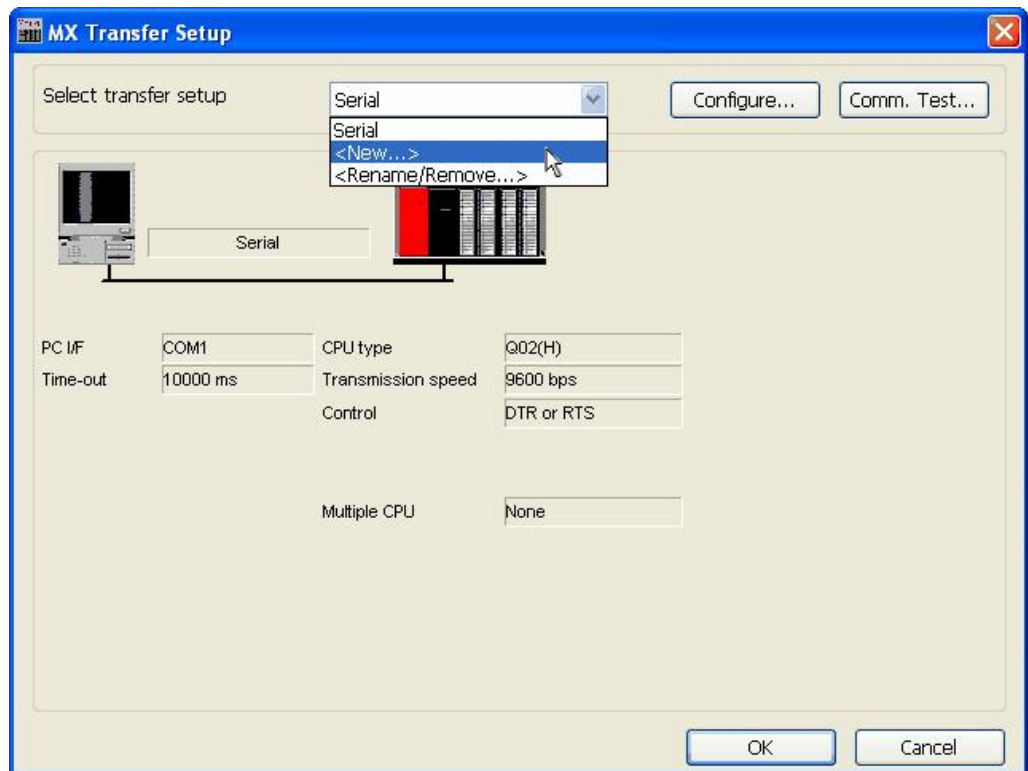
To create a new Ethernet communication path:

1. Right-click the **Address Space** tree control of the Configurator screen and select **New MX Device** from the pop-up menu, as shown in the figure below.



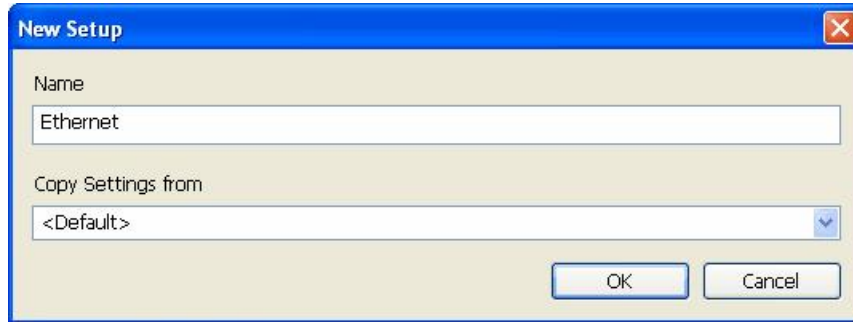
**Adding a New MX Device**

2. The **MX Transfer Setup dialog** will be shown. You can either select an existing transfer setup, or (as in the example below) select **<New...>** to create a new transfer setup.



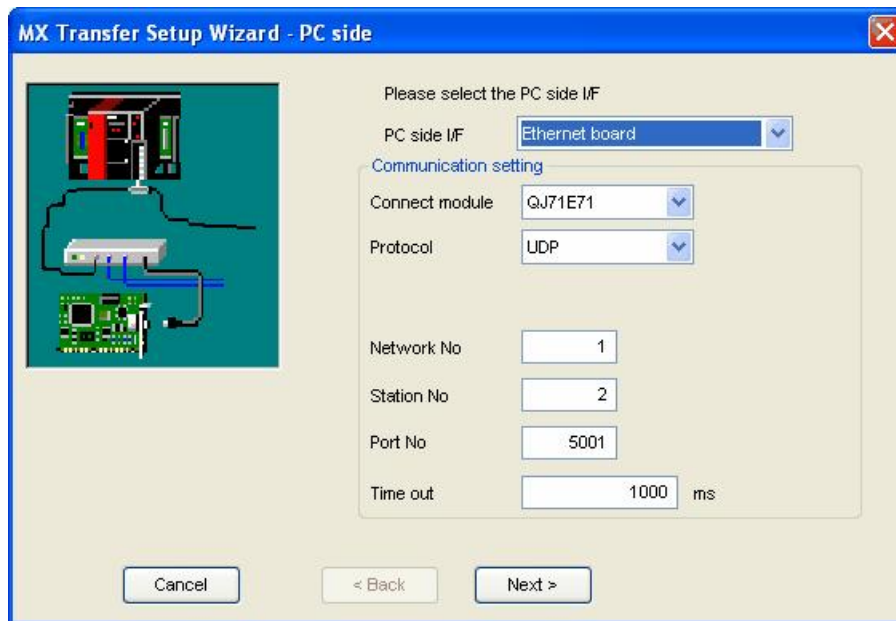
**MX Transfer Setup dialog**

3. The **New Transfer Setup dialog**<sup>[93]</sup> will appear. Enter a name for the new transfer setup (*Ethernet* in the picture below) and select *<Default...>* as the template, then click **OK**.



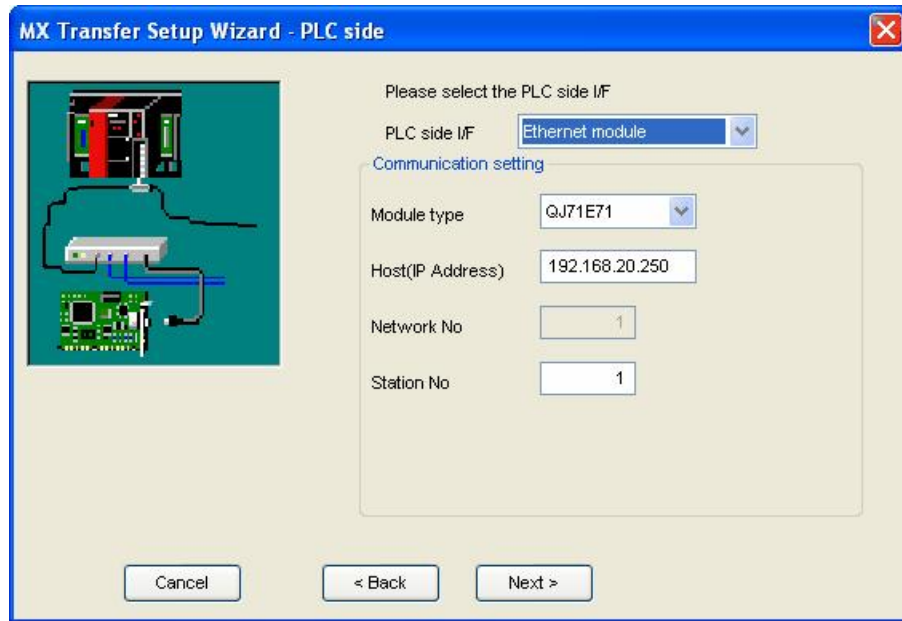
**Entering a transfer setup name**

4. The new transfer setup can now be edited using the **Communication Setting Wizard**, as shown in the figure below. Specify the communication type to use on the PC side, as shown in the figure below. In the **PC Side I/F** field, select the **Ethernet board** connection type from the drop-down list. Select a module number from the **Connect module** drop-down list. Click **Next** to continue.



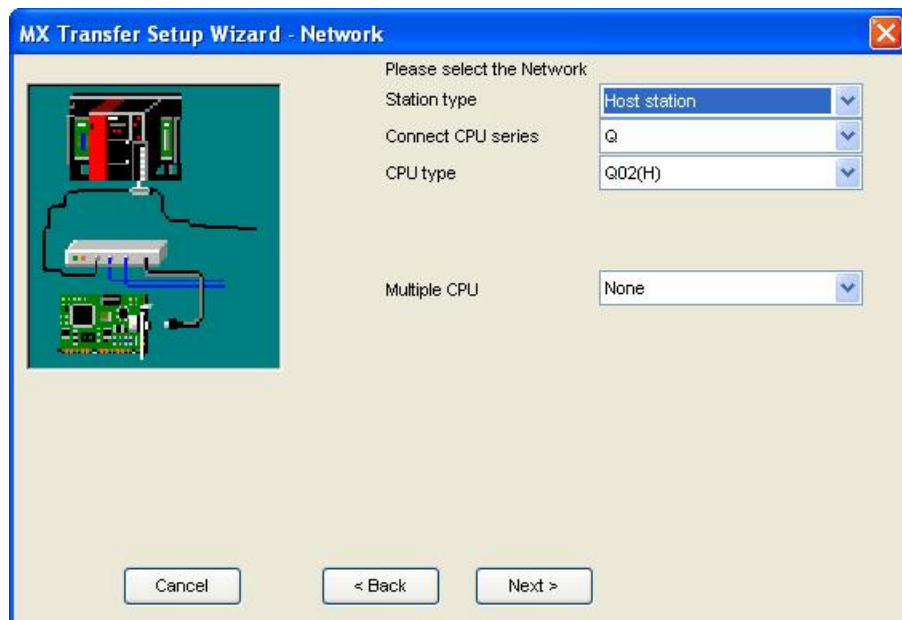
**Setting up the Connection for the PC Side**

- Now you must specify the communication type to use on the PLC (device) side, as shown in the figure below. In the **PLC Side I/F** field, select **Ethernet module** from the drop-down list as the connection type to use for the selected channel. Specify a Host name and a Port number. Click **Next** to continue.



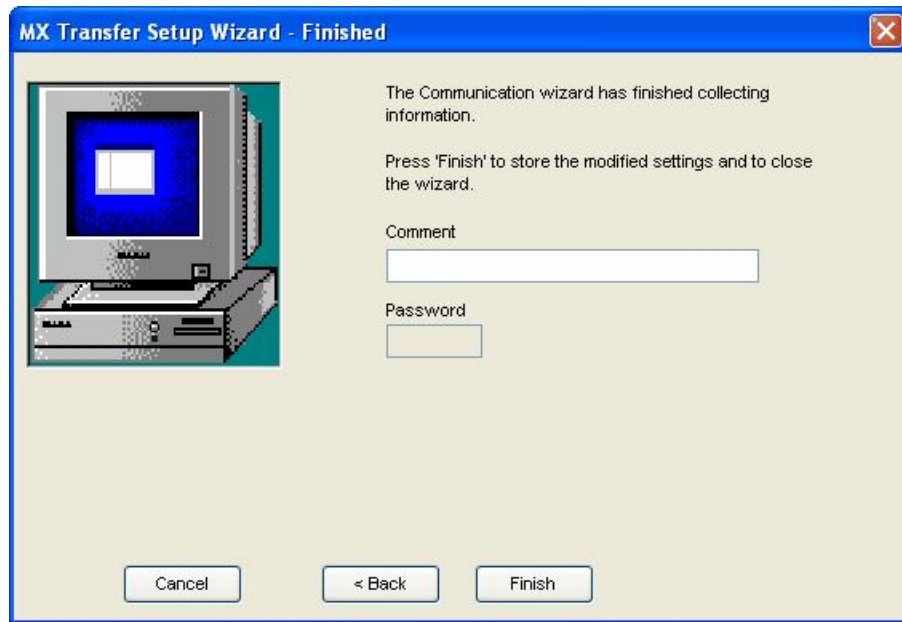
**Setting up the Connection for the PLC Side**

- Select the network **Station Type** and **CPU Type** (if applicable) from the respective drop-down lists, as shown in the figure below. Click **Next** to continue.



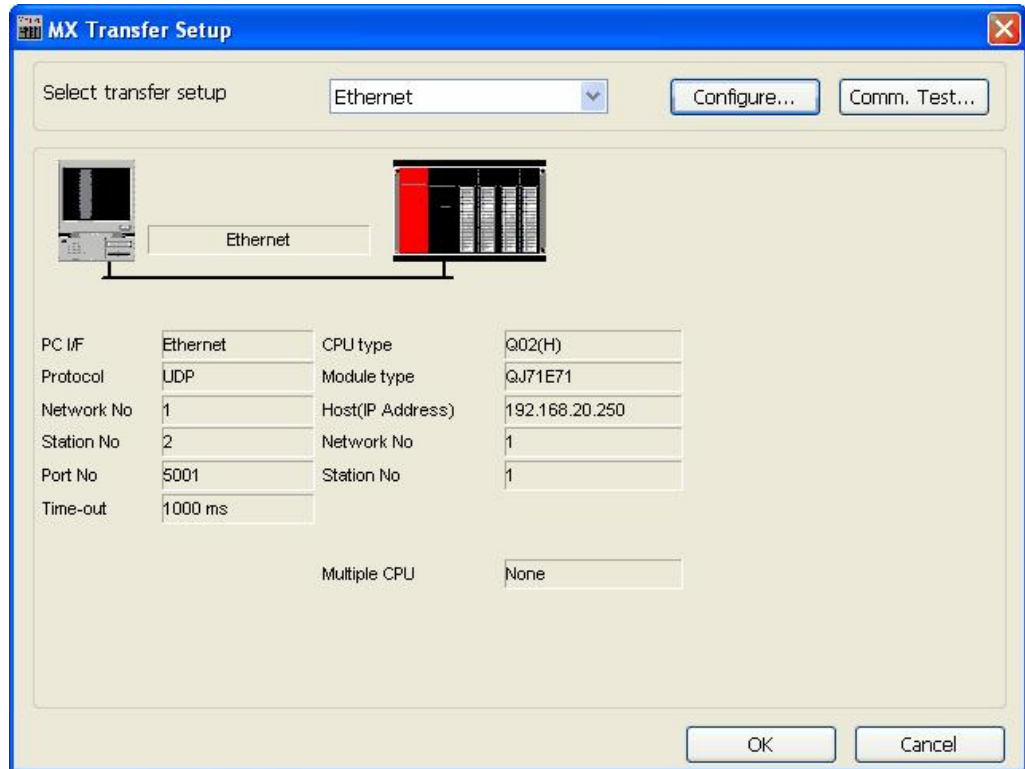
**Configuring Network Settings**

- 7. Click the **Finish** button to complete the communication channel setup, as shown in the figure below.



**Completing Channel Setup**

8. The **MX Transfer Setup** dialog box appears with the new transfer setup selected, as shown in the figure below. Each network node is shown graphically, with the node properties shown under the graphic. The left-hand side of the dialog lists the configuration properties for the PC side. You can double-click the image of a node (or double-click the properties list) to edit the configuration properties. The right-hand side of the dialog lists the configuration properties for the PLC device side. You can double-click the image of the PLC device (or double-click the configuration properties list) to edit the PLC device side configuration properties. You can also click the **Configure** button to return to the **Communication Setting Wizard**. Click the **OK** button to save the changes.



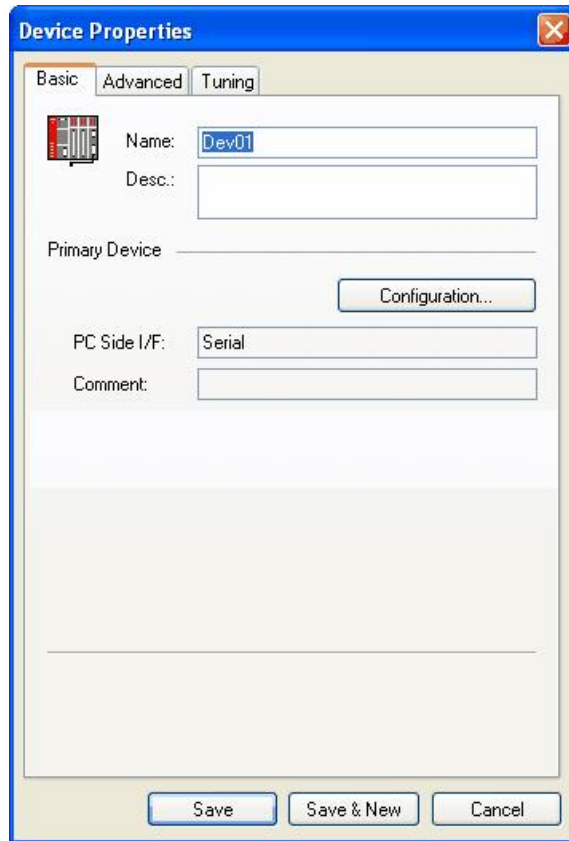
**MX Transfer Setup dialog (configured)**

**Note:** For complete information about channel configuration properties and the Communication Setting Wizard, see "[Configuring the address space](#)"<sup>[64]</sup>.

### 1.7.4 Configuring a New Device

In the previous section, we used the **Communication Setting Wizard** to set up Serial and Ethernet communication channels between the PC and PLC. Once you have completed the wizard:

1. The **Basic** tab of the **Device Properties** dialog box appears, as shown in the figure below. The communication type (e.g. Serial, Ethernet, etc.) is shown in the **PC Side I/F** field, as shown in the figure below.



**Configuring Device Properties**

2. In the **Name** field, type a name for the new device, and type a description for the device (optional). To specify which connection type and which physical connection (COM) port to use for the selected device, click the **Configuration** button to launch the Communication Setting Wizard for the Primary device.



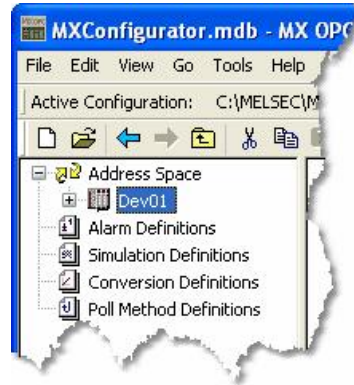
- Click on the **Advanced** tab to set the advanced parameters for the **Primary** and **Backup** devices, as shown in the figure below.

The screenshot shows the 'Device Properties' dialog box with the 'Advanced' tab selected. The dialog is divided into two sections: 'Primary Device' and 'Backup Device'. Each section has an 'Enable Device' checkbox (checked) and a 'Simulate' checkbox (unchecked). The 'Primary Device' section also includes a 'Configuration...' button. Both sections have three input fields: 'Reply timeout' (1000 ms), 'Delay time' (0 ms), and 'Max count for R/W operation' (16 words). The 'Number of retries' is set to 3, and the 'Merge gap' is set to 64 words. At the bottom, there are three buttons: 'Save', 'Save & New', and 'Cancel'.

**Device Properties: Advanced Tab**

- Check the **Enable Device** check box to activate the device.  
**Note:** For complete information about device configuration properties, see "[Configuring the address space](#)"<sup>[64]</sup>.

- When you have finished configuring the device properties, click the **Save** button. The new device appears under the **Address Space** tree control, as shown in the figure below.



**New Device Created for the Channel**

**Note:** For complete device configuration properties info, see ["Configuring the address space"](#)<sup>[64]</sup>.

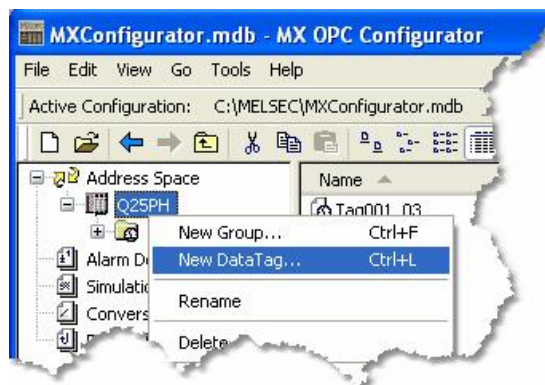
### 1.7.5 Creating Data Tags

Now you can create a new data tag that the OPC clients can request from the server.

**Note:** Data tags can be logically organized into **groups** (folders). You can configure as many folders as required. You can even create subfolders for each group to create a hierarchical organization of tags. See ["Groups"](#)<sup>[123]</sup> for details.

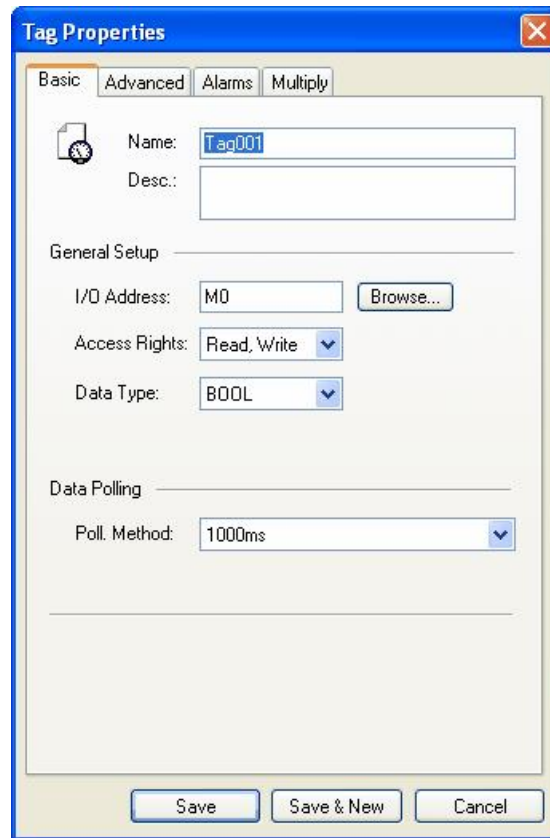
To create a new tag:

- Right-click on a device in the tree control of the Configurator screen and select **New Data Tag** from the pop-up menu, as shown in the figure below.



**Adding a New Data Tag**

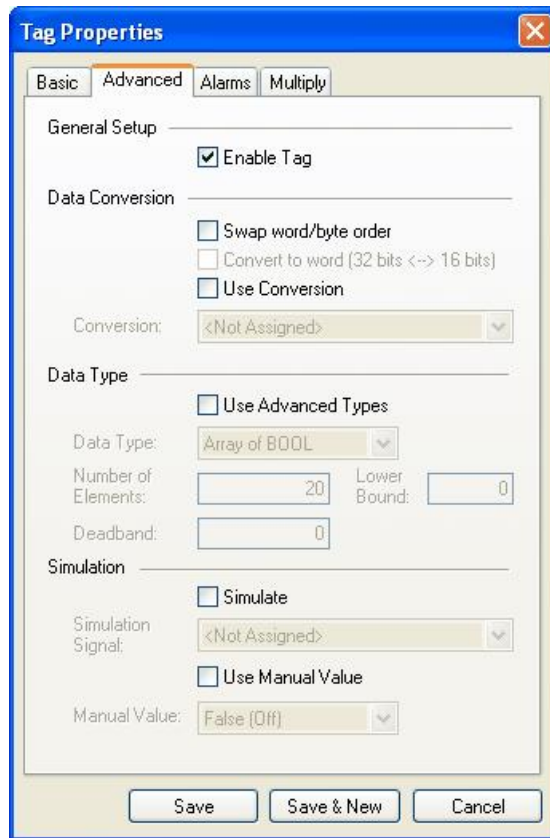
2. The **Basic** tab of the **Data Tag Properties** dialog box appears, as shown in the figure below.



**Configuring Data Tag Properties**

3. In the **Name** field, type a name for the new tag, and type a description for the tag (optional).
4. Set the parameters for the **I/O Address** and data **Access Rights**.

5. Click on the **Advanced** tab. Check the **Enable Tag** check box to activate the tag.



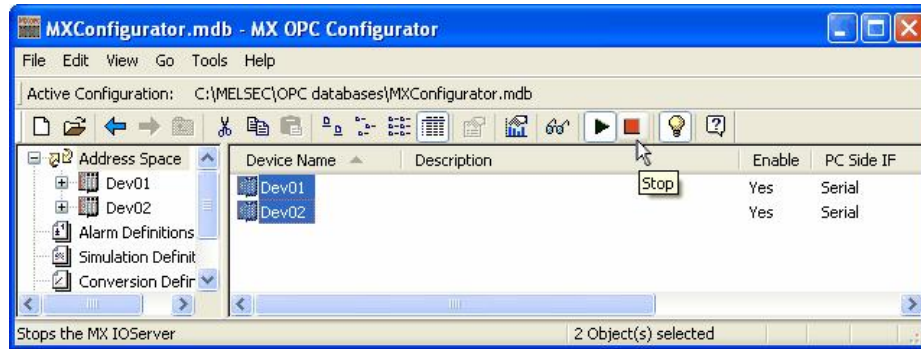
**Data Tag Properties: Advanced Tab**

6. Set the **Data Conversion** and **Data Type** parameters for the tag.
7. In the **Alarms** tab of the **Tag Properties** dialog box, you can check the **Generate Alarms** check box to make the server generate a limit alarm and/or a digital alarm based on the data item value.
8. When you have finished configuring the tag properties, click the **Save** button. The new tag appears under the **Device** tree control.

**Note:** For complete information about data tag configuration properties, see "[Data tags](#)"<sup>[105]</sup>.

### 1.7.6 Starting the MX OPC Runtime Driver

Once you are finished with your device and data tag configuration, you can start the MX OPC Runtime driver by clicking the Start button (green triangle icon) on the Standard toolbar of the MX OPC Configurator. This enables client/server communication. To stop the driver, click the Stop button (red square icon) on the Standard toolbar, as shown in the figure below.



Starting and Stopping the Driver

## 1.7.7 Connecting to the Server Using OPC DataSpy

### Introduction to OPC DataSpy

With the introduction of the OPC Foundation Plug and Play communications standard called OPC Data Access, and recently the OPC Alarm and Events Specifications, a universal diagnostics and analysis utility was needed by industry. This need is filled with OPC DataSpy.

### Features of OPC DataSpy

OPC DataSpy includes the following key standard features:

- Internet OPC over TCP/IP tag browsing.
- GenBroker Monitor.
- Simple-to-use OPC test client inspection and diagnostics utility.
- Determination of whether server is OPC compliant.
- Available as a stand-alone OPC test client.
- OPC Data Access (DA) test client.
- OPC Alarm and Events (AE) test client.
- OPC Data Access troubleshooting tool.
- OPC Alarm and Events troubleshooting tool.
- OPC-compliance testing of third-party servers.
- Determination of OPC standards for OPC servers.
- OPC loading and OPC traffic analysis.

The main concept behind the OPC DataSpy is to provide an easy-to-use OPC Test Client to test, diagnose, and troubleshoot industrial applications using the OPC Foundations, OPC Data Access, and Alarm and Events specifications.

The OPC DataSpy tree control capability consists of three main parts:

1. OPC Item Browsing
2. OPC Data and Alarm Monitors
3. GenBroker Monitor

The OPC item browser mode allows you to browse for OPC-compliant Data Access and OPC Alarm and Event servers, as well as OPC Historical Data Access servers, and to provide basic information allowing users to view, test, and troubleshoot any OPC third-party servers. The OPC monitors provide real-time monitoring of OPC server data.

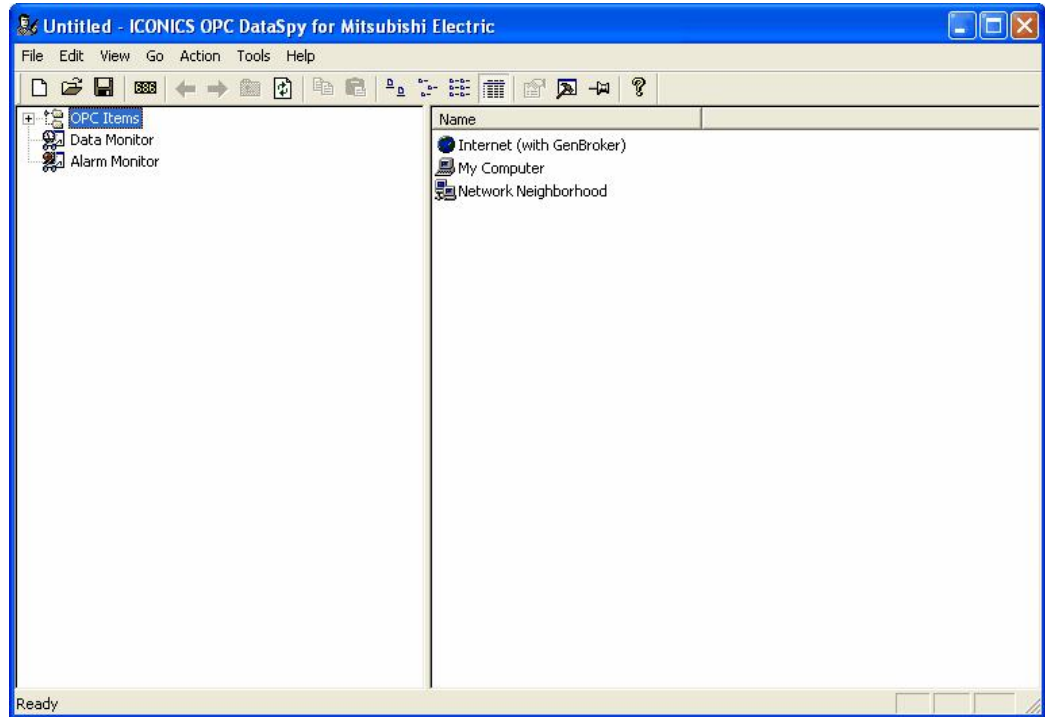
The OPC DataSpy user interface consists of the following elements:

- OPC Item browser
- OPC Data and Alarm Monitors
- Diagnostics and statistics and logging module (output window)

Once the MX OPC Runtime driver is running, you can connect to the MX OPC Server via OPC DataSpy. For this Quick Start, you will use OPC DataSpy provided with the MX OPC Server installation:

1. Open OPC DataSpy from the Windows **Start** menu by selecting **[All] Programs > MELSOFT Application > MX OPC Server 6.05 > OPC Data Spy > DataSpy.exe**. This opens OPC DataSpy, as shown in the figure below.

**Note:** The name of the program folder may vary depending on your local settings (e.g. language settings).



OPC DataSpy Screen

2. From the left navigation pane, expand the **"OPC Items"** item by clicking on the "[+]" symbol to the left of the icon. Expand the **"My Computer"** section then the **"OPC Data Access"** section then the **"Mitsubishi.MXOPC.6"** section.

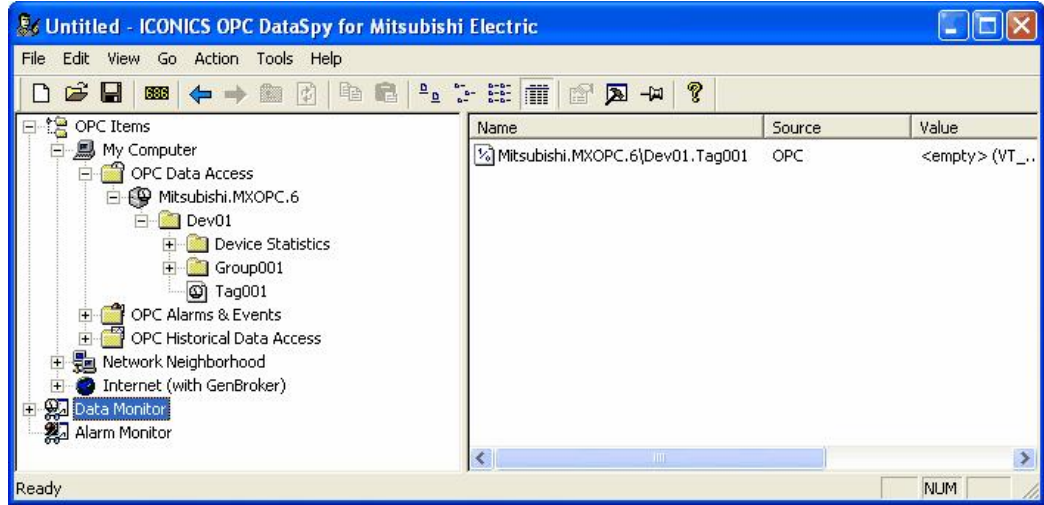
Users will then see all configured devices listed underneath **"Mitsubishi.MXOPC.6"** and can browse groups and data tags accordingly.

3. Right click on any selected tag listed beneath **"Mitsubishi.MXOPC.6"** and click on **"Monitor"**. Next, click on **"Data Monitor"** that is listed underneath OPC Items when that section is fully collapsed. In the right navigation pane, users will be able to see the values associated with selected tags.
4. If your client application does not support reading a single bit within a word or reading one element of an array, you may still be able to use these features by changing the name of the item that you add.

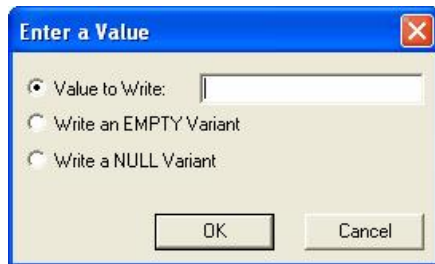
To access a bit within a word, define a new tag with an address such as D12.1 (bit 1 within the word at address D12, with bit numbering starting a 0 for the least significant bit) and set the type to BOOL. If write access is not required, a client can also address a bit within a word tag using the item syntax "tagname.bitNr" (e.g. tagname.0 for bit 0) - this is a read-only custom "OPC property" of the tag item.

A client can address a single element of an array tag using the item syntax 'arraytag[elemNr]' (e.g. arraytag[2] for element 2).

- To Write to Tags, right-click on the tag name in the right navigation pane. The "Enter a Value" dialog pops up. Users can then enter values in the text entry box.



**Note:** If you are writing a floating point value, the period (.) character is always used as the radix character (e.g. '123.4' not '123,4'). If you are writing to an array item, the values should be separated by commas (,).



## 2 Using the Configurator

This section will describe how to get started with the MX OPC Configurator, from creating new databases through to the menu and toolbar items available.



## 2.1 Starting the Configurator

To start the MX OPC Configurator:

1. From the Windows **Start** menu, select **[All] Programs > MELSOFT Application > MX OPC Server 6.05 > MX OPC Configurator**.

**Note:** The name of the program folder may vary depending on your local settings (e.g. language settings).

2. This opens the Configurator, as shown in the figure below. The screen consists of a split window with a tree control view in the left-hand pane and a configuration view in the right-hand pane. The Configurator provides a standard format for the configuration database, as well as a sample (default) configuration project. The Configurator also includes a toolbar and menus with many command functions.

**Note:** When the **Display Hints** check box on the **General** tab of the **Options** dialog box (see the **Tools** menu) is checked, helpful tips are displayed in the configuration view, as shown in the figure below.

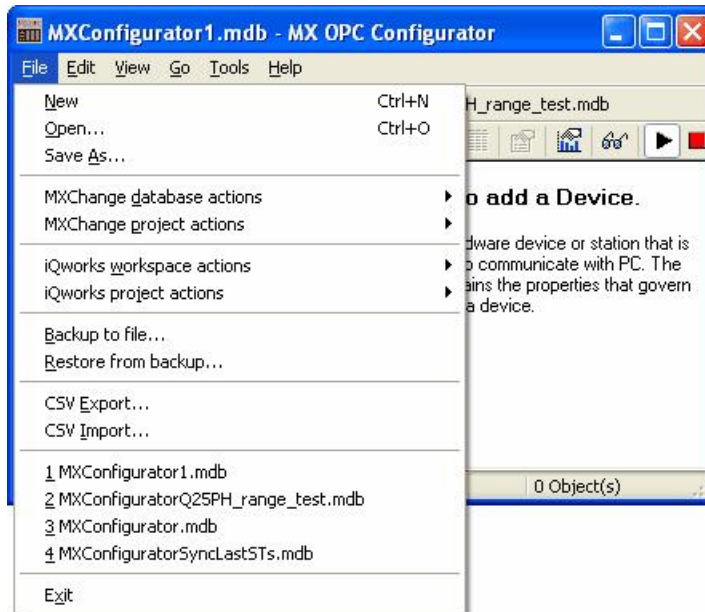


Configurator Screen

## 2.2 Creating Configuration Databases

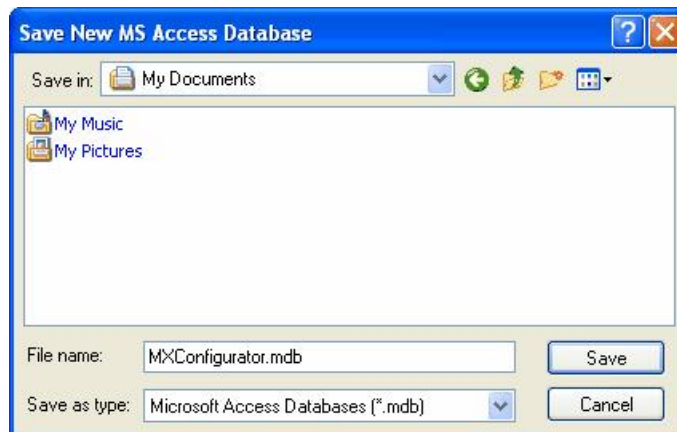
The Configurator uses Microsoft Access configuration databases. To create a new configuration database in the Configurator:

1. Select **New** from the **File** menu, as shown in the figure below.



**Creating a Configuration Database**

2. In the Save New Database dialog box, select the database type from the **Save As Type** drop-down list, as shown in the figure below. For a Microsoft Access database, the Configurator uses a single .mdb file. Browse for the target directory, give the file a name, and then click the Save button.






















**Saving the New Configuration Database**

## 2.3 Toolbars

The Configurator contains two toolbars: a **Standard toolbar** and a **Data Manipulation toolbar**.

### Standard Toolbar

To show or hide the **Standard toolbar**, select **Toolbars > Standard Buttons** from the **View** menu. The **Standard toolbar**, shown below, contains the following command buttons.

-  **New:** Creates a new configuration database.
-  **Open:** Opens an existing configuration database.
-  **Back:** Moves the cursor back to the previously selected item.
-  **Next:** Moves the cursor to the next item in the tree control.
-  **Up One Level:** Moves up one level in the tree control.
-  **Cut:** Deletes current selection, sending it to the clipboard.
-  **Copy:** Copies the current selection to the clipboard.
-  **Paste:** Pastes the current contents of the clipboard.
-  **Large Icons:** Displays items as large icons.
-  **Small Icons:** Displays items as small icons.
-  **List:** Displays items as a list.
-  **Details:** Displays items as a list with details.
-  **Properties:** Shows the properties for the selected item.
-  **Statistics Mode:** Switches the current dialog view to data statistics mode.
-  **Monitor View:** Displays OPC server data in a separate pane.
-  **Start:** Starts the MX I/O server.
-  **Stop:** Stops the MX I/O server.
-  **Make Active:** Activates the current configuration for use by the runtime server.
-  **Help Topics:** Launches online Help for the application.

## Data Manipulation Toolbar

To show or hide the **Data Manipulation** toolbar, select **Toolbars > Data Manipulation Buttons** from the **View** menu. The **Data Manipulation** toolbar, shown below, contains the following command buttons.



**New Simulation Definition:** Creates a new simulation signal.



**New Limit Alarm Definition:** Creates a new limit alarm definition.



**New Digital Alarm Definition:** Creates a new digital alarm definition.



**New Conversion Definition:** Creates a new conversion definition.



**New Poll Method Definition:** Creates a new poll method definition.



**New Device:** Inserts a new device under the **Address Space** tree control.



**New Modbus Device:** Inserts a new Modbus device under **Address Space**.



**New Group:** Inserts a new group for the selected device.



**New Data Tag:** Inserts a new data tag for the selected device.

## 2.4 Menus

The menu bar of the Configurator contains the following menus:

- [File](#)<sup>[49]</sup>
- [Edit](#)<sup>[52]</sup>
- [View](#)<sup>[54]</sup>
- [Go](#)<sup>[55]</sup>
- [Tools](#)<sup>[56]</sup>
- [Help](#)<sup>[63]</sup>

**Note:** You can also access many of the menu commands by right-clicking items in the tree control of the Configurator and selecting command functions from the pop-up menus.

### 2.4.1 File Menu

The **File** menu commands are listed as follows.

- **New** Creates a new configuration database.
- **Open** Opens an existing configuration database Microsoft Access (.mdb) file.
- **Save As** Saves the current database under a different name as a Microsoft Access (.mdb) file.
- [MXChange Database Actions](#)<sup>[50]</sup> These menu items are used to perform actions on an MXChange Database (a group of projects), such as opening or creating an MXChange Database and changing your security password.
- [MXChange Project Actions](#)<sup>[50]</sup> These menu items are used to perform actions on the current MX OPC Server project within MXChange, such as creating new devices by importing the tags from a GX IEC Developer project, or synchronising I/O references with the MXChange database.
- [iQ Works Workspace Actions](#)<sup>[51]</sup> These menu items are used to perform actions on an iQ Works workspace (a group of projects), such as connecting to an iQ Works workspace or removing unused projects from the workspace.
- [iQ Works Project Actions](#)<sup>[51]</sup> These menu items are used to perform actions on the current MX OPC Server project within iQ Works, such as creating new devices by importing the labels from a GX Works2 project, or synchronising I/O references with the iQ Works workspace.
- **Backup to File** Saves the present configuration to a new or existing Microsoft Access Database (.mdb).
- **Restore from Backup** Loads prior configuration from Microsoft Access Database (.mdb).
- **CSV Export** Exports configuration data from your database to a text file (.txt) or a Microsoft Excel file (.csv). You can specify the delimiters and what to export. For more information, see [Exporting configuration data to a CSV file](#)<sup>[127]</sup>.
- **CSV Import** Imports data into your configuration database from a text file (.txt) or a Microsoft Excel file (.csv). You can then specify the delimiters and choose from the import settings. This is described in [Importing configuration data from a CSV File](#)<sup>[128]</sup>.
- **Recent projects** The last four opened projects will be shown in the **File** menu. Select one of these projects to open it.
- **Exit** Closes the application. The current configuration database is automatically saved.

### 2.4.1.1 MXChange database actions

These menu items are used to perform actions on an MXChange Database (a group of projects), such as opening or creating an MXChange Database and changing your security password. For more information, see [Using MXChange](#)<sup>[173]</sup>.

The **MXChange database actions** menu commands are listed as follows.

- New            Create a new MXChange database and link it to the MX OPC Configurator project.
- Open           Open an existing MXChange database and link it to the MX OPC Configurator project.
- Save as        Save the MXChange database with a new name.
- Delete         Delete the MXChange database and all the projects it contains.  
**NOTE:** This destroys *ALL* application data in the MXChange database, not just the MX OPC Configurator project data. Do not use this option unless you intend to destroy all data in the MXChange server database for *ALL* applications.
- Compact       Compacts the MXChange server database.
- Change  
  password      Change the password required to use the MXChange server database.
- Close          Close the MXChange server database and disconnect the current project from MXChange.

### 2.4.1.2 MXChange project actions

These menu items are used to perform actions on the current MX OPC Server project within MXChange, such as creating new devices by importing the tags from a GX IEC Developer project, or synchronising I/O references with the MXChange database. For more information, see [Using MXChange](#)<sup>[173]</sup>.

- Load project  
  from  
  MXChange      Loads an MX OPC server project from the backup copy stored with the project in MXChange.
- Show  
  changes        Show changes made to linked tags by other applications such as GX IEC Developer - see [Synchronising with MXChange](#)<sup>[186]</sup>.
- Synchronise    Update the MX OPC server configuration and MXChange database with recent changes to tags - see [Synchronising with MXChange](#)<sup>[186]</sup>.
- Import GID  
  projects        Import one or more PLC projects from the MXChange database as new devices in the MX OPC server project. This is described in [Importing from MXChange](#)<sup>[184]</sup>.
- Export  
  devices to  
  GID            Export one or more existing MX OPC devices to the MXChange database as new GX Developer or GX IEC Developer projects. For more information, see [Exporting to MXChange](#)<sup>[182]</sup>.
- Delete project Completely deletes the MX OPC server project from MXChange.

### 2.4.1.3 iQ Works workspace actions

These menu items are used to perform actions on an iQ Works workspace (a group of projects), such as opening an iQ Works workspace. For more information, see [Using iQ Works](#)<sup>[193]</sup>.

The **iQ Works workspace actions** menu commands are listed as follows.

- Open workspace      Open an existing iQ Works workspace and link it to the MX OPC Configurator project. See [Connecting to iQ Works](#)<sup>[193]</sup>.
- Close workspace      Close the iQ Works workspace and disconnect the current project from iQ Works. For more information, see [Detaching from iQ Works](#)<sup>[202]</sup>.
- Delete unused projects      Select unused MX OPC Configurator projects to be permanently deleted from the workspace. This is described in more detail in [Deleting unused projects](#)<sup>[223]</sup>.

### 2.4.1.4 iQ Works project actions

These menu items are used to perform actions on the current MX OPC Configurator project within iQ Works, such as creating new devices by importing the tags from a GX Works2 project, or synchronising I/O references with the iQ Works workspace. For more information, see [Using iQ Works](#)<sup>[193]</sup>.

- Show changes      Show changes made to linked tags by other applications such as GX Works2 - see [Change notification](#)<sup>[217]</sup>.
- Synchronise      Update the MX OPC server configuration and iQ Works workspace with recent changes to labels - see [Synchronising with iQ Works](#)<sup>[219]</sup>.
- Import GX Works2 project      Import one or more PLC projects from the iQ Works workspace as new devices in the MX OPC Configurator project. This is described in [Importing from iQ Works](#)<sup>[210]</sup>.
- Export devices to GX Works2      Export one or more existing MX OPC devices to a GX Works2 project in the iQ Works workspace. For more information, see [Exporting to iQ Works](#)<sup>[203]</sup>.
- Unlink device      Breaks the link between a local device and a GX Works2 project in the iQ Works workspace. See [Unlinking devices from iQ Works](#)<sup>[220]</sup>.
- Delete project      Completely deletes the current MX OPC Configurator project from the iQ Works workspace. For more information, see [Detaching from iQ Works](#)<sup>[202]</sup>.
- Import CSV      Use this to connect to iQ Works with settings provided by another application. See [Using MXCSV files](#)<sup>[224]</sup>.

## 2.4.2 Edit Menu

The **Edit** menu commands are listed in the table below.

### Edit Menu Commands

Command	Shortcut Keys	Function
<a href="#">New MX Device</a> [77]	CTRL+E	Creates a new MX item.
<a href="#">New Modbus Device</a> [87]		Creates a new Modbus item.
Cut	CTRL+X	Cuts the selected object from the view and places it on the clipboard.
Copy	CTRL+C	Copies the selected object to the clipboard.
Paste	CTRL+V	Pastes the last object placed on the clipboard.
Select All	CTRL+A	Selects all objects in a list. The selection is shown in the upper-right-hand section of the viewer.
Invert Selection		Unselects all selected items and selects all unselected items.
Delete	DEL	Deletes the selected object.
Properties	Enter	Shows the properties for the selected item.

## 2.4.3 Rename

To rename an item, right click on it in the tree view, select **Rename** from the pop-up menu and type in the new name.

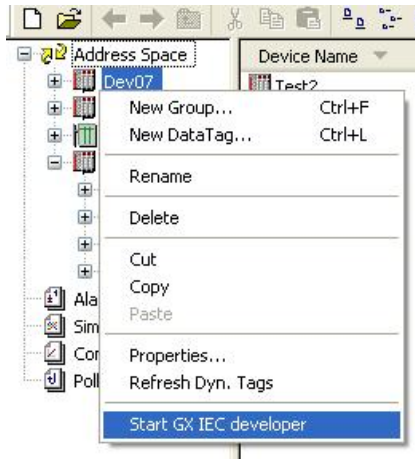
Any application requesting data from the I/O driver uses the device and item names to access points on the device. Each device that the driver communicates with should have a unique device name regardless of the device's channel.

The device name can be up to 32 alphanumeric characters, including underscores ( \_ ) and hyphens ( - ). It cannot begin with a digit.



### 2.4.4 Start GX IEC developer

In projects that are linked to [MXChange](#)<sup>[173]</sup>, some MX devices may correspond to GX IEC developer projects stored on the same computer. When you need to make changes to your system that require the use of GX IEC developer (such as some modifications to [DUT tags](#)<sup>[183]</sup>), you can right-click on the device in the tree view and select **Start GX IEC developer** from the pop-up menu. This will run GX IEC developer and immediately load the relevant project file.



**Starting GX IEC Developer**

## 2.4.5 View Menu

The **View** menu commands are listed in the table below.

### View Menu Commands

Command	Shortcut Keys	Function
Standard Buttons		Display buttons for standard use of application.
Data Manipulation		Display buttons related to using data within the application.
Toolbars		Shows/hides the Configurator toolbar buttons.
Active Configuration Bar		Shows/hides the database bar, which shows the file name and directory path for the active configuration database.
Status Bar		Shows/hides the status bar at the bottom of the Configurator window.
Large Icons	F7	Displays items as large icons.
Small Icons	F8	Displays items as small icons.
List	F9	Displays items as a list.
Details	F10	Displays items as a list along with detailed information about the configuration of each item.
Statistics Mode	F11	Switches the current dialog view to data statistics mode.
Monitor View	F12	Displays OPC server data in a separate pane.
Diagnostics		Displays Start time, Up time, Last update time, Group count, Server status, Client count, Version and Vendor info.
Sort By		Displays a list of options for sorting the columns in the right-hand pane of the screen. The options listed depend on the level within the view.
Show/Hide Columns		Displays a list of columns that you can choose to show or hide in the view.
Refresh	F5	Refreshes the data for the entire Configurator screen.

## 2.4.6 Go Menu

The **Go** menu commands are listed in the table below.

### Go Menu Commands

Command	Shortcut Keys	Function
Back	CTRL+ALT+ Left Arrow	Moves the cursor back to the previously selected item in the tree control.
Forward	CTRL+ALT+ Right Arrow	Moves the cursor to the next item in the tree control.
Up One Level		Moves the cursor up one level in the tree control.
Next Item	ALT+Down Arrow	Moves the cursor to the next visible item in the tree control.
Previous Item	ALT+Up Arrow	Moves the cursor to the previous visible item in the tree control.
Expand Item	ALT+Right Arrow	Expands an item that contains a submenu.
Collapse Item	ALT+Left Arrow	Collapses an item that contains a submenu.
Page Up	ALT+PgUp	Moves the cursor up to the first item in the tree.
Page Down	ALT+PgDown	Moves the cursor down to the last visible item in the tree.
Home	ALT+Home	Moves the cursor up to the first item in the tree.
End	ALT+End	Moves the cursor down to the last visible item in the tree.
Address Space	F2	Moves the cursor to the <b>Address Space</b> tree control.
Alarm Definitions	F3	Moves the cursor to the <b>Alarm Definitions</b> tree control.
Simulation Definitions	F4	Moves the cursor to the <b>Simulation Definitions</b> tree control.
Conversion Definitions	SHIFT+F3	Moves the cursor to the <b>Conversion Definitions</b> tree control.
Poll Method Definitions	SHIFT+F4	Moves the cursor to the <b>Poll Method Definitions</b> tree control.
Next Pane	F6	Moves the cursor to the next pane in the window.
Previous Pane	SHIFT+F6	Moves the cursor to the last pane used.

### 2.4.7 Tools Menu

The **Tools** menu commands are listed in the table below.

#### Tools Menu Commands

Command	Function
<a href="#">Options</a> <sup>[56]</sup>	Launches the <b>Options</b> dialog box.
<a href="#">Compact/Repair MS Access Database</a> <sup>[62]</sup>	Opens the <b>Compact/Repair MS Access Database</b> dialog box.

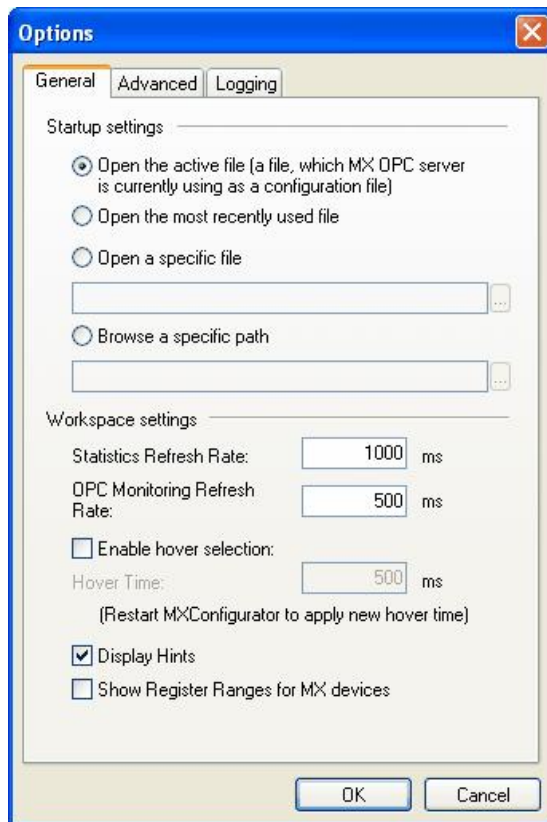
#### 2.4.7.1 Options

Selecting **Options** from the **Tools** menu opens the **Options** dialog box, which contains the following tabs:

- [General](#)<sup>[56]</sup>
- [Advanced](#)<sup>[58]</sup>
- [Logging](#)<sup>[60]</sup>

##### 2.4.7.1.1 General Tab

The **General** tab of the **Options** dialog box, shown in the figure below, sets the startup and workspace parameters for the Configurator.



**Options Dialog Box: General Tab**

### Startup Settings

Under the **Startup Settings** section, select one of the following:

**Open the active file:** Launches the currently active configuration database upon startup.

**Open the most recently used file:** Launches the recently opened configuration database upon startup.

**Open a specific file:** Specifies a configuration database to launch upon startup. To select a database, click the ... button next to the text box and browse for the file. When this option is selected, the "active" database is overridden by the specified database.

**Browse a specific path:** Specifies a file path to launch upon startup. To select a path, click the ... button next to the text box and browse for the path.

### Workspace Settings

The following **Workspace Settings** are available:

**Statistics Refresh Rate:** Specifies the update frequency (in milliseconds) of the items in the Statistics View.

**OPC Monitoring Refresh Rate:** Specifies the update frequency (in milliseconds) of the items in the Monitor View. The monitor scans the server and displays the tag values at the bottom of the Configurator screen.

**Enable hover selection:** Checking this option allows you to highlight an item by moving the mouse pointer over that item and keeping it there for a specified **Hover Time** (in milliseconds). If you change the hover time, the new hover time will be applied when the MX OPC Configurator is restarted.

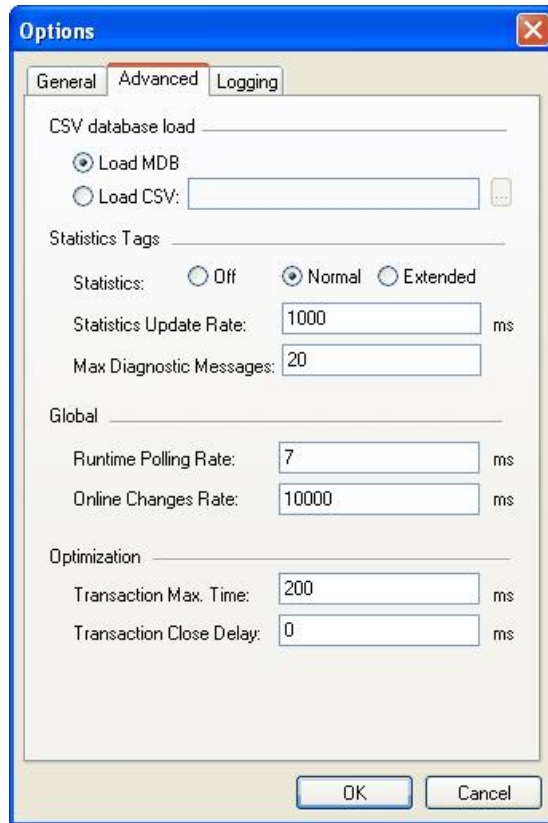
**Display Hints:** When this check box is checked, helpful tips are displayed in the configuration view.

**Show Register Ranges for MX devices:** The new server runtime will be able to provide the list of possible register ranges of each MX device. This list will be provided to the OPC clients as a special group called "Hints" in the address space of the device containing tags. The names of these tags will be in the form <register><starting address>-<ending address>, where all three items are in the same form as seen in the Supported Devices dialog (clicking Browse from Tag Properties Basic tab). D0.0-7999.15 is a valid example for the FX3U(C) CPU type. This function can be enabled or disabled through the Options dialog.

## 2.4.7.1.2 Advanced Tab

The **Advanced** tab of the **Options** dialog box, shown in the figure below, configures the advanced parameters for your driver.

**NOTE: Advanced settings are for fine-tuning your driver and should not be changed unless you have an intimate knowledge of how the driver operates and know that you need to make some adjustments.**



Options Dialog Box: Advanced Tab

#### CSV database load

- **Load MDB:** This selection will load the configuration from the active MS Access database.
- **Load CSV:** If selected, the runtime will load the specified CSV file.

#### Statistics tags

In a similar way to the "Hints" described in the ['General tab' section](#)<sup>[56]</sup>, the MX Runtime can optionally expose statistics tags in a group called "Device Statistics". The options in this section are:

- **Statistics** - This sets the level of detail available, and can be either:
  - **Off** - no statistics branch is available in the runtime
  - **Normal** - basic statistics tags will be available
  - **Extended** - detailed statistics tags are available
- **Statistics update rate** - the rate at which the statistics tags are updated, in milliseconds
- **Max diagnostics messages** - the diagnostics messages queue length.

For a more detailed explanation of the statistics tags, and a list of the tags added for each **Statistics** level, see [Statistical tags in OPC address space](#)<sup>[151]</sup>.

### Global

In the **Global** section, you can configure:

- the **Runtime polling rate** - this is the rate at which the runtime system looks for new data, in milliseconds. Smaller numbers mean that the checks will be performed more frequently, at a cost of higher CPU usage. Larger numbers mean less frequent checks, but lower CPU usage. This value should normally be left at the default setting.
- the **Online changes rate** - this is the rate at which the runtime module looks for changes to the configuration, in milliseconds. As new tags are not often added to a database after it has been configured, a much longer interval is used than the **Runtime polling rate**. This value should normally be left at the default setting.

### Optimization

The settings in the **Optimization** section are:

- **Transaction max time** the maximum duration of one transaction (a set of read/write/request point operations, executed together).
- **Transaction close delay** - the time that the runtime waits for another request before it closes the transaction.

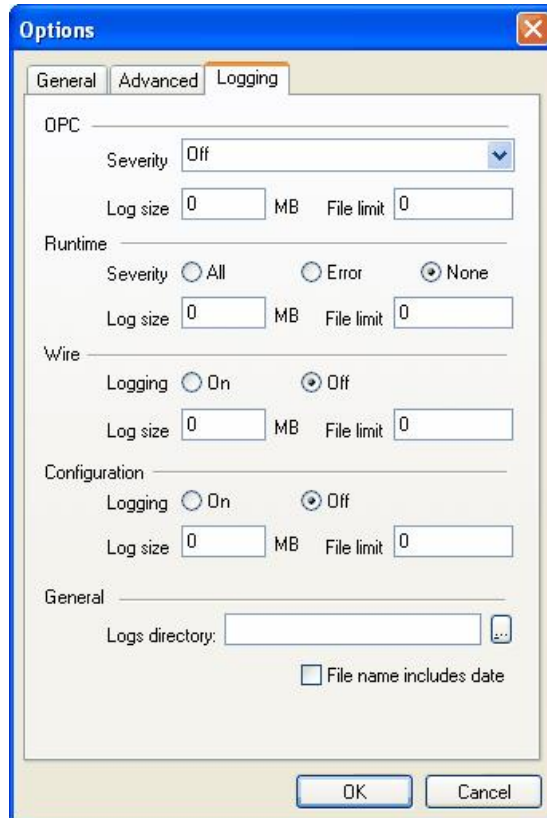
## 2.4.7.1.3 Logging Tab

MX OPC Configurator allows the logging of diagnostic information to the disk.

The logger is a simple logging tool built in to the Runtime and the Configurator. All changes that are made to the settings on the following option page will affect only diagnostic messages that are generated from this point on. For example if logging is disabled and the user enables it at 12:00, no messages will be stored to the log files that occurred before 12:00, even if they are shown in the Statistics dialog among the Last Transaction Statuses.

The log files are created on demand, i.e. the log file will only be created on the disk the first time a message is logged.

Logging can be enabled from the **Logging** page of the **Options** dialog, as shown in the following screenshot:



**Options Dialog Box: Logging Tab, 'General Runtime Diagnostics' Section**

This page is divided into five sections:

- **OPC** - logging settings for OPC layer
- **Runtime** - logging settings for the runtime module
- **Wire** - logs communication with the hardware (also called 'Protocol & Wire' settings in earlier versions of MX OPC Configurator)
- **Configuration** - logging settings for configuration changes to the database such as adding, deleting or modifying items (also called 'User changes logging' in earlier versions of MX OPC Configurator)
- **General** - general settings that apply to all types of logging



### OPC settings

- **Severity** - the severity of OPC layer messages to log. The possible settings are:
  - **Off** - Do not log any messages
  - **Severe errors** - Very serious errors
  - **Errors** - Errors that could affect program operation (e.g. failure to read from the configuration database)
  - **Warnings** - Errors that do not stop operation from continuing (e.g. failure to read a list of PLC ranges)
  - **Normal** - Normal operations (e.g. opening / closing connections, etc.)
  - **Debug** - Debugging information used by developers
  - **Verbose** - Very detailed debug information used by developers
  - **All** - Show all trace information

These levels are in decreasing order of severity. Each severity level also includes all the log messages for the levels above it. For example, if 'Warnings' is selected, messages at the 'Severe error' and 'Errors' levels will also be logged.

- **Log size** - Maximum size in MB of one OPC layer log file. A setting of '0' means that the size is not limited.
- **File limit** - the maximum number of log files from the OPC layer source. If exceeded, the oldest file is deleted. A setting of '0' allows any number of files to be created.

### Runtime settings

- **Severity** - the severity of runtime messages to log. The possible settings are:
  - **None** - Do not log any messages
  - **Error** - Errors that could affect program operation
  - **All** - Show all trace information

These levels are in decreasing order of severity. Each severity level also includes all the log messages for the levels above it. For example, if 'Warnings' is selected, messages at the 'Severe error' and 'Errors' levels will also be logged.

- **Log size** - Maximum size in MB of one runtime log file. A setting of '0' means that the size is not limited.
- **File limit** - the maximum number of log files from the runtime source. If exceeded, the oldest file is deleted. A setting of '0' allows any number of files to be created.

### Wire settings

- **Logging** - either 'On' to enable wire logging, or 'Off' if no wire messages should be logged.
- **Log size** - Maximum size in MB of one wire log file. A setting of '0' means that the size is not limited.
- **File limit** - the maximum number of log files from the wire source. If exceeded, the oldest file is deleted. A setting of '0' allows any number of files to be created.

### Configuration settings

- **Logging** - either 'On' to enable configuration change logging, or 'Off' if no configuration change messages should be logged.
- **Log size** - Maximum size in MB of one configuration changes log file. A setting of '0' means that the size is not limited.
- **File limit** - the maximum number of log files from the configuration changes source. If exceeded, the oldest file is deleted. A setting of '0' allows any number of files to be created.

### General settings

- **Logs directory** - the directory where all the log files will be written. Use the ... button to browse for a directory. The exact file name used for a log will depend on this directory setting and the log type.
- **File name includes date** - if this check box is selected, the log file name will end with the date in the format 'yyyymmdd'.

All log files will be written in XML format. A default XSL style sheet is provided to allow the log files to be viewed directly in Internet Explorer.

### 2.4.7.2 Compact/Repair MS Access Database

You can compact Microsoft Access databases, which can be either configuration databases or historical databases, using the **Compact/Repair MS Access Database** dialog box, shown in the figure below. To open this dialog box, select **Compact/Repair MS Access Database** from the **Tools** menu. Microsoft Access-based databases are subject to database fragmentation over time, and the support for the database will compact the target database, reclaim unused space, and drastically improve database performance.

**Note:** It is critical that no users or client applications are connected to the database at the time of compacting and that, if the **Backup Original Database** option is selected, there is plenty of available hard disk space. Particularly, it is not possible to compact a database if it is active at the same time. For this reason, the "Connected Configuration Database" option is grayed out when the database is active

By default, the currently **Connected Configuration Database** is selected. If you want to compact/repair a Microsoft Access database other than the currently connected database, select the **Other Database** option. Then click the ... button and browse for a .mdb file. If the database you select is protected by Microsoft Access security, you will need to enter the **User Name** and **Password** required to gain access to the database. Unchecking the **Use Blank Password** check box makes the **Password** field available for editing.



**Compact/Repair MS Access Database Dialog Box**

## 2.4.8 Help Menu

The **Help** menu commands are listed in the table below.

### Help Menu Commands

Command	Shortcut Keys	Function
Help Topics	F1	Launches the online Help for the Configurator.
About Application		Launches the About Box, which contains information about the product version number and copyright.

### 3 Configuring the Address Space

The MX I/O Server is the I/O driver core. The I/O Server contains objects and interfaces that:

- Maintain the I/O driver configuration.
- Read and write process hardware data.

The MX OPC Configurator is a client of the MX OPC Runtime that accesses the I/O Server and lets you view and modify properties for devices and data tags. The **Address Space** tree control in the Configurator sets the properties and connection parameters for the following hierarchy of server objects:

- **Devices:** A device is a hardware device or station that uses the I/O driver to communicate with a client PC. The device item contains the properties and methods that govern the behavior of a device. A device is visible to the OPC client.
- **Data tags:** A data tag is an object that makes device data accessible to OPC clients. Data tags can be logically organized into groups (folders).

#### 3.1 Tree View

The configuration is shown as a tree view and list view, similar to Windows Explorer. Selecting an item in the tree view on the left shows the associated items in the list view on the right.



**Address Space Tree Control**

#### 3.2 List View

The **List View** shows elements (devices, groups, tags, etc.) in either the left or right navigation pane based on name of the element – rather than a large icon, small icon or full detail view.

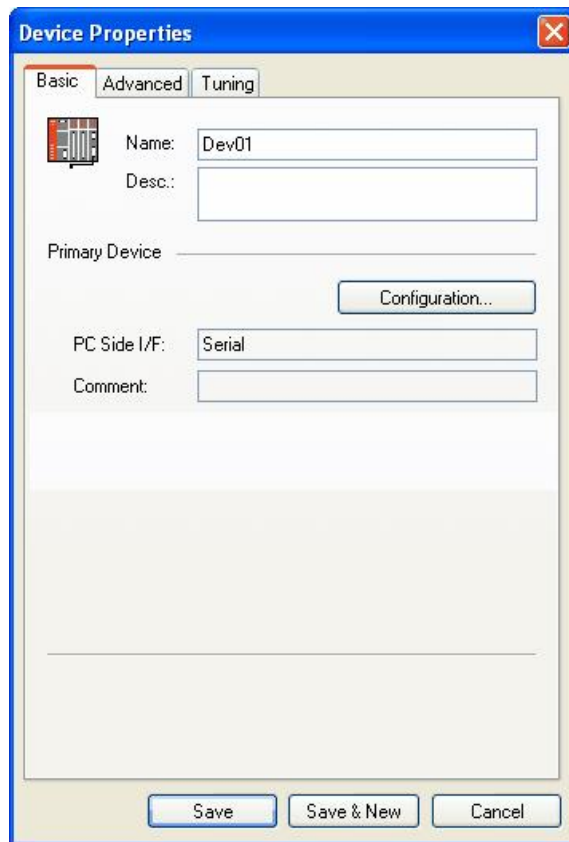
#### 3.3 Devices

Every **device** is connected to a particular port. The device is represented by its symbolic name, and it is uniquely identified by the address value. It is impossible to have two devices with the same address connected to one port.

From version 5.0 onwards, MX OPC Server supports Modbus Serial and Modbus Ethernet devices. To support the configuration of these devices, the address space is enhanced so the user can create Modbus devices in addition to the MX devices known from previous versions.

### 3.3.1 Basic Device Properties

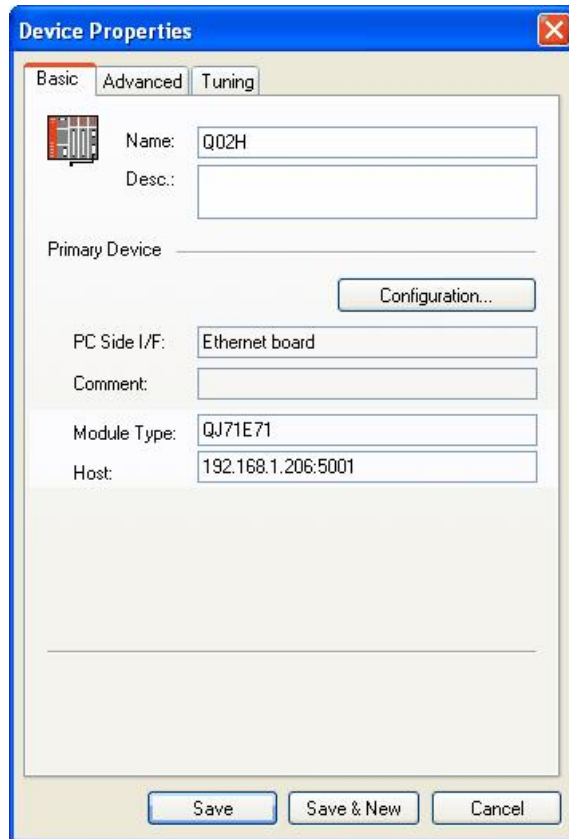
In the **Basic tab** of the **Device Properties** dialog box, shown below, configure the following settings:



**Device Properties: Basic Tab**

When the MX Device's PC Side I/F is configured as **Ethernet board**, the **Basic** page of the **Device Properties** sheet shows additional information: **Module Type** and **Host** IP address with port number, as shown in the screenshot below.

These two controls will appear only when I/F is configured as **Ethernet board**.



**Device Properties: Basic Tab (Ethernet)**

Configure the following basic properties for the device:

- **Name:** Specifies the name of the selected device. Any application requesting data from the I/O driver uses this name to access points on the device. Each device that the driver communicates with should have a unique device name regardless of the device's channel. The name can be up to 32 alphanumeric characters, including underscores ( \_ ) and hyphens ( - ). It cannot begin with a digit.
- **Description:** Allows you to enter text about the selected device. Entries in this field can be very helpful when you go back to look at old configuration or report files, or when you need to modify an existing configuration. The more detailed and specific the information you enter in this field, the easier it is to identify the device. The description can be up to 50 alphanumeric characters and symbols.
- **PC Side I/F:** Specifies which connection type and which physical connection (COM) port to use for the selected device. The connection parameters (listed in the tables below) are selected in the **Communication Setting Wizard**. Click the **Configuration** button to launch the Communication Setting Wizard.

*Note:* If the transfer setup for this item has been changed since the device was last edited, you will see the warning message below when you try to configure the device. Select either:

- **Yes** - to update the stored transfer setup with this name using the properties from this device, or
- **No** - to continue the wizard with the properties in the stored transfer setup. The current device properties will be overwritten when the changes are saved.



- **Comment:** Displays any comments entered in the Communication Setting Wizard.
- **Module Type:** Shows the connected module, for Ethernet boards only.
- **Host:** Shows the IP address and port number, for Ethernet boards only.

<b>Communication Path</b>	<b>Description</b>
Serial	Communication over a serial link can be either direct CPU or to a serial communications card, e.g. AJ71C24, QJ71C24, LJ71C24.
MELSECNET/10	Communication over MELSECNET/10 is via a special function module.
MELSECNET/H	Communication over MELSECNET/H is via a special function module.
CC-Link board	Communication over CC-Link network.
CC-Link IE board	Communication over CC-Link IE network.
CC-Link IE Field board	Communication over CC-Link IE field wiring.
CC-Link IE Field ethernet adapter	Communication over CC-Link IE field wiring via ethernet adapter.
Ethernet board	Communications via Ethernet using TCP/IP or UDP/IP protocols.
CPU board	Communications via a slot PLC.
GOT transparent	Communication indirectly via a GOT (serial, USB or Ethernet).
Q Series Bus	Communication over Q Series bus is via a back plane.
GX Simulator	Communication over GX Simulator is via GX Developer.
GX Simulator2	Communication over GX Simulator2 uses GX Works2.
MT Simulator2	Communication over MT Simulator2 uses MT Works2.
Modem	Communication over MODEM is via special function module.
USB	Communication over a serial link can be direct to CPU USB port.
SX Controller	Not supported.
COM1 to COM2 (if available)	Computer Ports.
COM3 to COM256	Expansion Ports.



**IMPORTANT:** Verify that the COM port you select is not used by another channel or application. Assigning two channels to the same port results in a communication failure on one of the channels. Assigning a channel to a COM port used by another application (such as an alarm printer) can produce problems with driver communication.

- **Save:** Saves all changes specified in the properties dialog box. The device appears under the **Address Space** tree control of the configurator.
- **Save & New:** Saves all changes specified in the properties dialog box and immediately starts configuration of a new device.
- **Cancel:** Closes the properties dialog box.

### 3.3.2 Advanced Device Properties

In the **Advanced** tab of the **Device Properties** dialog box, shown below, configure the following settings:

The screenshot shows the 'Device Properties' dialog box with the 'Advanced' tab selected. The dialog is organized into two main sections: 'Primary Device' and 'Backup Device'. Each section contains a 'Enable Device' checkbox (checked), a 'Simulate' checkbox (unchecked), and a 'Password' field. The 'Primary Device' section also includes a 'Configuration...' button. Both sections feature a set of configuration parameters: 'Reply timeout' (1000 ms), 'Delay time' (0 ms), 'Number of retries' (3), 'Merge gap' (64 words), and 'Max count for R/W operation' (16 words). The 'PC Side I/F' field in the 'Backup Device' section is set to 'Modem'. At the bottom of the dialog are three buttons: 'Save', 'Save & New', and 'Cancel'.

Device Properties: Advanced Tab

## Primary Device

- **Enable Device:** Enables the driver to poll the channel when the check box is checked. If you clear the check box, the driver does not poll the selected device. You may want to disable one or more devices in the following situations:
  - You are swapping hardware for repair or maintenance and do not want to display errors.
  - You do not need to collect data from all your devices and you want to reduce the communications load.
  - You want to isolate a device for debugging.
- **Simulate:** Simulates polling of the channel by the driver when the check box is checked.
- **Password:** The previous screenshot shows the **Advanced** page of the MX Device **Device Properties** sheet configured for a serial PC Side I/F. When the device is configured as a Modem device, the user can also specify a password on the Advanced page.
- **Reply timeout** - the length of time in milliseconds that the driver will wait for a response from the process hardware after sending a request. The retry handling is described in more detail in *Setting Device Timing Properties*, below.
- **Number of retries** - Specifies how many times the driver resends a failed message to the device before marking the data as failed.
- **Delay time** - after the communication has failed as many times as specified by **Number of retries**, the driver will wait for this length of time in milliseconds before trying again.
- **Merge gap** - The maximum gap where data blocks can merge together into one read/write operation. For example if this value is 1, it means that a block made of consecutive addresses D20-D50 can be joined with a block D52-D80.
- **Max count for R/W operation** - this is the maximum number of individual register read/writes that can be grouped together into one batch read/write operation. This setting applies to individual registers that are read separately, not to blocks of registers (e.g. D20-D80 as described in **Merge gap** above). For example, if the device needed to read items D1, D500 and D1000 and this value was set to 2, D1 and D500 would be grouped together into one batch read operation, and D1000 would be included in the next batch operation. If the value was set to 3, all three registers would be read as part of the same batch operation. If your system is working normally, you do not need to change this value. It is not recommended to increase this value above the default.

## Backup Device

- **Enable Device:** Enables the backup device.
- **Comment:** Displays any comments entered in the Communication Setting Wizard.
- **Password:** When the backup device is configured as a Modem device, a password can be specified independently from the primary device.
- **Reply timeout** - the length of time in milliseconds that the driver will wait for a response from the process hardware after sending a request. The retry handling is described in more detail in *Setting Device Timing Properties*, below.
- **Number of retries** - Specifies how many times the driver resends a failed message to the device before marking the data as failed.
- **Delay time** - after the communication has failed as many times as specified by **Number of retries**, the driver will wait for this length of time in milliseconds before trying again.
- **Merge gap** - The maximum gap where data blocks can merge together into one read/write operation. For example if this value is 1, it means that a block made of consecutive addresses D20-D50 can be joined with a block D52-D80.
- **Max count for R/W operation** - this is the maximum number of individual register read/writes that can be grouped together into one batch read/write operation. This setting applies to individual registers that are read separately, not to blocks of registers (e.g. D20-D80 as described in **Merge gap** above). For example, if the device needed to read items D1, D500 and D1000 and this value was set to 2, D1 and D500 would be grouped together into one batch read operation, and D1000 would be included in the next batch operation. If the value was set to 3, all three registers would be read as part of the same batch operation. If your system is working normally, you do not need to change this value. It is not recommended to increase this value above the default.

- **PC Side I/F:** Specifies which connection type and which physical connection (COM) port to use for the selected device. The connection parameters (listed in the tables below) are selected in the **Communication Setting Wizard**. Click the **Configuration** button to launch the Communication Setting Wizard.

Communication Path	Description
Serial	Communication over a serial link can be either direct CPU or to a serial communications card, e.g. AJ71C24, QJ71C24, LJ71C24.
MELSECNET/10	Communication over MELSECNET/10 is via a special function module.
MELSECNET/H	Communication over MELSECNET/H is via a special function module.
CC-Link board	Communication over CC-Link network.
CC-Link IE board	Communication over CC-Link IE network.
CC-Link IE Field board	Communication over CC-Link IE field wiring.
CC-Link IE Field ethernet adapter	Communication over CC-Link IE field wiring via ethernet adapter.
Ethernet board	Communications via Ethernet using TCP/IP or UDP/IP protocols.
CPU board	Communications via a slot PLC.
GOT Transparent	Communication indirectly via a GOT (serial, USB or Ethernet).
Q Series Bus	Communication over Q Series bus is via a back plane.
GX Simulator	Communication over GX Simulator is via GX Developer.
GX Simulator2	Communication over GX Simulator2 uses GX Works2.
MT Simulator2	Communication over MT Simulator2 uses MT Works2.
Modem	Communication over MODEM is via a special function module.
USB	Communication over a serial link can be direct to the CPU USB port.
SX Controller	Not supported.
COM1 to COM2 (if available)	Computer Ports.
COM3 to COM256	Expansion Ports.

**IMPORTANT:** Verify that the COM port you select is not used by another channel or application. Assigning two channels to the same port results in a communication failure on one of the channels. Assigning a channel to a COM port used by another application (such as an alarm printer) can produce problems with driver communication.

## Setting Device Timing Properties

**Reply Timeout**, **Number of Retries**, and **Delay Time** are the timing properties of the driver and are set for each device.

The timing sequence is as follows:

1. The I/O driver sends a message to the process hardware and waits the length of time specified in the **Reply Timeout** field for the device to respond.
2. If the device does not respond, the driver resends the message for the number of times specified in the **Number of Retries** field.
3. The driver marks the data as failed after all retries have been sent and the device has not responded.
  - a. If a back-up device is specified, the driver immediately switches devices, waits the length of time specified in the **Delay Time** field, and sends the message to read the same data from the back-up device. The timeout and reply process repeats and, if the back-up device fails, the driver switches back to the primary device and starts the messaging cycle again.
  - b. If a back-up device is not specified, the driver waits for the specified **Delay Time** and re-initiates the polling process with the selected device.
4. The device may have many items of data to read. In this situation, the driver uses its Quick Fail logic and only performs one cycle through the Timeout, Retries, and Delay process for the messages currently in the Write queue. It then marks all the data on that device as failed and moves on to the next device. The next time the driver attempts to send the message to the failed device, it ignores the Retries and only makes one attempt. If the attempt is successful, the driver recovers all data on the device and marks it as ready for messages.
5. By using Quick Fail, the driver saves time and bypasses the problem device, thereby increasing its efficiency.

### Example

Reply Timeout = 5000 milliseconds (5 seconds)

Retries = 5

Delay Time = 300,000 milliseconds (5 minutes)

Backup Device = none

The driver attempts to send a message to the process hardware. After 5 seconds, the device still has not responded, so the driver resends the message.

The driver tries to send the message 6 times (the first time and then the 5 retries) with 5-second intervals between each attempt.

Each attempt fails; consequently, the driver marks all the data as failed. If the driver has messages for other data items on the same device, it adds the messages to a queue for the failed device and sends them only once without retries.

The driver waits 5 minutes and starts the message send cycle again unless there is other data defined for the device. If there is other data, the driver fails all data items and goes on to the next device.

**Reply Timeout:** Specifies how long (in milliseconds) the I/O driver waits for a response from the selected device.

**Example**

To specify a timeout of...	Enter...
1 second	1000
1 minute	60000
1 hour and 30 minutes	324000000

**Number of Retries:** Specifies how many times the driver resends a failed message to the device before marking the data as failed and initiating the **Delay Time**.

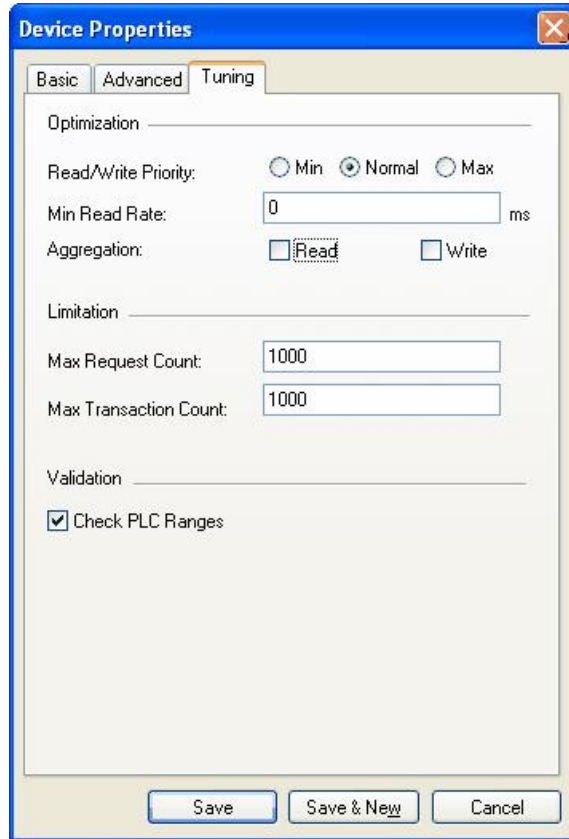
**Delay Time:** Specifies how long (in milliseconds) the I/O driver waits after all retries, specified in the **Retries** field, have failed.

**Example**

To specify a delay of...	Enter...
1 second	1000
1 minute	60000
1 hour and 30 minutes	324000000

### 3.3.3 Device tuning

The **Tuning** tab of the **Device Properties** dialog box, shown below, can be used to optimize the way that data is read from the device, depending on your application.



**Device Properties: Tuning Tab**

#### Optimization

The available optimization settings for the device are:

- **Read/write priority** - the priority of read/write operations compared to the update (polling) operations. The options are:
  - **Min** - updates have the highest priority
  - **Max** - Reads/writes are preferred
  - **Normal** - there is a balance between read/write and update operations
- **Min read rate** - the shortest time between two PLC reads of the same item. If another request to read the same item is received before this time expires, the read will not take place and instead the last value is returned. After the time expires, the value is read again.
- **Aggregation** - this setting controls how the runtime deals with read or write requests that are received when there is already a read/write request for the same data in the queue. If aggregation is switched on, the requests are combined. This will usually improve efficiency, but is only acceptable if the device only needs to see the last value that was written. If the device relies on receiving all intermediate values for an item (for example, when the device is counting the on/off transitions for a boolean value), do not use write aggregation.



**Limitation**

The limitation settings are:

- **Max request count** - Maximum permitted size of the PLC request queue
- **Max transaction count** - Maximum permitted number of outstanding transactions

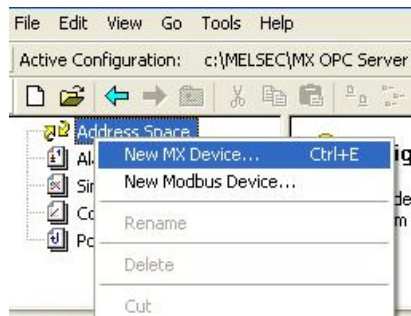
**Validation**

If the 'Check PLC ranges' option is selected, the runtime will test addresses against the device to see if they are valid with its current configuration. If this option is switched off, addresses will only be checked against a local database of valid addresses, which may include addresses that are not valid on the device as it is currently configured.

### 3.3.4 Adding a New MX Device

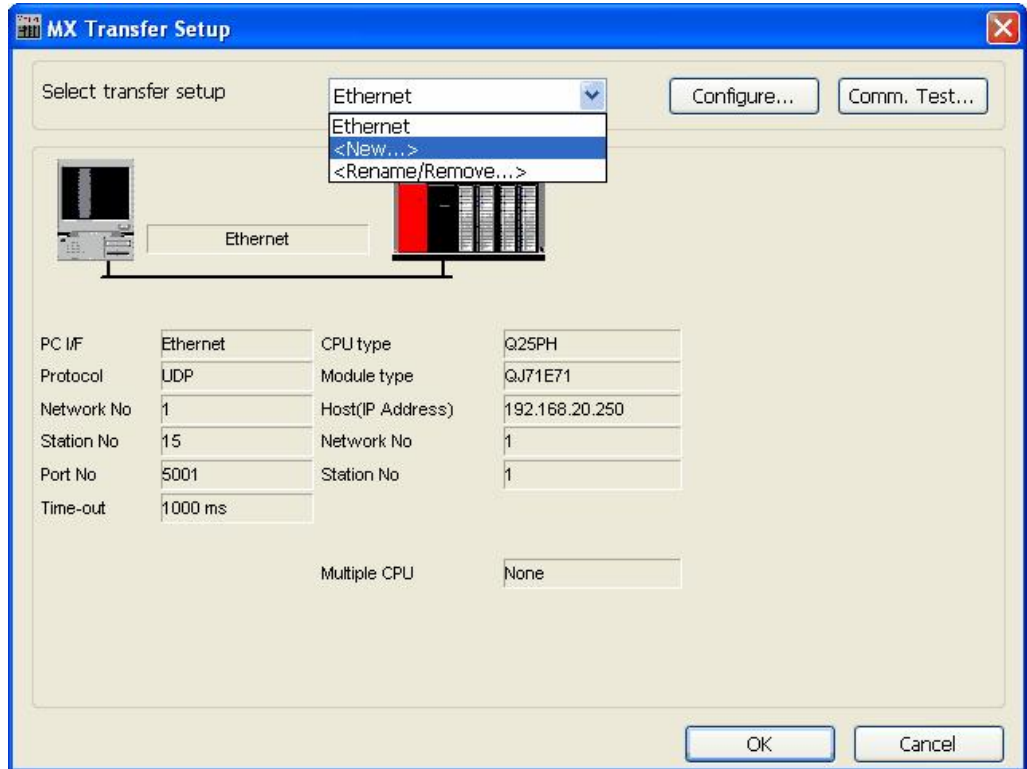
To add a new device:

1. Right-click the **Address Space** tree control of the Configurator screen and select **New MX Device** from the pop-up menu, as shown in the figure below.



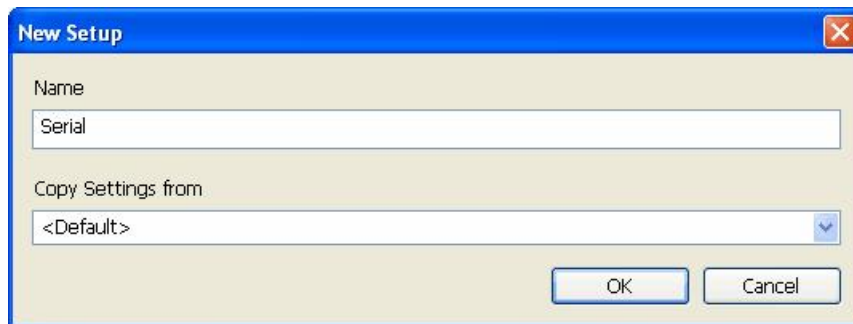
**Adding a New MX Device**

- The **MX Transfer Setup dialog**<sup>[91]</sup> will be shown. You can either select an existing transfer setup, or (as in the example below) select <New...> to create a new transfer setup.



**MX Transfer Setup dialog**

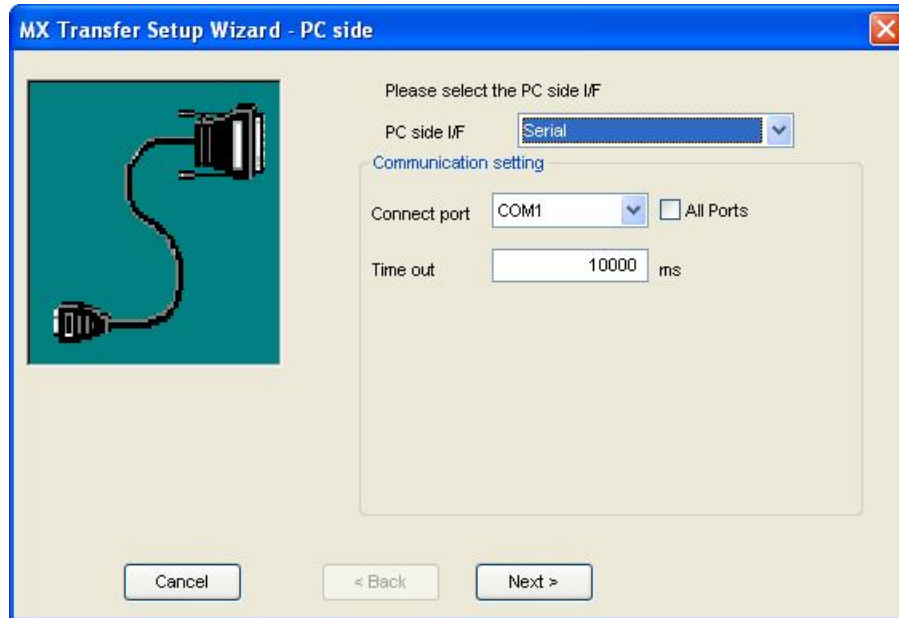
- The **New Transfer Setup dialog**<sup>[93]</sup> will appear. Enter a name for the new transfer setup (*Serial* in the picture below) and select <Default...> as the template, then click **OK**.



**Entering a transfer setup name**

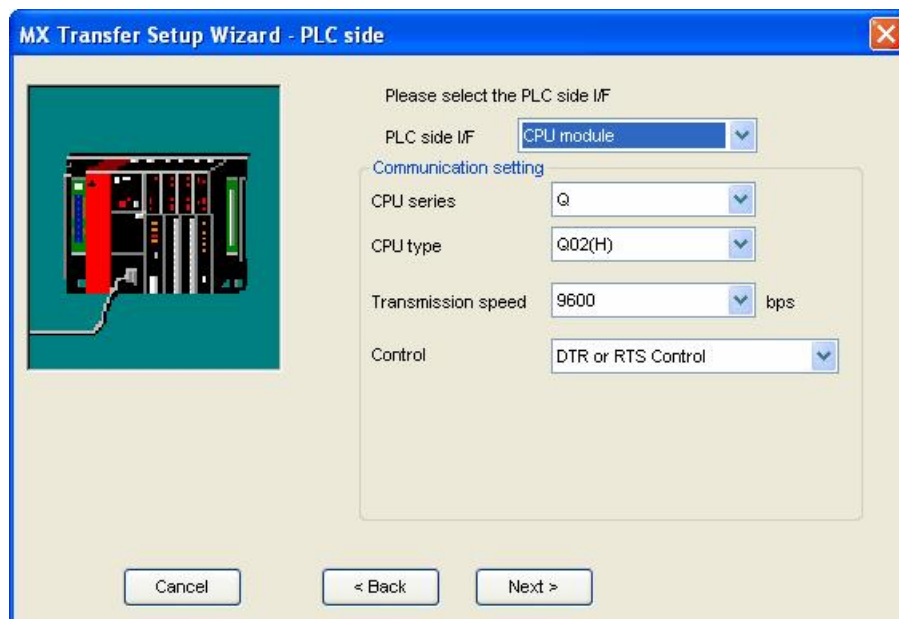
- The new transfer setup can now be edited using the **Communication Setting Wizard**, as shown in the figure below. Specify the communication type to use on the PC side, as shown in the figure below. In the **PC Side I/F** field, select a connection type to use for the selected channel from the drop-down list. Click **Next** to continue.

**Note:** For more information, see the "[Communication Setting Wizard](#)"<sup>[95]</sup> section.



**Setting up the Connection for the PC Side**

- Now you must specify the communication type to use on the PLC (device) side, as shown in the figure below. In the **PLC Side I/F** field, select a communication type from the drop-down list as the connection type to use for the selected channel. Click **Next** to continue.



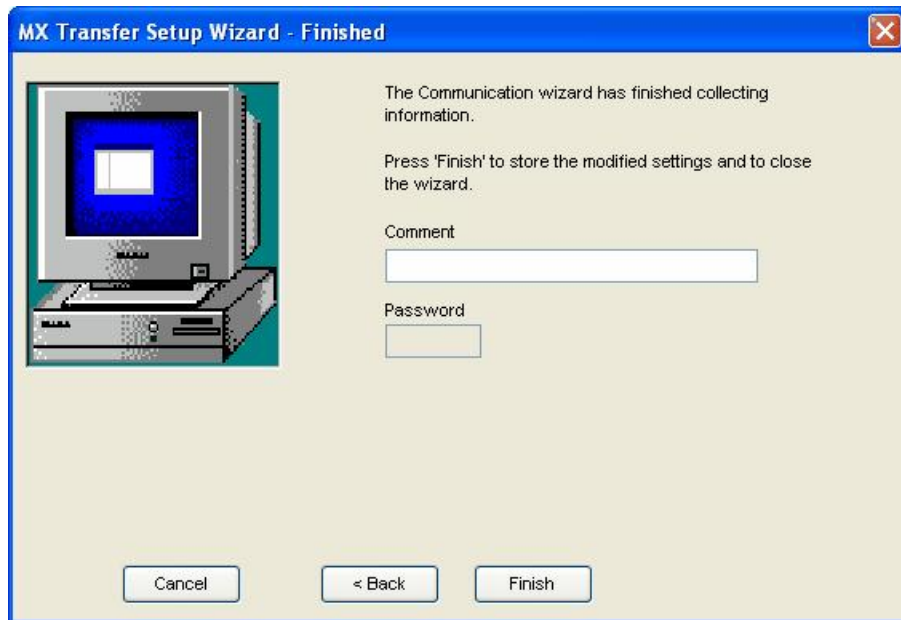
**Setting up the Connection for the PLC Side**

6. Select the network **Station Type** and **CPU Type** (if applicable) from the respective drop-down lists, as shown in the figure below. Click **Next** to continue.



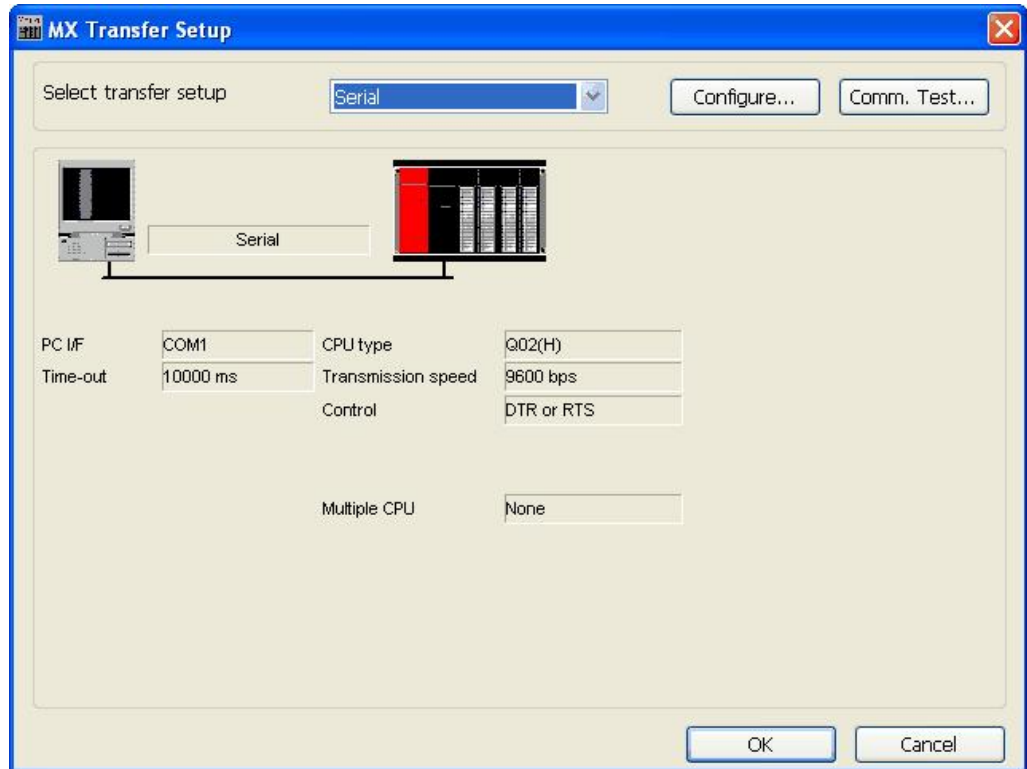
**Configuring Network Settings**

7. Click the **Finish** button to complete the communication channel setup, as shown in the figure below.



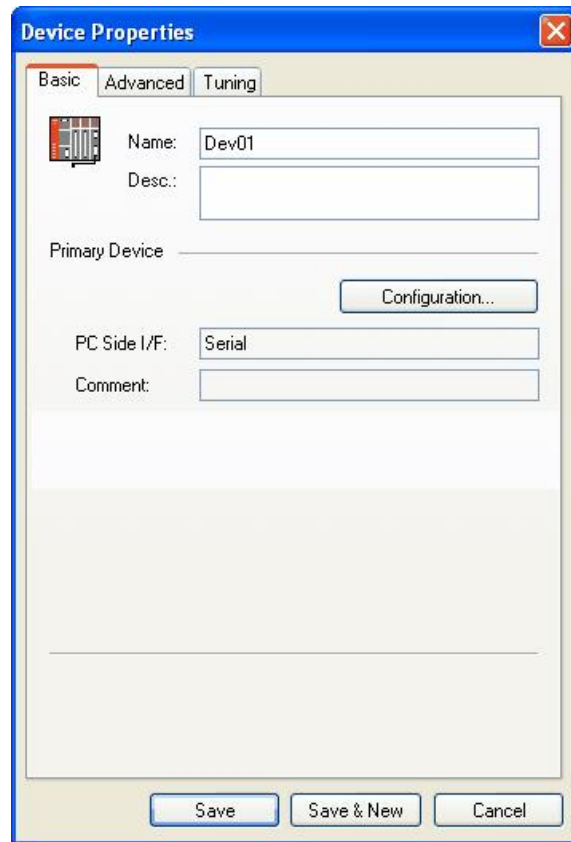
**Completing Channel Setup**

8. The **MX Transfer Setup** dialog box appears with the new transfer setup selected, as shown in the figure below. Each network node is shown graphically, with the node properties shown under the graphic. The left-hand side of the dialog lists the configuration properties for the PC side. You can double-click the image of a node (or double-click the properties list) to edit the configuration properties. The right-hand side of the dialog lists the configuration properties for the PLC device side. You can double-click the image of the PLC device (or double-click the configuration properties list) to edit the PLC device side configuration properties. You can also click the **Configure** button to return to the [Communication Setting Wizard](#)<sup>[95]</sup>. Click the **OK** button to save the changes.



**MX Transfer Setup dialog (configured)**

9. The **Basic** tab of the **Device Properties** dialog box appears, as shown in the figure below.



**Configuring Device Properties**

10. In the **Name** field, type a name for the new device, and type a description for the device (optional). To specify which connection type and which physical connection (COM) port to use for the selected device, click the **Configuration** button to launch the Communication Setting Wizard for the Primary device.

- Click on the **Advanced** tab to set the advanced parameters for the **Primary** and **Backup** devices, as shown in the figure below.

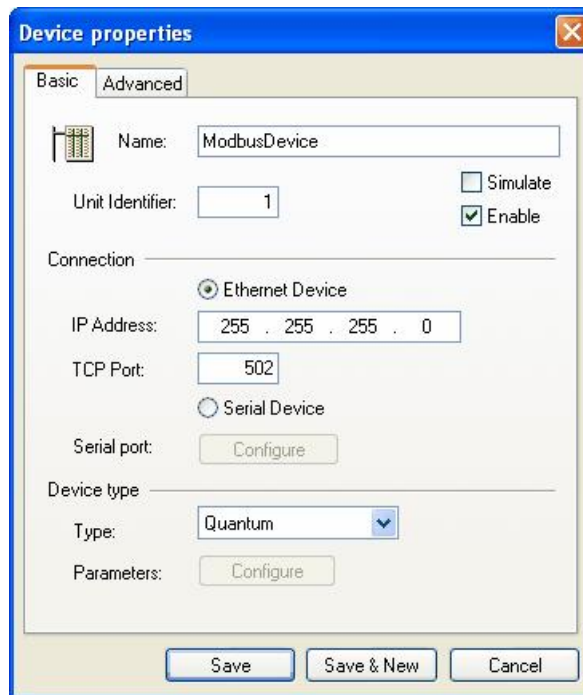
The screenshot shows the 'Device Properties' dialog box with the 'Advanced' tab selected. The dialog is divided into two sections: 'Primary Device' and 'Backup Device'. In the 'Primary Device' section, the 'Enable Device' checkbox is checked, and the 'Simulate' checkbox is unchecked. Below these are input fields for 'Reply timeout' (1000 ms), 'Delay time' (0 ms), 'Number of retries' (3), 'Merge gap' (64 words), and 'Max count for R/W operation' (16 words). The 'Backup Device' section has the 'Enable Device' checkbox unchecked, a 'Configuration...' button, and a 'PC Side I/F' dropdown menu set to '<none>'. There is also a 'Comment' text area. At the bottom of the dialog are three buttons: 'Save', 'Save & New', and 'Cancel'.

**Device Properties: Advanced Tab**

- Check the **Enable Device** check box to activate the device.
- When you have finished configuring the device properties, click the **Save** button. The new device appears under the **Address Space** tree control.

### 3.3.5 Basic Modbus Properties

In the **Basic** tab of the **Device Properties** dialog box, shown below, configure the following settings:



The screenshot shows the "Device properties" dialog box with the "Basic" tab selected. The "Name" field is set to "ModbusDevice". The "Unit Identifier" is set to "1". The "Simulate" checkbox is unchecked, and the "Enable" checkbox is checked. Under the "Connection" section, "Ethernet Device" is selected with a radio button. The "IP Address" is set to "255 . 255 . 255 . 0" and the "TCP Port" is set to "502". The "Serial Device" option is unselected. The "Serial port" field has a "Configure" button next to it. Under the "Device type" section, the "Type" dropdown menu is set to "Quantum", and there is a "Configure" button for "Parameters". At the bottom of the dialog, there are three buttons: "Save", "Save & New", and "Cancel".

**Modbus Device Properties: Basic Tab**



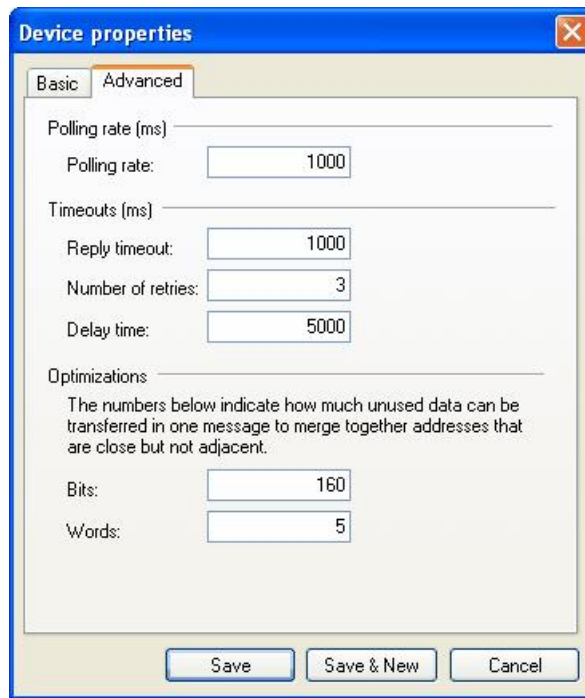
Configure the following basic properties for the device:

- **Name:** Specifies the name of the selected device. Any application requesting data from the I/O driver uses this name to access points on the device. Each device that the driver communicates with should have a unique device name regardless of the device's channel. The name can be up to 32 alphanumeric characters, including underscores ( \_ ) and hyphens ( - ). It cannot begin with a digit.
- **Unit identifier** - enter the Unit identifier which provides a unique ID for this device.
- **Simulate** - Simulates polling of the channel by the driver when the check box is selected.
- **Enable** - Enables the driver to poll the channel when the check box is selected. If you clear the check box, the driver does not poll the selected device. You may want to disable one or more devices in the following situations:
  - You are swapping hardware for repair or maintenance and do not want to display errors.
  - You do not need to collect data from all your devices and you want to reduce the communications load.
  - You want to isolate a device for debugging.
- **Connection** - use the radio button to select which connection type to use for the selected device. The connection parameters (listed below) will depend on whether the device connects using Ethernet or a serial port.
- **IP address** - this is the host address for Ethernet devices (enabled only if the **Ethernet device** radio button has been selected). The default IP address is set at 255.255.255.0.
- **TCP port** - this is the TCP/IP port number for the device (enabled only if the **Ethernet device** radio button has been selected). The default port is set at 502.
- **Serial port** - if the **Serial device** radio button is selected, use the **Configure** button to configure the serial port settings. For a description of the serial port configuration dialog, see [Modbus serial port details](#)<sup>[89]</sup>.

**IMPORTANT:** If you are using a serial connection, verify that the COM port you select is not used by another channel or application. Assigning two channels to the same port results in a communication failure on one of the channels. Assigning a channel to a COM port used by another application (such as an alarm printer) can produce problems with driver communication.
- **Device type** - choose a Device **Type** from a range including "Custom, 184, 384, 484, 584, 584L, 884, 984, Micro 84, Quantum, ModCell, Other (any)". A device with the most limited parameters and the lowest performance would be considered "Other (any)". If you select "Custom" as the device type, use the button "Configure" to configure the parameters of your device (see [Modbus device parameters](#)<sup>[90]</sup> for a detailed description). The default Device type is set at Quantum.
- **Save:** Saves all changes specified in the properties dialog box. The device appears under the **Address Space** tree control of the configurator.
- **Save & New:** Saves all changes specified in the properties dialog box and immediately starts configuration of a new device.
- **Cancel:** Closes the properties dialog box.

### 3.3.6 Advanced Modbus Properties

1. Enter the **Polling Rate** (in milliseconds), as shown in the image below.



**Modbus Device Properties: Advanced Tab**

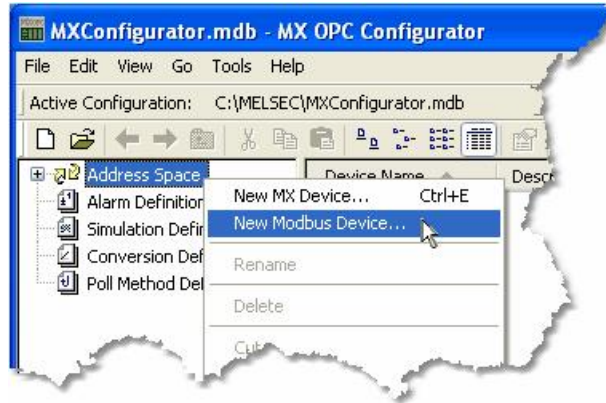
2. Enter the **Reply Timeout**.
3. Enter the **Number of retries** to set the number of consecutive read/write attempts that time out before the OPC server will suspend communication with the device.
4. Enter the **Delay time** to set the amount of time that the OPC server will wait before attempting to reconnect to the suspended device.

When the server tries to optimize communication with devices by requesting as much data in one message as possible, consecutive registers are merged together into one request for efficiency. The server can end up reading registers that are not really requested, when allowed to join two blocks of requested registers. The numbers entered under Optimizations specify the block length of adjacent unused data. Enter **Bits** to set how many unused bits can be in one message to merge together addresses that are close but not adjacent. Enter **Words** to set how many unused words can be in one message to merge together addresses that are close but not adjacent.

### 3.3.7 Adding a New Modbus Device

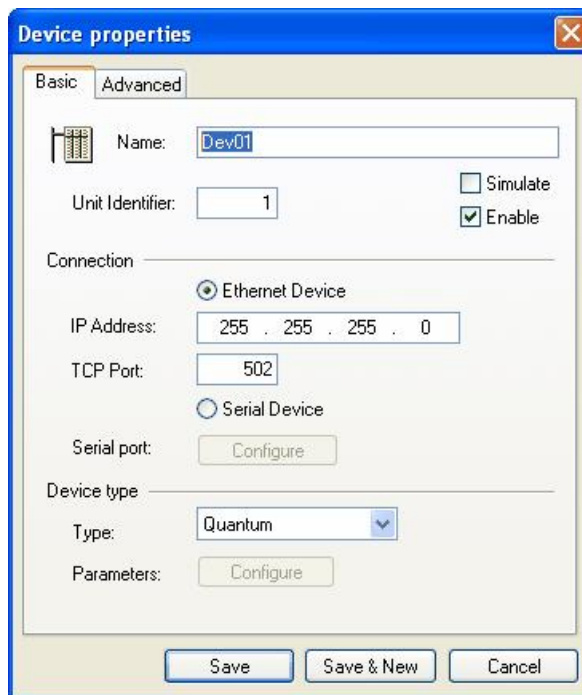
To add a new device:

1. Right-click the **Address Space** tree control of the Configurator screen and select **New Modbus Device** from the pop-up menu, as shown in the figure below.



**Adding a New Modbus Device**

2. This opens the **Device properties** dialog box, as shown in the figure below. In the **Name** field, type a name for the new device.

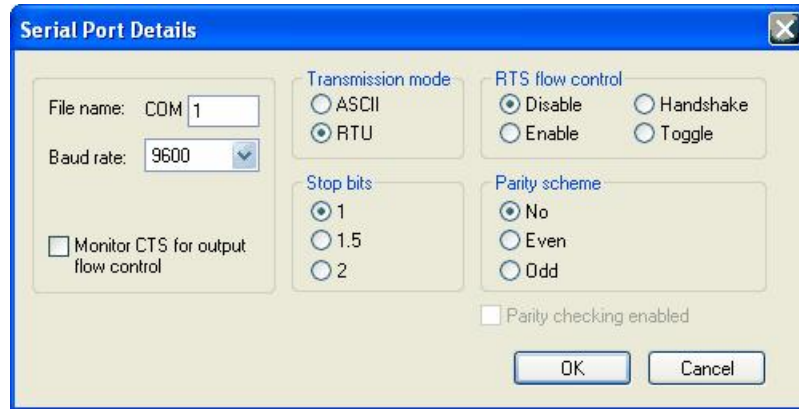


3. Enter the **Unit Identifier** which provides a unique ID for this device.
4. Select the **Simulate** (in order to simulate all Data Items on this device) and/or **Enable** (activate device) check boxes if required.
5. The radio button within the Connection section helps signify if the device is an **Ethernet Device** or a **Serial Device**. Default is "Ethernet Device".
6. Enter the **IP Address** for the Ethernet device. (This section is enabled only if the Ethernet Device radio button has been selected). The default IP address is set at 255.255.255.0.
7. Enter the **TCP Port** for the Ethernet device. (This section is enabled only if the Ethernet Device radio button has been selected). The default TCP Port is set at 502.
8. Configure the **Serial Port** for the Serial device. (This section is enabled only if the Serial Device radio button has been selected). See '[Modbus serial port details](#)'<sup>[89]</sup> for a description of the serial port configuration.
9. Choose a Device **Type** from a range including "Custom, 184, 384, 484, 584, 584L, 884, 984, Micro 84, Quantum, ModCell, Other (any)". A device with the most limited parameters and the lowest performance would be considered "Other (any)". If you select "Custom" as the device type, use the button "Configure" to configure the parameters of your device (see '[Modbus device parameters](#)'<sup>[90]</sup> for a detailed description). The default Device type is set at Quantum.
10. **Parameters:** Opens a dialog for configuring the parameters of this modbus device. This dialog is described in '[Modbus device parameters](#)'<sup>[90]</sup>

### 3.3.8 Modbus serial port details

The **Serial Ports Details** dialog (shown below) allows the user to configure the serial port for the current device. It is shown when the connection type for a Modbus device is set to 'Serial' and the user clicks on the **Configure** button.

When the Parity checking enabled check box is not checked, it will force the server to ignore the parity bits in the message.



**Serial Port Details Dialog**

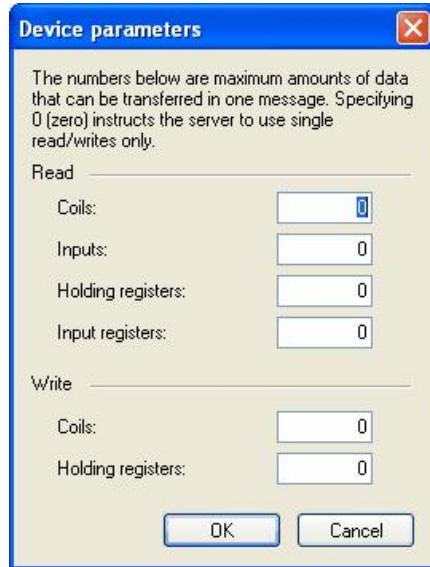
1. Enter the port name in the **File name** entry field.
2. Select the **Baud rate** from the pulldown menu.
3. The **Monitor CTS for output flow control** checkbox allows users to select whether or not the CTS signal for output flow control should be monitored.
4. Select either ASCII or RTU under **Transmission mode**.
5. Under **RTS flow control**, select either Disable, Enable, Handshake or Toggle.
6. Choose the number of **Stop bits** (1, 1.5 or 2)
7. Check which **Parity scheme** is to be used (No, Even, or Odd).
8. The **Parity checking enabled** checkbox allows users to decide whether or not the parity bits should be ignored. This section is enabled when **Parity scheme** is set to "No".

**IMPORTANT:** If you are using a serial connection, verify that the COM port you select is not used by another channel or application. Assigning two channels to the same port results in a communication failure on one of the channels. Assigning a channel to a COM port used by another application (such as an alarm printer) can produce problems with driver communication.

### 3.3.9 Modbus device parameters

The **Modbus Device Parameters** dialog lets the user specify custom device parameters. It can be shown by selecting the device type as 'Custom' in the basic properties page for a Modbus device, then clicking on the **Parameters** button.

The meaning of the numbers in the device parameters dialog, shown below, is the maximum amount of data that can be transferred in one message. Setting the value equal to zero forces the server to use single read/write messages only.



The numbers below are maximum amounts of data that can be transferred in one message. Specifying 0 (zero) instructs the server to use single read/writes only.

Read

Coils:

Inputs:

Holding registers:

Input registers:

Write

Coils:

Holding registers:

OK Cancel

**Modbus Device Parameters Dialog**

#### READ

1. **Coils:** Enter the maximum number of coil bits sent in one message.
2. **Input:** Enter the maximum number of input bits sent in one message.
3. **Holding registers:** Enter the maximum number of holding registers sent in one message.
4. **Input registers:** Enter the maximum number of input registers sent in one message.

#### WRITE

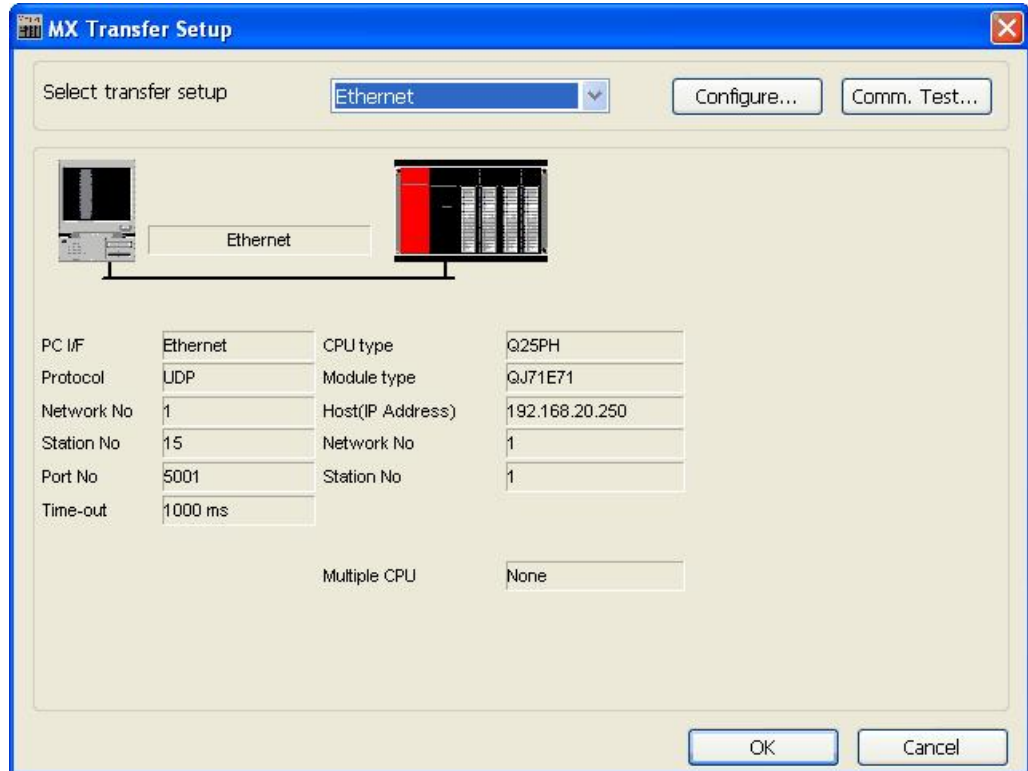
5. **Coils:** Enter the maximum number of coil bits sent in one message for write operations.
6. **Holding registers:** Enter the maximum number of holding registers sent in one message for write operations.

### 3.3.10 Transfer setup management

When new MX devices are added to the MX OPC project, the communication type (e.g. Ethernet, Serial etc.) and details (e.g. serial port number, host address etc.) must be set up so that the runtime module knows how to communicate with the device.

Once you have configured the communication details for a device, it can be useful to recall them instantly without having to enter all of the details again. In MX OPC Configurator, these transfer details are handled using the **MX Transfer setup** dialog.

When you configure the connections for a device, the **MX Transfer setup** dialog will be shown, as in the figure below.



**MX Transfer setup dialog**

The network architecture and settings for the selected transfer setup are displayed with graphics representing each network node. The parameters associated with a network node are shown under the graphic. The PC side connection is shown on the left, and the PLC side connection on the right.

Using this dialog, you can:

- [Add new transfer setups](#)<sup>[92]</sup>
- [Rename existing transfer setups](#)<sup>[93]</sup>
- [Remove existing transfer setups](#)<sup>[93]</sup>
- Modify a transfer setup
- Select a transfer setup to be used for the device
- Test the communication with a selected transfer setup

The controls on this dialog are:

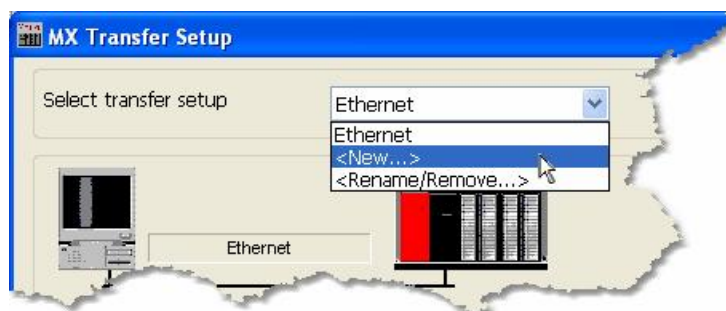
- **Select transfer setup** - this list contains the previously configured transfer setups, along with the entries:
  - **<New...>** - select this to show the New Setup dialog, to create a new transfer setup definition.
  - **<Rename/Remove...>** - select this to rename or delete other entries in the list.
- **Configure** - if an existing transfer setup is selected in the **Select transfer setup** list, the [Communication Setting Wizard](#)<sup>[93]</sup> is shown to modify it. If **<New...>** is selected, the [New Setup dialog](#)<sup>[93]</sup> is displayed.
- **Comm. test** - attempt a connection to the device using the selected transfer setup details, and confirm whether or not the connection was successful.
- **OK** - accept the changes and continue
- **Cancel** - exit to the previous screen

You can also double-click on a network node graphic or on the properties to edit the properties for that node in the [Communication Setting Wizard](#)<sup>[95]</sup>.

The transfer setup details are stored in the local configuration database. If you have moved the configuration database from one computer to another, the connection details for a named transfer setup may be different from those stored locally. If this happens, you may be prompted to overwrite the local settings when you first edit the device on the new computer. If there is not enough space to store the new settings, you will be prompted to select another transfer setup to overwrite.

### 3.3.10.1 Creating new transfer setups

To create a new transfer setup, select **<New...>** from the **Select transfer setup** list, as shown in the picture below:

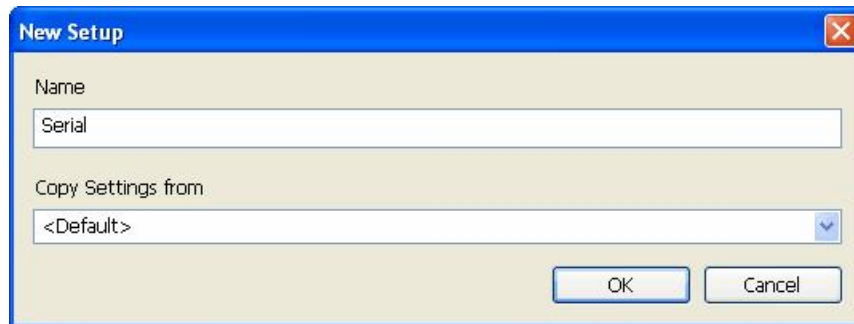


Creating a new transfer setup

**Note:** A maximum of 32 different transfer setups can be stored. If this number is exceeded, an error message will be shown, and you will need to either delete one of the existing transfer setups to make space for the new one, or modify another existing setup instead.



You will be prompted for the name of the new transfer setup, and given the option to copy its settings from another transfer setup. To use default settings, select *<Default>*, as in the example figure below:

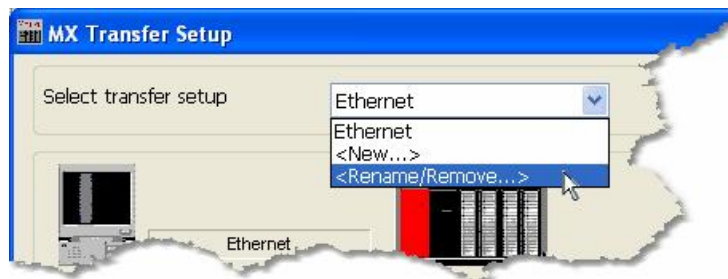


New transfer setup

After you click **OK**, the [Communication Setting Wizard](#)<sup>[95]</sup> will be shown to edit the details of the new connection.

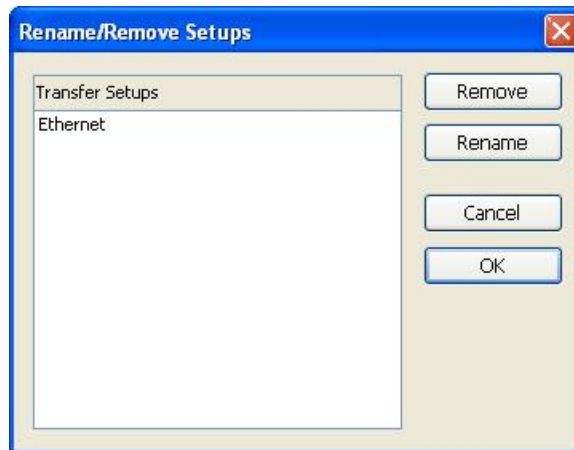
### 3.3.10.2 Renaming / removing transfer setups

To rename or remove an existing transfer setup, select *<Rename/Remove...>* from the **Select transfer setup** list, as shown in the picture below:



Renaming or removing an existing transfer setup

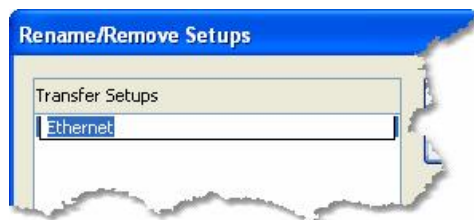
A list of existing transfer setups will appear:



**Rename/remove transfer setups dialog**

The controls on this dialog are:

- **Remove** - Delete the selected transfer setup from the list. You will not be prompted to confirm this, but if you accidentally delete the wrong entry pressing **Cancel** will undo the change.
- **Rename** - Change the name of the selected entry. Modify the name in-place (as in the picture below) then press Enter. You can also rename an item by double-clicking on it in the list.



**Renaming a transfer setup**

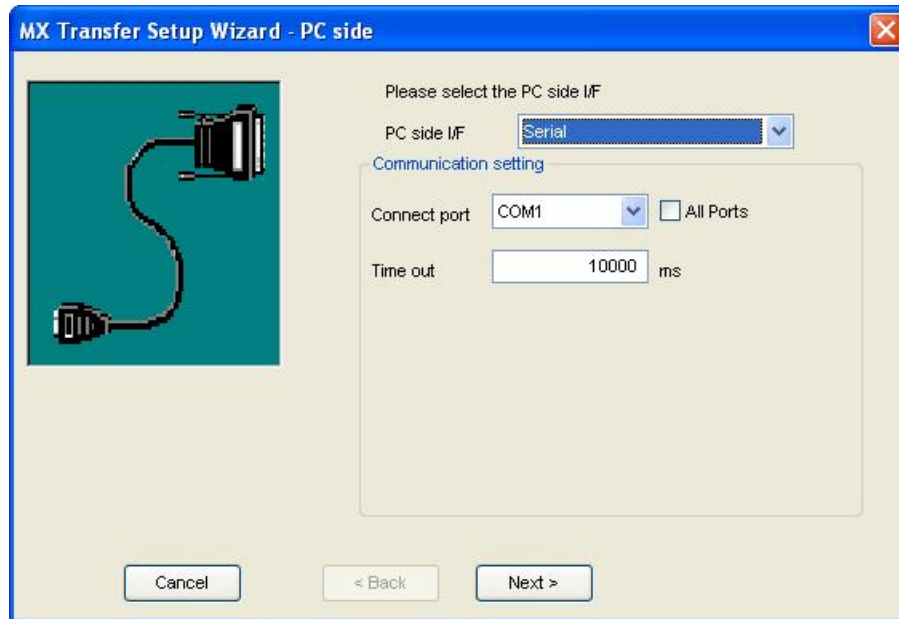
- **Cancel** - exit without saving the changes.
- **OK** - save changes and close the dialog.

### 3.3.11 Communication Setting Wizard

A **channel** is equivalent to a physical port in the computer. To correctly configure a channel means to set up the specific communication parameters for the port like the serial channel speed (Baud rate) and protocol characteristics: Transmission mode (ASCII or RTU), RTS flow control, Stop bits and Parity scheme. When configuring a device, the channel communication properties are configured in the Communication Setting Wizard to set up the communication between the PC and the PLC device.

#### Communication Settings on the PC Side

First you will specify the communication type to use on the PC side, as shown in the figure below.



Setting up the Connection for the PC Side

**PC Side I/F:** Specifies which connection type and which physical connection (COM) port to use for the selected device. The connection parameters (listed in the following tables) are selected in the **Communication Setting Wizard**. Click the **Configuration** button to launch the Communication Setting Wizard.

**Note:** The configuration parameters vary depending on which connection type is selected in the **PC Side I/F** field.

Communication Path	Description
Serial	Communication over a serial link can be either direct CPU or to a serial communications card, e.g. AJ71C24, QJ71C24, LJ71C24.
MELSECNET/10	Communication over MELSECNET/10 is via a special function module.
MELSECNET/H	Communication over MELSECNET/H is via a special function module.
CC-Link board	Communication over CC-Link network.
CC-Link IE board	Communication over CC-Link IE network.
CC-Link IE field board	Communication over CC-Link IE field wiring.
CC-Link IE field ethernet adapter	Communication over CC-Link IE field wiring via ethernet adapter.
Ethernet board	Communications via Ethernet using TCP/IP or UDP/IP protocols.
CPU board	Communications via a slot PLC
GOT transparent	Communication indirectly via a GOT (serial, USB or Ethernet).
Q Series Bus	Communication over Q Series bus is via a back plane.
GX Simulator	Communication over GX Simulator is via GX Developer.
GX Simulator2	Communication over GX Simulator2 uses GX Works2.
MT Simulator2	Communication over MT Simulator2 uses MT Works2.
Modem	Communication over MODEM is via a special function module.
USB	Communication over a serial link can be direct to the CPU USB port.
SX Controller	Not supported.

**Connect Port:** Specifies which physical COM port to use for the selected channel. You can specify as many channels as needed for communication with your hardware.

COM1 to COM2 (if available)	Computer Ports
COM3 to COM256	Expansion Ports

By default, you will only see the physical COM ports that are actually available on your computer. If you are preparing a configuration to be used on another computer which has different COM ports, check the **All ports** box to show a full list of possible serial ports.

**IMPORTANT:** Verify that the COM port you select is not used by another channel or application. Assigning two channels to the same port results in a communication failure on one of the channels. Assigning a channel to a COM port used by another application (such as an alarm printer) can produce problems with driver communication.

**Board Number:** Specifies the module (1-4) of the selected board.

**Connect Module:** Select a unit type from the drop-down list.

**Protocol:** Specifies the primary device's communication protocol as the Transmission Control Protocol (**TCP**) or the User Datagram Protocol (**UDP**). This selection must match the protocol used by your process hardware.

**Packet Type:** Select **Binary** or **ASCII** from the drop-down list.

**Port Number:** This is the port address to be used when connecting to a Melsec Ethernet module. Valid range is 256-65535. The Ethernet module must be configured to accept connections through this port.

**Time out:** Specify a connection time out (in milliseconds).

*Note:* If you are using UDP connections to a PLC, and the OPC server and the PLC are connected through a network with a large volume of traffic, you may experience a high number of connection timeouts. This is due to the nature of the UDP protocol, which drops network 'packets' under high load conditions without informing the sending application that the data was lost.

To solve this problem, either improve the quality of the connection (for example by ensuring that there is enough bandwidth in the network, or that the two systems are connected through a single network switch), or change the connection to use the TCP/IP protocol instead. TCP/IP is designed to guarantee that data sent is eventually received, even when the underlying network link sometimes loses individual 'packets'.

**Network Number:** The Network Number (1-255) only applies to MelsecNet/10, and is the number given to a MelsecNet/10 network using the network module's rotary switches.

**Station Number:** Specifies the station number (0-64 and 255) for the selected device. The station number is the number given to a PLC connected to a second tier network. This is a PLC station number on a Mitsubishi Melsec Network, such as NetB, NetII or Net10.

For A and Q PLCs, the setting range is 0-64 and 255. The range 0-64 can be set on the appropriate Network modules rotary switches.

For MelsecNet/B the setting range is 0-31 where Zero is the master.

For MelsecNet/II the setting range is 0-64 where Zero is the master.

For MelsecNet/10 the setting range is 1-64 (master-less network).

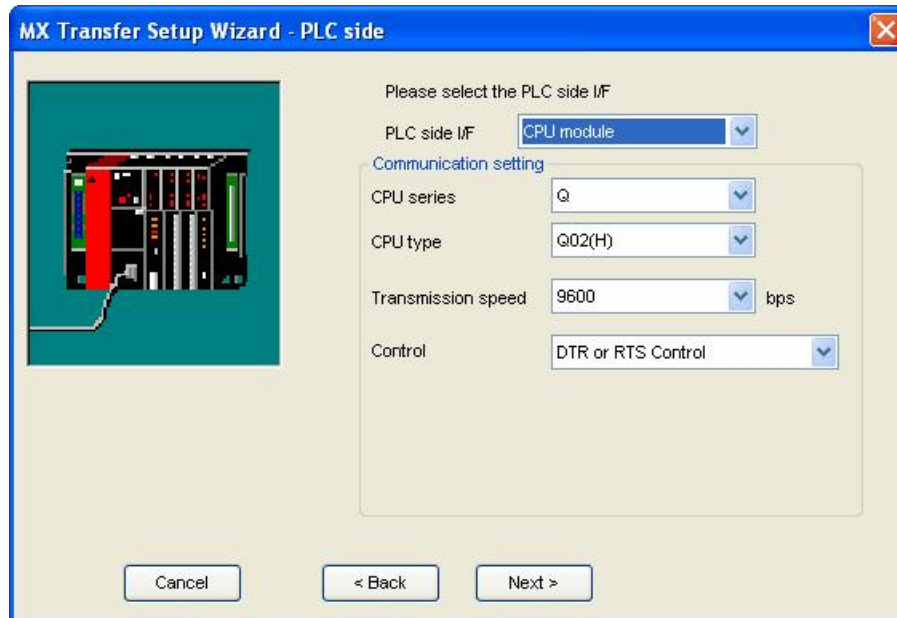
**Notes:**

- i) A setting of 255 (FF Hex) is the default for a PLC when no network is connected.
- ii) Station number has no meaning for FX range of PLC's.
- iii) This setting is often referred to as PC number in other documentation and drivers.

## Communication Settings on the PLC Side

Now you must specify the communication type to use on the PLC (device) side, as shown in the figure below.

**PLC Side I/F:** In the **PLC Side I/F** field, select a communication module from the drop-down list. (The module types listed depend on the communication type you specified for the PC side.)



### Setting up the Connection for the PLC Side

**Note:** The configuration parameters vary depending on which connection type is selected in the **PLC Side I/F** field.

**CPU type:** Specifies the CPU type of the selected device. Select a CPU type from the drop-down list.

Process hardware, such as programmable controllers and interface modules, has specific values for the baud rate, data bits, stop bits, and parity. These properties are the language of the devices – how they communicate with each other. In order for the I/O driver to communicate with these devices, it must speak the same language.

Consequently, when configuring a driver, enter and verify that the driver uses the same baud rate, data bits, stop bits, and parity that your process hardware devices use. If you specify incorrect values, you can never establish communication between driver and hardware. Refer to your hardware vendor for correct values.

**Transmission Speed:** Specifies the baud rate for the selected channel. Baud rate is a characteristic of serial communication. It indicates the number of electrical impulses (data bits) transferred to the communications line per second. For example, transmitting a raw 8-bit code at the rate of 1200 characters-per-second is expressed as 9600 baud.

The baud rate you enter in this field must match the baud rate of the device that the driver is communicating with. The device cannot respond to requests or communicate if you enter an incorrect baud rate. Refer to your hardware documentation for the correct baud rate setting.

The **Transmission Speed** field lets you select from a set of applicable baud rates:

- 300
- 600
- 1200
- 2400
- 4800
- 9600
- 19200
- 38400
- 57600
- 115200

**Control:** Request to Send (RTS) and Data Terminal Ready (DTR) are signals that control the flow of process data for serial communications. Your process hardware may not require these signals, but all devices on a channel must use the same RTS and DTR signal settings. The driver uses the flow control signal as follows:

- **RTS:** Raises the RTS line high when the driver starts. The signal remains high while the driver runs.
- **DTR:** Raises the DTR line high when the driver starts. The signal remains high while the driver runs.

**Station Number:** Specifies the station number (0-64 and 255) for the selected device. The station number is the number given to a PLC connected to a second tier network. This is a PLC station number on a Mitsubishi Melsec Network, such as NetB, NetII or Net10.

For A and Q PLCs, the setting range is 0-64 and 255. The range 0-64 can be set on the appropriate Network module's rotary switches.

For MelsecNet/B the setting range is 0-31 where Zero is the master.

For MelsecNet/II the setting range is 0-64 where Zero is the master.

For MelsecNet/10 the setting range is 1-64 (master-less network).

**Notes:**

- i) A setting of 255 (FF Hex) is the default for a PLC when no network is connected.
- ii) Station number has no meaning for FX range of PLC's.
- iii) This setting is often referred to as PC number in other documentation and drivers.

**Network Number:** The Network Number (1-255) only applies to MelsecNet/10, and is the number given to a MelsecNet/10 network using the network module's rotary switches.

**Data Bits:** Specifies the number of data bits for the selected channel. Data bits are a characteristic of serial communications. The unit of data that is passed along a serial data line consists of data bits, parity, and stop bits. The data bits comprise the actual message being communicated.

The number of data bits used for communication can differ between devices. Devices may use 7-bit, or 8-bit code. The number specified in this field must match the number of data bits used by the device that the driver communicates with or communication cannot be established. Refer to your hardware documentation for the correct data bits value. Valid entries are 7 or 8.

**Stop Bits:** Specifies the number of stop bits for the selected channel. Stop bits are a characteristic of serial communications. The unit of data that is passed along a serial data line consists of data bits, parity, and stop bits. The stop bits indicate the end of a unit of data.

The number of stop bits varies between devices. Some devices require two stop bits while others only require one. The number specified in this field must match the number of stop bits for the device that the channel is communicating with. Refer to your hardware documentation for the correct stop bits setting. A defective stop bit generates a framing error. Valid entries are 1 or 2.

**Parity:** Specifies the selected channel's parity. Parity is a characteristic of serial communications. The unit of data that is passed along a serial data line consists of data bits, parity, and stop bits. Parity is an error-checking method for a unit of data on a serial data line. A parity bit is added to a unit of data to ensure that the unit of data conforms to the particular parity rule.

The type of parity specified for the channel must match the parity of the device it communicates with. Refer to your hardware documentation for the correct parity setting.

None	The parity bit is not used and no parity is specified.
Odd	If odd parity is in effect and an even number of 1s is received, an error is detected.
Even	If even parity is in effect and an odd number of 1s is received, an error is detected.

**Sum Check:** Does the message type require a checksum?

- **None:** The check sum is not used.
- **Existence:** The check sum is used.

**Module Type:** Select a unit type from the drop-down list.

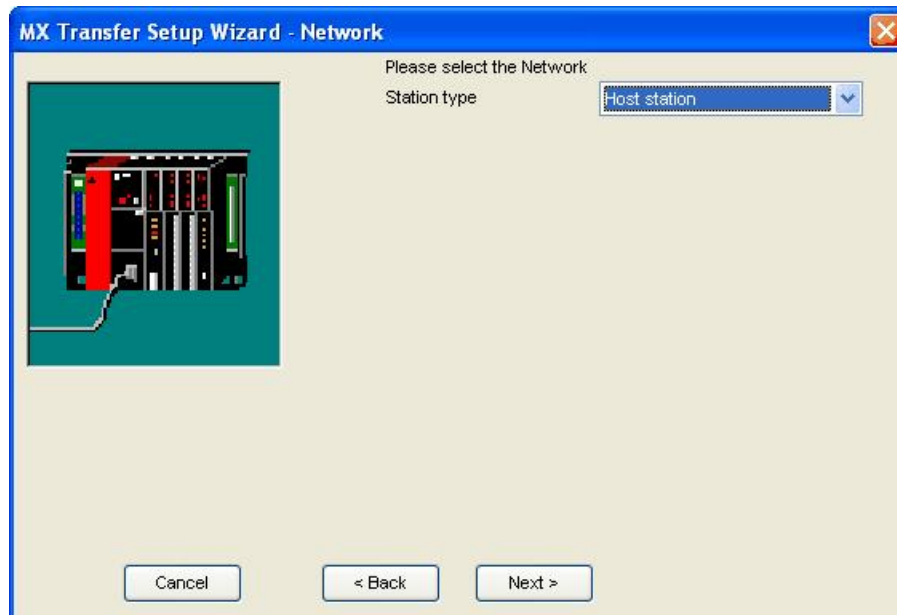
**Port Number:** This is the port address to be used when connecting to a Melsec Ethernet module. Valid range is 256-65535. The Ethernet must be configured to accept connections through this port.

**Transmission wait time:** Specify a connection time out (in milliseconds).

**Mode:** Specify a mode for the selected communication module.

### Specifying the Network Station Type

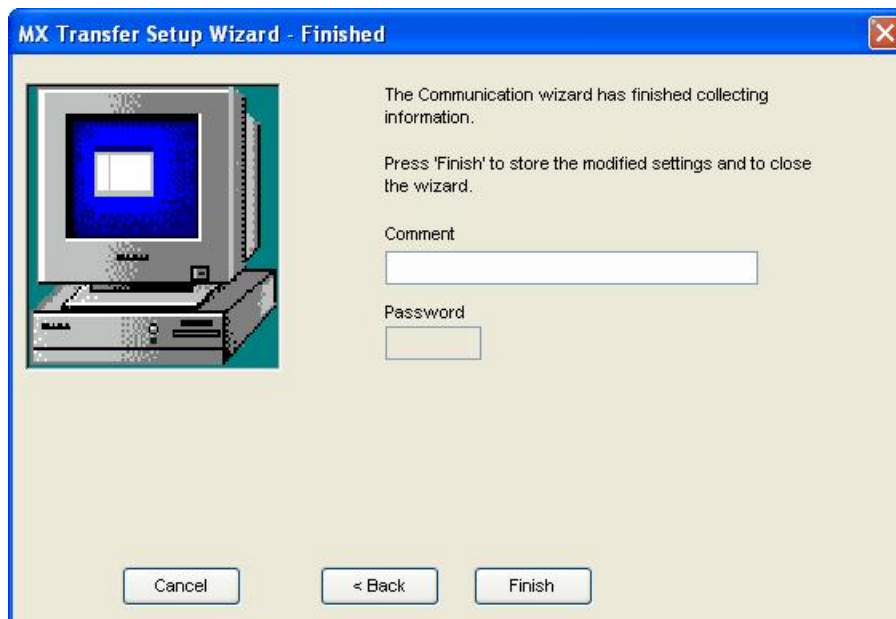
Select the network **Station Type** (host or other) and **CPU Type** (if applicable) from the respective drop-down lists, as shown in the figure below.



Configuring Network Settings



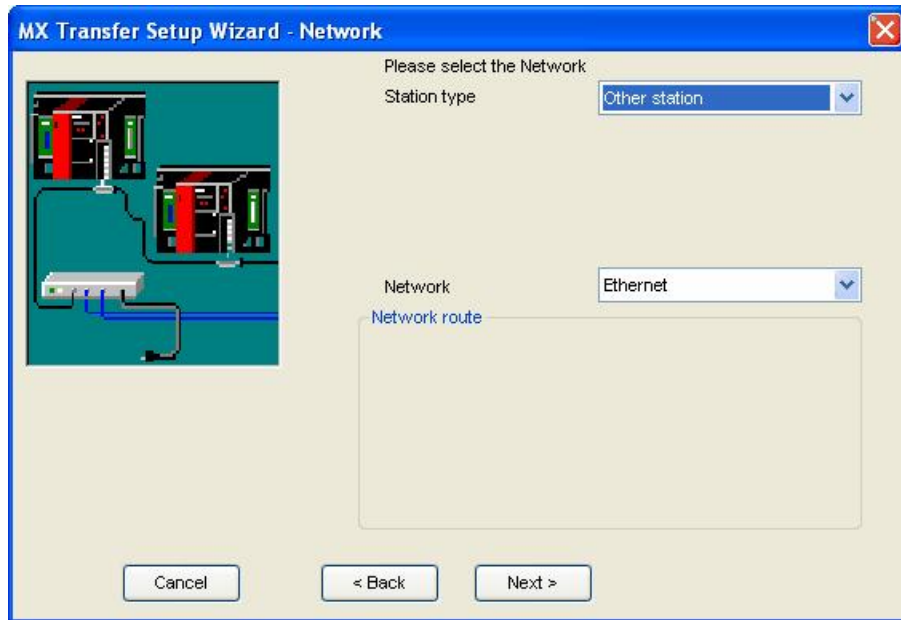
Click the **Finish** button to complete the communication channel setup, as shown in the figure below.



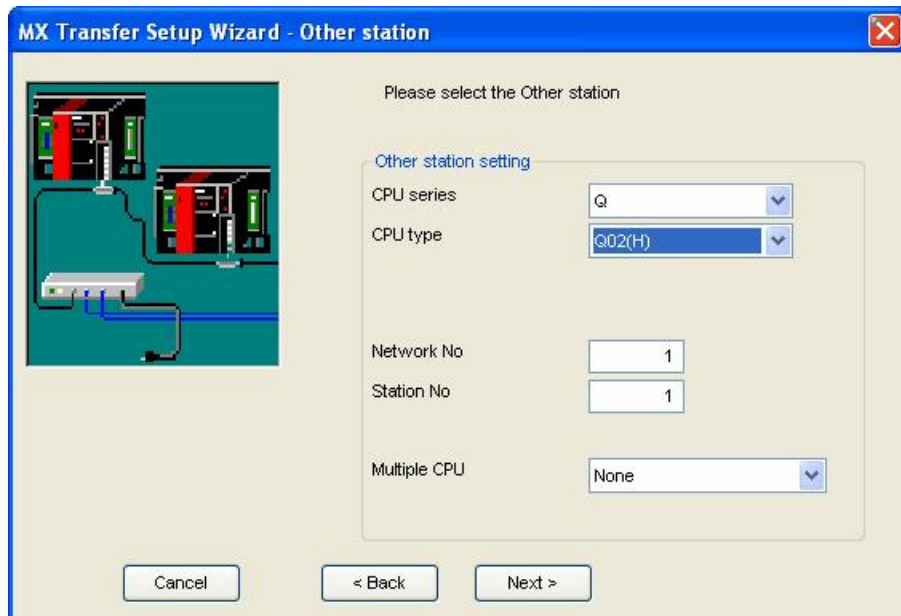
**Completing Channel Setup**

### Properties for Station Type "Other Station"

If you select **Other Station** under network **Station Type** in the Communication Setting Wizard, the **Other Station** properties dialog appears, as shown in the figure below. Select a **Station Type** from the drop-down list, and specify the other parameters (which will depend on the **Station Type**). For some station types there may be another page of parameters. Click the **Next** button when you are finished. Then click the **Finish** button to complete the communication channel setup.



Properties for "Other" Station



Additional properties for "Other" Station type "Ethernet"

### 3.3.12 Modem support

MX OPC Server 6.05 has MODEM configurations:

1. MODEM in Device Configuration Wizard:

- PC Side I/F: **MODEM**
- Connect Module: **Q6TEL, A6TEL, FXCPU, AJ71QC24N, QJ71C24, LJ71C24**

2. MODEM in Device Advanced Setting:

- PC Side I/F: **Serial**
- PLC Side I/F: **CPU module** or **C24 module (AJ71C24, AJ71UC24, AJ71QC24N, QJ71C24, LJ71C24)**
- Advanced setting in device modem: **tone** or **pulse**

PLC	Connection Type	Connection Hardware	Mitsubishi Modem	Non-Mitsubishi Modem	Results	Comments
Q06	MODEM 1	QJ71C24	✓			
Q06	MODEM 1	QJ71C24		✓	✓	Code required in the PLC
Q06	MODEM 2	QJ71C24	✓		✓	Note B
Q06	MODEM 2	QJ71C24		✓		
Q06	MODEM 2	Prog. Port	✓			
Q06	MODEM 2	Prog. Port		✓		
FX2N	MODEM 1	Prog. Port or 232BD	✓		✓	AT%ADP=3 at both ends
FX2N	MODEM 1	Prog. Port or 232BD		✓	✓	
FX2N	MODEM 2	Prog. Port or 232BD	✓		✓	AT%ADP=3 at both ends
FX2N	MODEM 2	Prog. Port or 232BD		✓	✓	
Q2AS	MODEM 1	Q6 TEL	✓		✓	No available settings that will work
Q2AS	MODEM 1	Q6 TEL		✓	✓	DIP switch 1 ON in Q6 TEL.
Q2AS	MODEM 1	QC24	✓		✓	AT%ADP=Q at PLC end & AT%ADP=D at PC end
Q2AS	MODEM 1	QC24		✓		
Q2AS	MODEM 2	HMI to Prog. Port	✓			
Q2AS	MODEM 2	HMI to Prog. Port		✓		
Q2AS	MODEM 2	Q6 TEL	✓			
Q2AS	MODEM 2	Q6 TEL		✓		

PLC	Connection Type	Connection Hardware	Mitsubishi Modem	Non-Mitsubishi Modem	Results	Comments
Q2AS	MODEM 2	QC24	✓			
Q2AS	MODEM 2	QC24		✓		
A2AS	MODEM 1	Q/A6 TEL	✓			
A2AS	MODEM 1	Q/A6 TEL		✓	✓	All DIP switches off in A/Q6 TEL
A2AS	MODEM 2	C24	✓		✓	AT%ADP=2 at PLC end & AT%ADP=0 at PC end
A2AS	MODEM 2	C24		✓		
A2AS	MODEM 2	HMI to Prog. Port	✓			
A2AS	MODEM 2	HMI to Prog. Port		✓		
A2AS	MODEM 2	Q/A6 TEL	✓			
A2AS	MODEM 2	Q/A6 TEL		✓		

**NOTES:**

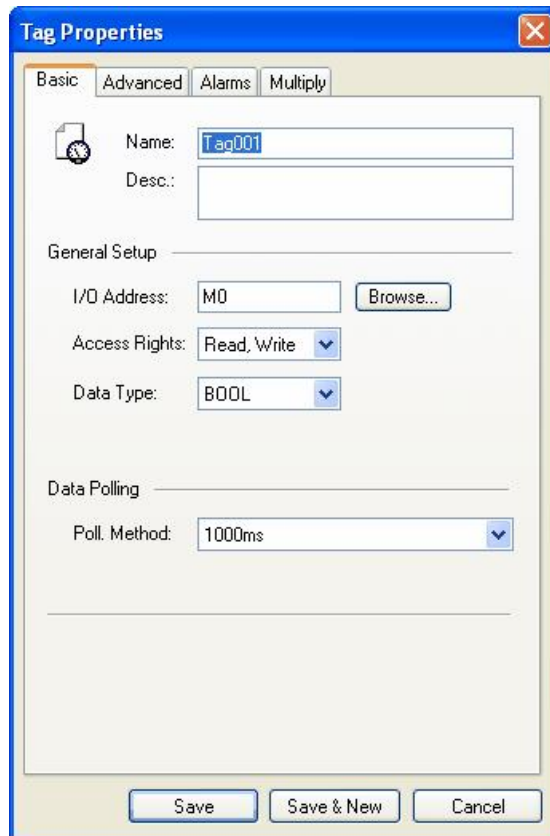
- A. For all non-Mitsubishi modems it is mandatory that the setting &C1 is set and saved in the modem before it is used, e.g. AT&C1&W.
- B. All settings throughout this connection method must be 9600 Baud, no Parity, 8 data bits, 1 stop bit and checksum on. For both modems this means that AT%ADP=T,9600,8,N,1 must be set.
- C. The Q6 TEL must be configured for default settings such as the Modem initialisation string. The only way found to do this was by using features available in GX Developer i.e. from the TOOLS menu use 'Set Tel Data' and its sub-menu.

## 3.4 Data Tags

**Data tags** represent an OPC data item, a register in the device or a range of registers. A symbolic name and description is associated with each tag. An OPC client can obtain the tag description.

### 3.4.1 Basic MX Data Tag Properties

In the **Basic** tab of the **Tag Properties** dialog box, shown below, configure the following settings.



The screenshot shows the 'Tag Properties' dialog box with the 'Basic' tab selected. The 'Name' field contains 'Tag001'. The 'Desc.' field is empty. Under 'General Setup', 'I/O Address' is 'M0', 'Access Rights' is 'Read, Write', and 'Data Type' is 'BOOL'. Under 'Data Polling', 'Poll. Method' is '1000ms'. The 'Save', 'Save & New', and 'Cancel' buttons are visible at the bottom.

**MX Data Tag Properties: Basic Tab**

- **Name:** Enter a logical name for the data item.
- **Description:** Allows you to enter text about the tag. Entries in this field can be very helpful when you go back to look at old configuration or report files, or when you need to modify an existing configuration. The more detailed and specific the information you enter in this field, the easier it is to identify the tag. The description can be up to 128 alphanumeric characters and symbols.

- **I/O Address:** Indicates the data starting address. This may be edited to access a bit within a word for certain register types. For example, if you enter the address 'D100.0' the **Data Type** will automatically change to *BOOL* and the tag will reference bit 0 within the word at address D100. The bit numbers are always entered in decimal - this is different from GX IEC Developer and GX Developer, which both use hexadecimal bit numbering (e.g. GX IEC Developer address D123.B would be entered as D123.11 in MX OPC Configurator). If you enter an address with a bit number higher than 15, it will be adjusted to use a bit number from 0-15 in one of the following words - for example, the address D123.33 would be corrected to D125.1.

Use the **Browse** button to see a list of supported device types and maximum ranges for your hardware. There will be an additional *BIT* entry in this list for word registers that support access to bits within a word.

**Note:** You can also read bit values from a word by using the 'dot' notation when defining the item in the client. For more information, see [Accessing bits](#)<sup>[158]</sup>.

- **Access Rights:** Enables/disables **Read**, **Write**, or **Read/Write** operations.

**Note:** If a tag is defined with only "Write" access, its value will appear as "?" in the monitor view, and it will not be possible to read the tag's value from any other application.

- **Data Type:** Specifies which type of data the tag will expose to OPC clients.

Data Type	Description
INT	Signed 16-bit integer.
UINT, WORD	Unsigned 16-bit integer.
REAL	32-bit float point (IEEE).
BOOL	Digital, one bit.
UDINT, DWORD	Unsigned 32-bit integer.
DINT	Signed 32-bit integer.

To use other types (e.g. STRING and ARRAY types), the data type must be selected in the [Advanced tag properties](#)<sup>[109]</sup>.

The following devices, data types and ranges are supported:

**Note:** The device ranges listed in the table below are the maximum ranges possible on any PLC. The actual range supported depends on the type of PLC used. The **Browse** button next to the **I/O Address** field can be used to view the typical maximum supported ranges for your hardware.

Device	Device	Data Type	Device Range	Size Bits
Link relay	B	BOOL	0-EFFFh	1
Counter coil	CC	BOOL	0-32767	1
Counter Present value	CN	WORD, UINT, INT	0-32767	16
Counter Contact	CS	BOOL	0-32767	1
Data register	D	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING, BOOL	0-4910079	16

Device	Device	Data Type	Device Range	Size Bits
GOT register	EG	DWORD, WORD, UDINT, UINT, DINT, INT, REAL	0-32767	16
Annunciator	F	BOOL	0-32767	1
Free run timer	FT	DWORD, UDINT, DINT	-	32
Latch relay	L	BOOL	0-32767	1
Internal relay	M	BOOL	0-61439	1
Motion device	#	WORD	0-12287	16
File register	R	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING, BOOL	0-32767	16
Link special relay	SB	BOOL	0-7FFFh	1
Special register	SD	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	0-2255	16
Special relay	SM	BOOL	0-2255	1
Retentive Timer Coil	STC	BOOL	0-32767	1
Retentive Timer Present value	STN	WORD, UINT, INT	0-32767	16
Retentive Timer Contact	STS	BOOL	0-32767	1
Link special register	SW	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	0-7FFFh	16
Timer Coil	TC	BOOL	0-32767	1
Timer Present value	TN	WORD, UINT, INT	0-32767	16
Timer Contact	TS	BOOL	0-32767	1
Shared device	U3E nG	BOOL, DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	10000-24335	16
Link register	W	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	0-49FFFFh	16
CC-Link input register	WR	DWORD, WORD, UDINT, UINT, DINT, INT, REAL	0-255	16
CC-Link output register	WW	DWORD, WORD, UDINT, UINT, DINT, INT, REAL	0-255	16
Input relay	X	BOOL	0-1FFFh	1
Output relay	Y	BOOL	0-1FFFh	1
File register	ZR	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING, BOOL	0-16777215	16

- **Polling Method:** Select a polling method from the drop-down list, which lists all polling methods configured in the **Polling Method Definitions** tree control of the Configurator.
- **Save:** Saves all changes specified in the properties dialog box. The data tag appears in the list view for the selected device.
- **Save & New:** Saves all changes specified in the properties dialog box and immediately starts configuration of a new data tag.

- **Cancel:** Closes the properties dialog box.

### 3.4.2 Addressing

**ER:MemoryBlock:Address** – Where the memory block is between addresses 1-256, the MX OPC Configurator will address ER linear in a decimal format.

**Note:** ZR registers must be addressed in a decimal format.

#### Address Ranges

The address ranges available will depend on the PLC type and the hardware parameters that have been configured. For details of the addresses, refer to the hardware manual. To check the maximum possible addresses for a new data tag, use the **Browse** button in the [Basic MX Data Tag Properties](#)<sup>[105]</sup> page.

#### Starting Address

Specifies the location in the device where the selected data block begins.

#### Valid Data Block Registers

X, Y, L, M, SM, SB, TS, TC, TN, CS, CC, CN, STS, STC, STN, D, SD, SW, U3E $\pi$ G, Z, R, ER, B, W, WR, WW, ZR, F, FT, #

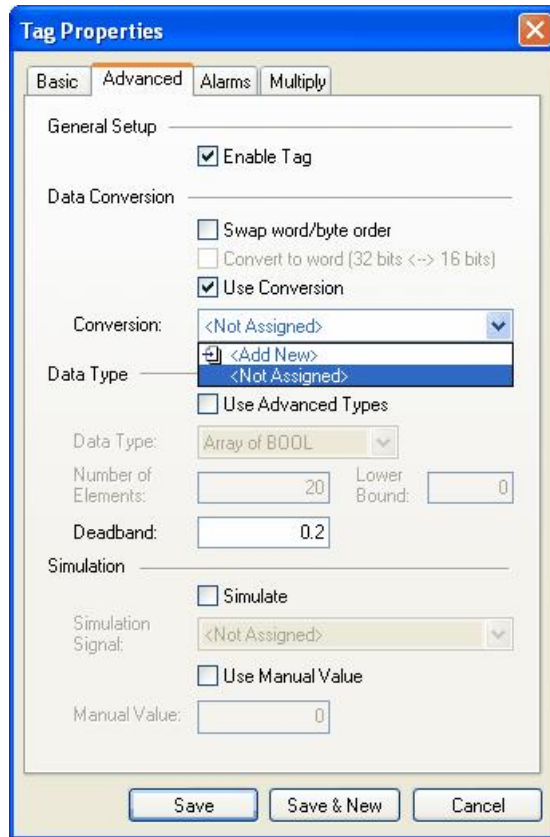
#### Ending Address

Specifies the location in the device where the selected data block ends.



### 3.4.3 Advanced MX Data Tag Properties

In the **Advanced** tab of the **Tag Properties** dialog box, shown below, configure the following settings:



**MX Data Tag Properties: Advanced Tab**

To modify a tag, the tag must first be enabled by checking the **Enable Tag** check box. Once the tag is enabled, all configuration fields can be modified.

- **Save:** Saves all changes specified in the properties dialog box. The data tag appears in the list view for the selected device.
- **Save & New:** Saves all changes specified in the properties dialog box and immediately starts configuration of a new data tag.
- **Cancel:** Closes the properties dialog box.

### Data Conversions

To get the data value converted to another form, choose one of the predefined or user-defined conversions. MX OPC Server includes the following conversions:

**1. Swap word/byte order**

- WORD conversion B1/B2 <-> B2/B1
- DWORD conversion B1/B2 B3/B4 <-> B4/B3 B2/B1

**2. Convert to Word (32bits <-> 16bits)**

If an OPC client requests a VT\_I4 (DINT), where the tag is defined in the MX OPC Configurator as a VT\_I2 (INT), the MX OPC Server will reduce it to VT\_I2 when writing and expand it to VT\_I4 when reading. The sign of the data will be taken into account.

**3. Use Conversion**

To select a conversion definition, check the Use Conversion check box and then select a conversion definition from the drop-down list, which lists all conversion definitions configured in the Conversion Definitions tree control of the Configurator.

### Data Types

Checking the **Use Advanced Types** check box enables the fields under the **Data Types** section. The **Data Type** field specifies which type of data the tag will expose to OPC clients.

Data Type	Description
STRING	Text stored internally as a 16-bit integer.
Array of INT	Array of signed 16-bit integers.
Array of UINT, Array of WORD	Array of unsigned 16-bit integer.
Array of REAL	Array of 32-bit float point (IEEE).
Array of BOOL	Array of digital (bit) values.
Array of UDINT, Array of DWORD	Array of unsigned 32-bit integers.
Array of DINT	Array of signed 32-bit integers.

To use other types (e.g. non-array types), uncheck **Use Advanced Types** and change the type in the [Basic tag properties](#) <sup>105</sup>.

The following devices, data types and ranges are supported:

**Note:** The device ranges listed in the table below are the maximum ranges possible on any PLC. The actual range supported depends on the type of PLC used. The **Browse** button next to the **I/O Address** field on the **Basic** tab can be used to view the maximum supported ranges for your hardware.

Device	Device	Data Type	Device Range	Size Bits
Link relay	B	BOOL	0-EFFFh	1

Device	Device	Data Type	Device Range	Size Bits
Counter coil	CC	BOOL	0-32767	1
Counter Present value	CN	WORD, UINT, INT	0-32767	16
Counter Contact	CS	BOOL	0-32767	1
Data register	D	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING, BOOL	0-4910079	16
GOT register	EG	DWORD, WORD, UDINT, UINT, DINT, INT, REAL	0-32767	16
Annunciator	F	BOOL	0-32767	1
Free run timer	FT	DWORD, UDINT, DINT	-	32
Latch relay	L	BOOL	0-32767	1
Internal relay	M	BOOL	0-61439	1
Motion device	#	WORD	0-12287	16
File register	R	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING, BOOL	0-32767	16
Link special relay	SB	BOOL	0-7FFFh	1
Special register	SD	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	0-2255	16
Special relay	SM	BOOL	0-2255	1
Retentive Timer Coil	STC	BOOL	0-32767	1
Retentive Timer Present value	STN	WORD, UINT, INT	0-32767	16
Retentive Timer Contact	STS	BOOL	0-32767	1
Link special register	SW	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	0-7FFFh	16
Timer Coil	TC	BOOL	0-32767	1
Timer Present value	TN	WORD, UINT, INT	0-32767	16
Timer Contact	TS	BOOL	0-32767	1
Shared device	U3EnG	BOOL, DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	10000-24335	16
Link register	W	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	0-49FFFFh	16
CC-Link input register	WR	DWORD, WORD, UDINT, UINT, DINT, INT, REAL	0-255	16
CC-Link output register	WW	DWORD, WORD, UDINT, UINT, DINT, INT, REAL	0-255	16
Input relay	X	BOOL	0-1FFFh	1
Output relay	Y	BOOL	0-1FFFh	1
File register	ZR	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING, BOOL	0-16777215	16

## Deadband

The Deadband feature is added to the tag properties and is enabled only for real tags. It specifies the maximum acceptable data fluctuation in an exception-based tag. Entering a Deadband value prevents the tag's current value from changing when there is only a slight fluctuation in data.

### Example:

Suppose the current value of a tag is 50.0 and you set a deadband of 20.0. MX OPC Server updates the exception-based tag when incoming values are less than or equal to 30.0 or greater than or equal to 70.

## Simulation

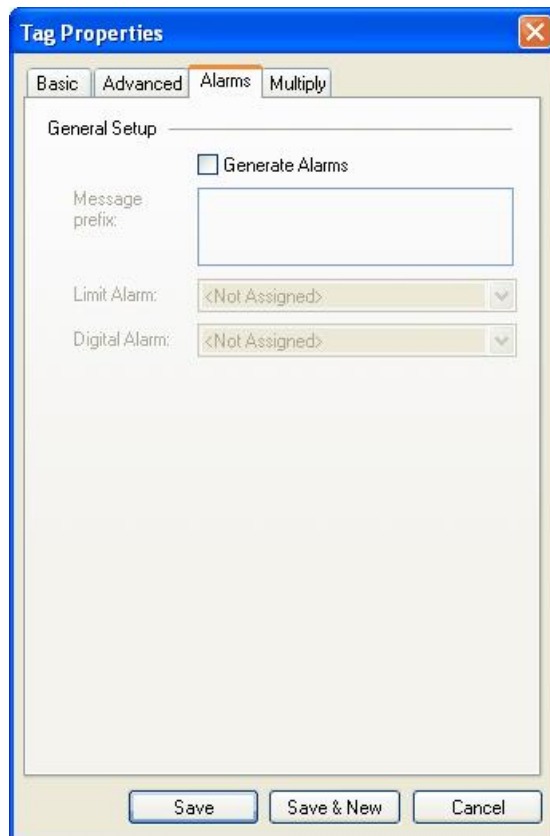
To test the client functionality, check the **Simulate** check box and then choose a **Simulation Signal** from the drop-down list.

Checking the **Use Manual Value** check box enables the fields under the **Manual Value** section. The **Manual Value** field specifies with which value the simulated tag will be initialized.

### 3.4.4 MX Tag Properties: Alarms

In the **Alarms** tab of the **Tag Properties** dialog box, shown below, you can check the **Generate Alarms** check box to make the server generate a limit alarm and/or a digital alarm based on the data item value. The **Message Prefix** parameter is the text of the message for this data item; it will be followed by the text configured for a particular alarm type. The second part of the alarm message will contain the **Message Body** string (see Alarm Definitions). The server allows having any number of Alarm Definitions (templates) predefined. You can then combine one of them with the specific tags.

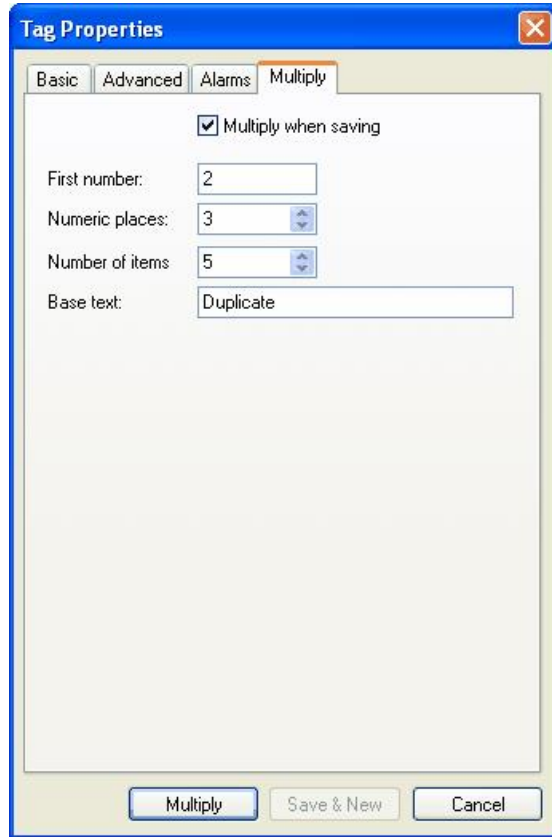
- **Digital Alarm:** Select a Digital Alarm Definition from the drop-down list.
- **Limit Alarm:** Select a Limit Alarm Definition from the drop-down list.
- **Save:** Saves all changes specified in the properties dialog box. The data tag appears in the list view for the selected device.
- **Save & New:** Saves all changes specified in the properties dialog box and immediately starts configuration of a new data tag.
- **Cancel:** Closes the properties dialog box.



MX Data Tag Properties: Alarms Tab

### 3.4.5 Multiply pages on Tag dialogs

This feature is available from the MX Tag property sheets, and can be used to add many tags at the same time. It replaced the 'Multiply' item on the main menu and pop-up menus in earlier versions of MX OPC server. The following screenshots show the Multiply tab on the Tag property sheets.



The **Multiply when saving** checkbox changes the text on the **Save** button to "Multiply" and disables the **Save & New** button. When the user now clicks the **Multiply** button, the tag is saved and multiplied.

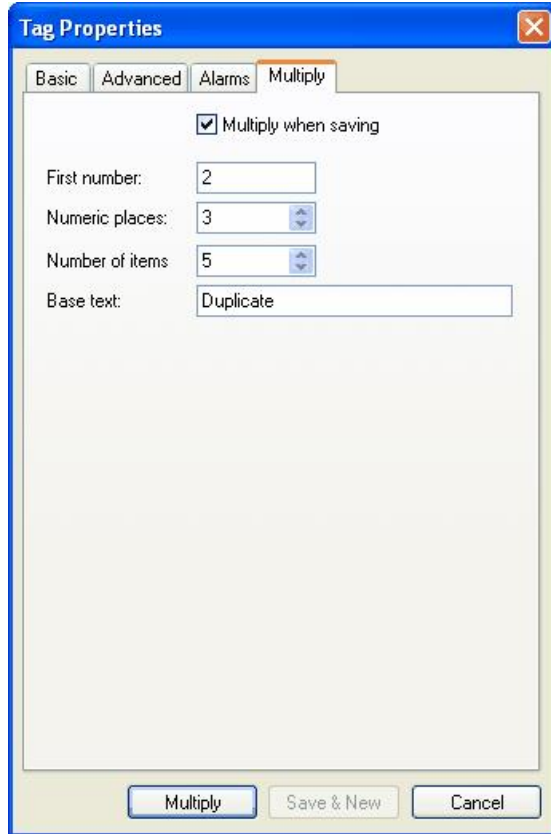
All the items on this property page (except the **Multiply when saving** checkbox) are enabled if and only if the **Multiply when saving** checkbox is checked.

The new tags will be created with addresses immediately following the existing tag, and with other details (such as data type) unchanged. The name is formed from the base text followed by an incrementing number. The fields are:

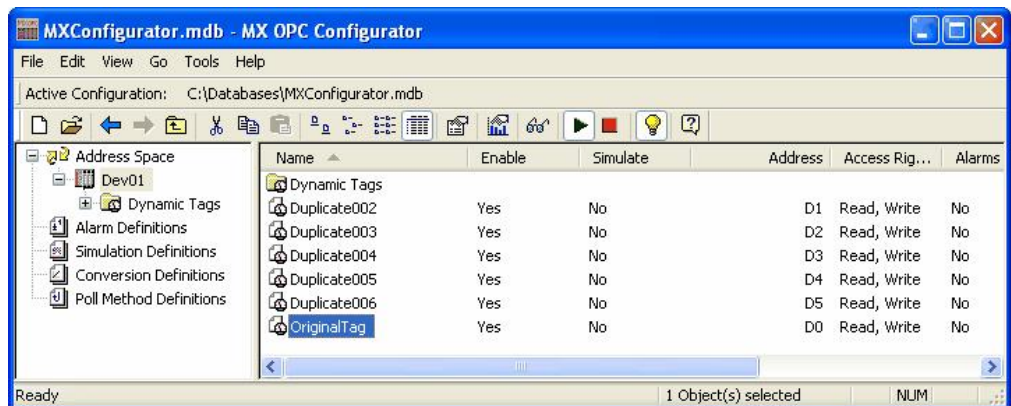
- **Base text** - the text used for the start of the new tag names.
- **First number** - the value to be used for the number part of the first tag
- **Numeric places** - the number of digits to add in the number. For example, if you will be creating 10000 tags with the same base text, using 5 numeric places will ensure that the numbers are sorted as expected in the list.
- **Number of items** - the number of new tags to create (in addition to the original tag which is being saved).

*Example*

Let's assume the user has a device in his configuration with no items. Then he adds a new item, names it "OriginalTag", configures it, but before he closes the window with the "Save" button, he changes to the **Multiply** tab and sets it up as on the next screenshot:



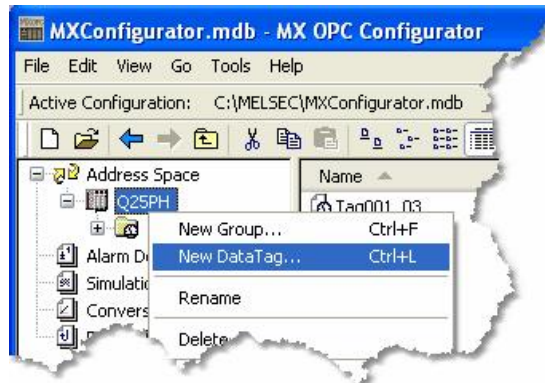
Now, when he pushes "Multiply", the tag is saved and five duplicates created.



### 3.4.6 Adding a New MX Data Tag

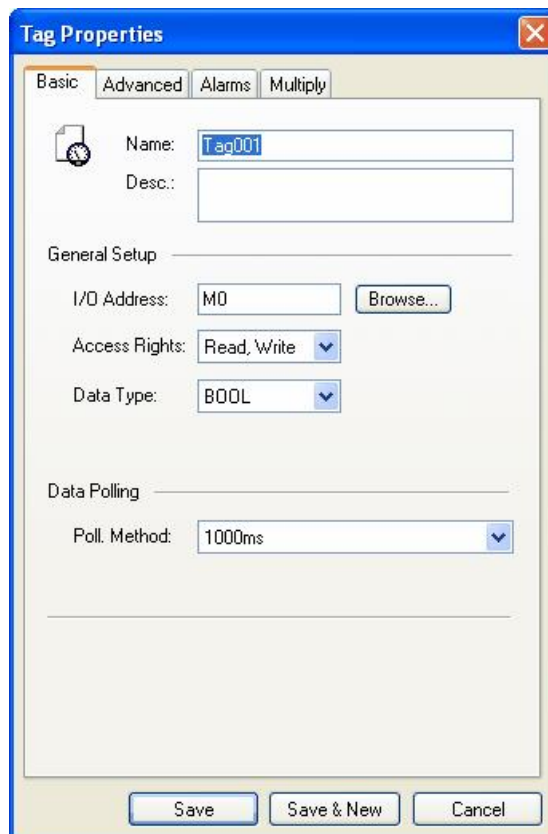
To add a new data tag:

1. Right-click a device on the tree control of the Configurator screen and select **New Data Tag** from the pop-up menu, as shown in the figure below.



**Adding a New MX Data Tag**

2. The **Basic** tab of the **Data Tag Properties** dialog box appears, as shown in the figure below.

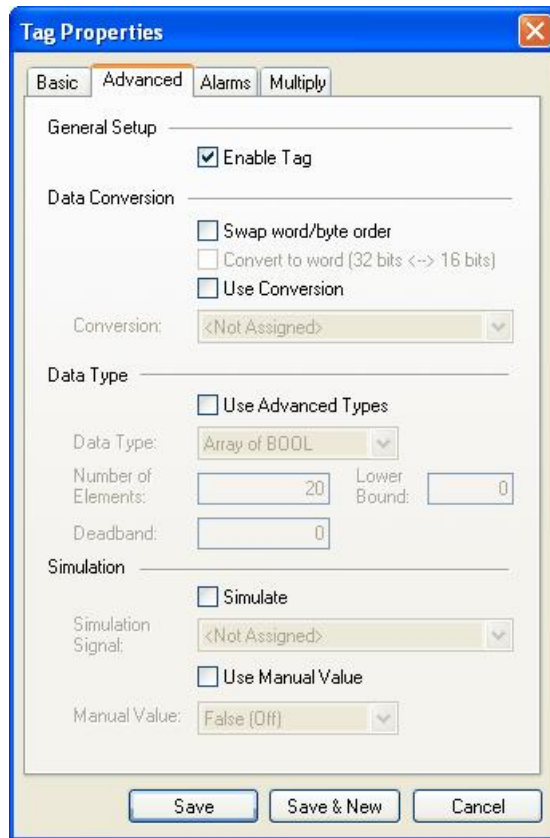


**Configuring MX Data Tag Properties**

3. In the **Name** field, type a name for the new tag, and type a description for the tag (optional).
4. Set the parameters for the **I/O Address** and data **Access Rights**.



5. Click on the **Advanced** tab. Check the **Enable Tag** check box to activate the tag.

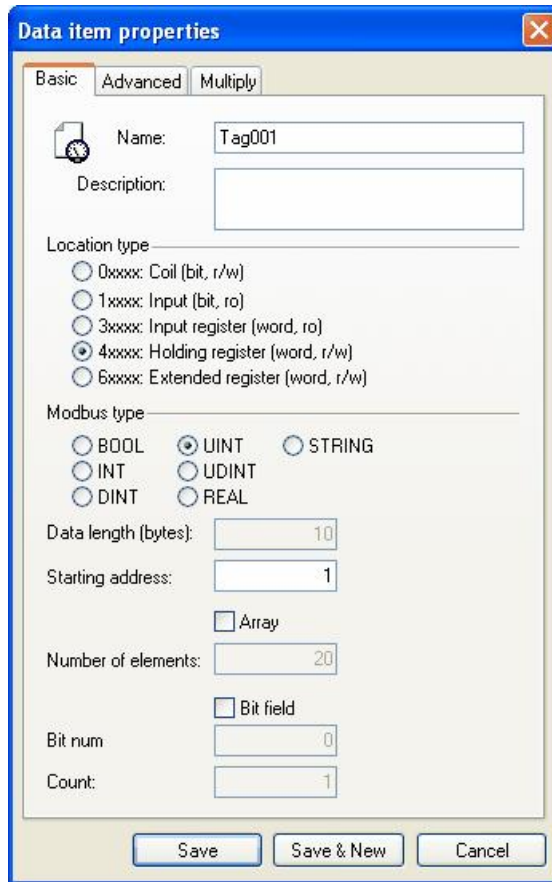


**MX Data Tag Properties: Advanced Tab**

6. Set the **Data Conversion** and **Data Type** parameters for the tag.
7. In the **Alarms** tab of the **Tag Properties** dialog box, you can check the **Generate Alarms** check box to make the server generate a limit alarm and/or a digital alarm based on the data item value.
8. When you have finished configuring the tag properties, click the **Save** button. The new tag appears under the device tree control.

### 3.4.7 Basic Modbus Data Tag Properties

In the **Basic** tab of the **Tag Properties** dialog box, shown below, configure the following settings.



**Basic Modbus tag properties**

- **Name:** Enter a logical name for the data item.
- **Description:** Allows you to enter text about the tag. Entries in this field can be very helpful when you go back to look at old configuration or report files, or when you need to modify an existing configuration. The more detailed and specific the information you enter in this field, the easier it is to identify the tag. The description can be up to 50 alphanumeric characters and symbols.
- **Location type:** Location type is a type of a register in the device. The register types are read as **Input** (1 bit long) or **Input Register** (16 bits), or written to as **Coil** (1 bit) or **Holding Register** (16 bits). The table below explains the name conventions used:

	Read Only	Read / Write
1 bit	Input	Coil
16 bit	Input Register	Holding Register

When the Location type is set as **Input** or **Coil**, the only possible **Modbus Type** is **BOOL**, **Bit Field** is off and disabled.

- **Modbus type:** Allows you to choose how the location type (device data) will be understood as Modbus type (OPC data type). Modbus data type also depends on the Location type selected.
- **Data Length (bytes):** String length, enabled only when **Modbus type** is set to **STRING**
- **Starting address:** Specifies the data item address (register number) in the device data space.
- **Array:** Whether to handle this data item as a vector of values. When **Modbus type** is set to **BOOL** or **STRING**, it is disabled and off.
- **Number of elements:** How many items the vector comprises of. Only enabled when **Array** checked.
- **Bit field:** It is possible to extract bits from the register and use them as a Boolean or integer value (this functionality is read-only). Enabled only for the **BOOL**, **UDINT** and **UINT Modbus Types** when **Location Type** is set to **Input Register** or **Holding Register**.
- **Bit num:** The Boolean bit field starts at this bit position. Enabled only when **Bit Field** checked.
- **Count:** The length of the bit field. Enabled only when **Bit Field** checked.

### 3.4.8 Advanced Modbus Data Tag Properties

In the **Advanced tab** of the **Tag Properties** dialog box, shown below, configure the following settings:

Advanced Modbus tag properties

To modify a tag, the tag must first be enabled by checking the **Enable Tag** check box. Once the tag is enabled, all configuration fields can be modified.

- **Save:** Saves all changes specified in the properties dialog box. The data tag appears in the list view for the selected device.
- **Save & New:** Saves all changes specified in the properties dialog box and immediately starts configuration of a new data tag.
- **Cancel:** Closes the properties dialog box.

## Data Conversions

To get the data value converted to another form, choose one of the predefined or user-defined conversions. MX OPC Server includes the following conversions:

### 1. Swap word/byte order

- **WORD** conversion B1/B2 <-> B2/B1
- **DWORD** conversion B1/B2 B3/B4 <-> B4/B3 B2/B1

### 2. Convert to Word (32bits <-> 16bits)

If an OPC client requests a VT\_I4 (DINT), where the tag is defined in the MX OPC Configurator as a VT\_I2 (INT), the MX OPC Server will reduce it to VT\_I2 when writing and expand it to VT\_I4 when reading. The sign of the data will be taken into account.

### 3. Use Conversion

To select a conversion definition, check the **Use Conversion** check box and then select a conversion definition from the drop-down list, which lists all conversion definitions configured in the **Conversion Definitions** tree control of the Configurator.

## Simulation

To test the client functionality, check the **Simulate** check box and then choose a **Simulation Signal** from the drop-down list.

Checking the **Use Manual Value** check box enables the fields under the **Manual Value** section. The **Manual Value** field specifies with which value the simulated tag will be initialized.

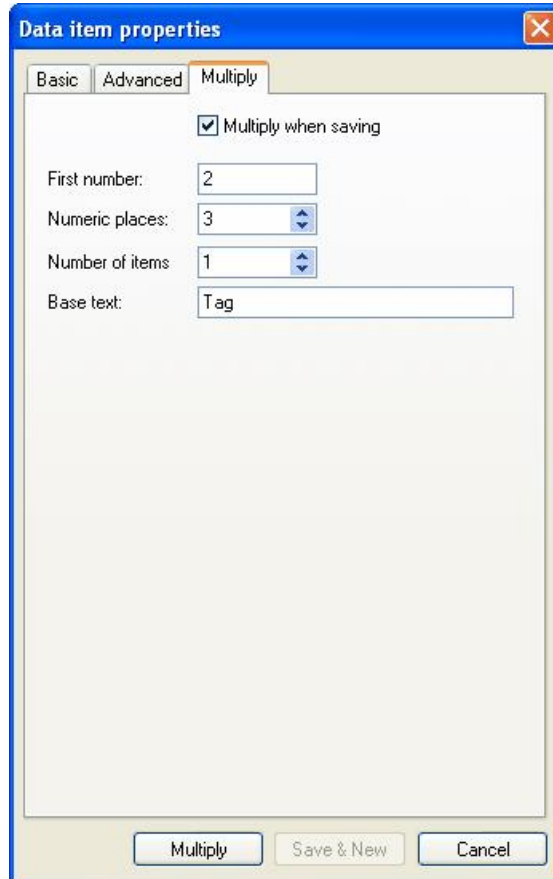
## Alarms

To test the client functionality, check the **Simulate** check box and then choose a **Simulation Signal** from the drop-down list.

- **Generate Alarms:** Check the Generate Alarms check box to make the server generate alarms based on the data item value. When **Modbus type** is set to STRING, it is disabled and off.
- **Mess. Prefix:** The Message Prefix parameter is the text of the message for this data item; it will be followed by the text configured for a particular alarm type. The second part of the alarm message will contain the Message Body string from the alarm definition. Enabled only when **Generate alarms** checked.
- **Limit Alarm:** Choose one of the previously defined limit alarms or create a new one. Enabled only when **Generate alarms** checked and **Modbus type** is not set to BOOL.
- **Limit Alarm:** Choose one of the previously defined digital alarms or create a new one. Enabled only when **Generate alarms** checked and **Modbus type** is set to BOOL.

### 3.4.9 Multiply pages on Modbus Tags

This feature is available from the Modbus Tag property sheets, and can be used to add many tags at the same time. It replaces the 'Multiply' item on the main menu and pop-up menus in earlier versions of MX OPC Server. The screen below shows the Multiply tab on the Tag property sheet.



**Data item properties dialog**

The **Multiply when saving** checkbox changes the text on the **Save** button to "Multiply" and disables the **Save & New** button. When the user now clicks the **Multiply** button, the tag is saved and multiplied.

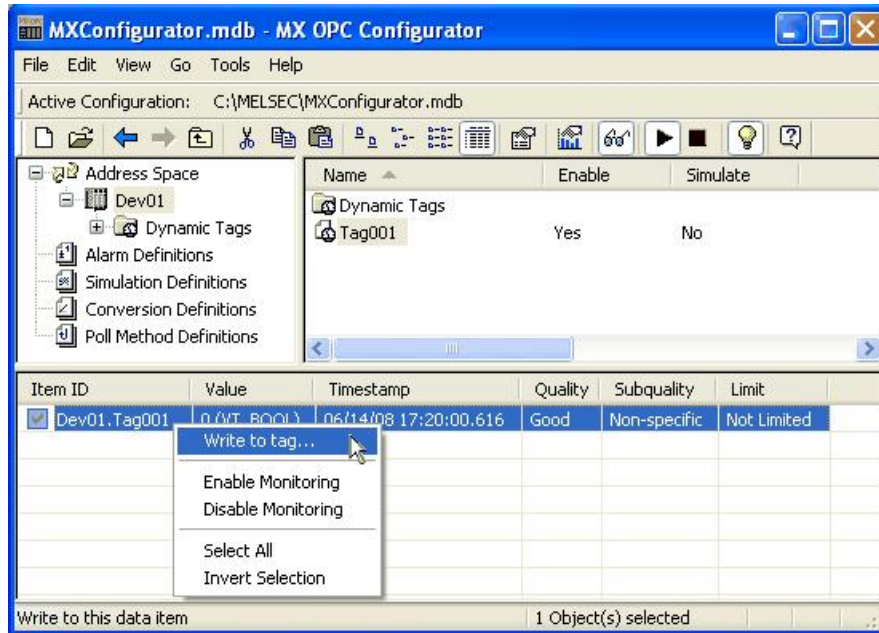
All the items on this property page (except the **Multiply when saving** checkbox) are enabled if and only if the **Multiply when saving** checkbox is checked.

The new tags will be created with addresses immediately following the existing tag, and with other details (such as data type) unchanged. The name is formed from the base text followed by an incrementing number. The fields are:

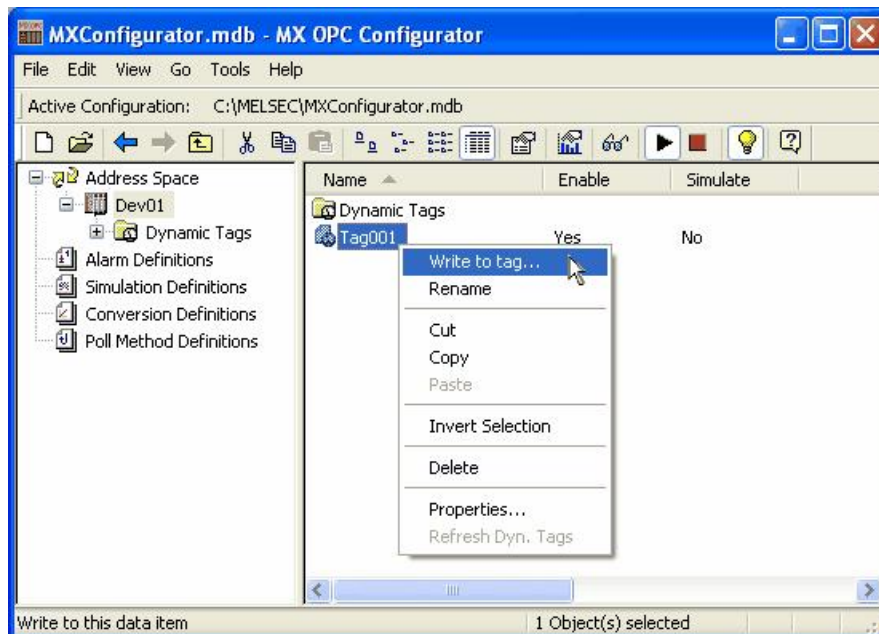
- **Base text** - the text used for the start of the new tag names.
- **First number** - the value to be used for the number part of the first tag
- **Numeric places** - the number of digits to add in the number. For example, if you will be creating 10000 tags with the same base text, using 5 numeric places will ensure that the numbers are sorted as expected in the list.
- **Number of items** - the number of new tags to create (in addition to the original tag which is being saved).

### 3.4.10 Writing to tags

Right clicking on a tag in the Monitor view or List view will open a context menu offering the possibility to write into the tag:

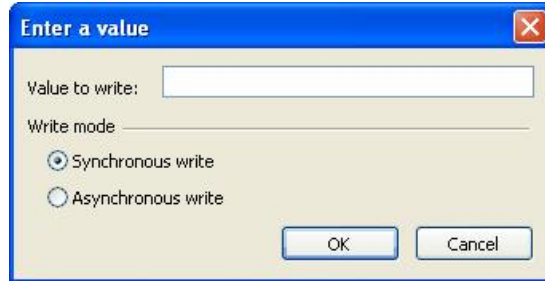


The "Write to tag" menu item is enabled only if exactly one writable item is selected, that is when an MX or Modbus data tag is selected in the Monitor view or the List view.



Clicking the new menu item will open a dialog box, as seen below.

The values accepted depend on the value type of the tag, that the value is written into. For all numeric data types, the value of the number is checked if it is in range of the tag's data type. The radix character for real numbers is a dot, regardless of the current locale.



To write data to arrays, simply separate the elements by commas or semicolons, e.g. writing to an array of floats: 0.1, 0.2, 0.3 or 0.1; 0.2; 0.3. When a semicolon is found in the input value, the configurator assumes that values are separated by semicolons; otherwise it assumes that commas are used as a separator.

Writing to string tags is simple; write the string directly to the edit box. The length of the string is not constrained.

When any of the previous formatting requirements are not met, a dialog box appears showing the error, as in the example below.

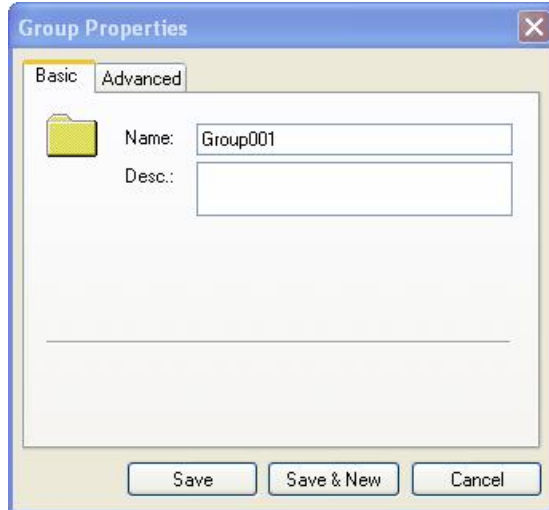


## 3.5 Groups

Data tags can be logically organized into **groups** (folders). You can configure as many folders as required. You can even create subfolders for each group to create a hierarchical organization of tags. The use of folders is optional; tags can be defined under the device level without using folders at all.

### 3.5.1 Basic Group Properties

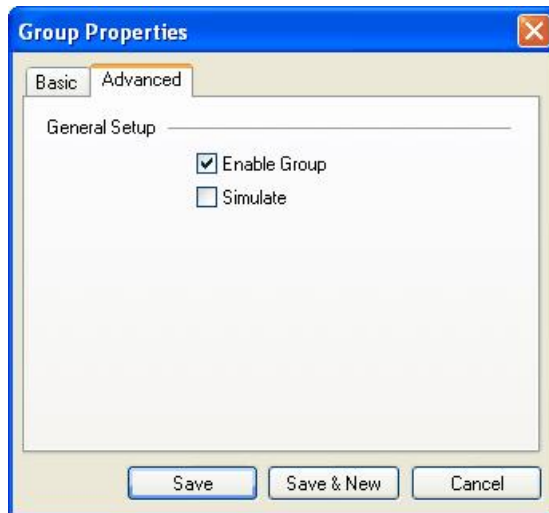
In the **Basic** tab of the **Group Properties** dialog box, shown below, type a name for the group and a description for the group (optional).



Group Properties: Basic Tab

### 3.5.2 Advanced Group Properties

In the **Advanced** tab of the **Group Properties** dialog box, shown below, check the **Enable Group** check box to activate the group. Check the **Simulate** check box if you want to simulate data for the data tags contained within the group.



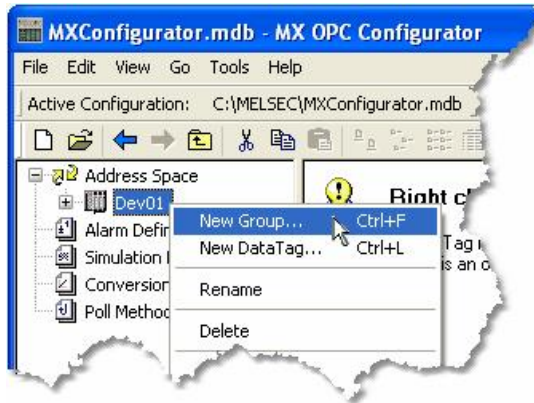
Group Properties: Advanced Tab



### 3.5.3 Adding a New Group

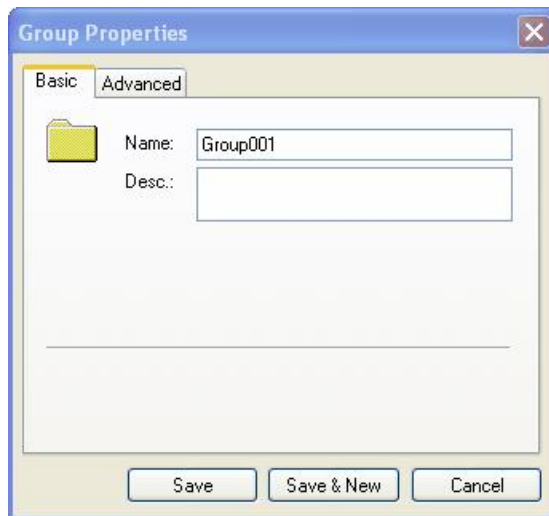
To add a new group:

1. Right-click a device on the tree control of the Configurator screen and select **New Group** from the pop-up menu, as shown in the figure below.



**Adding a New Group**

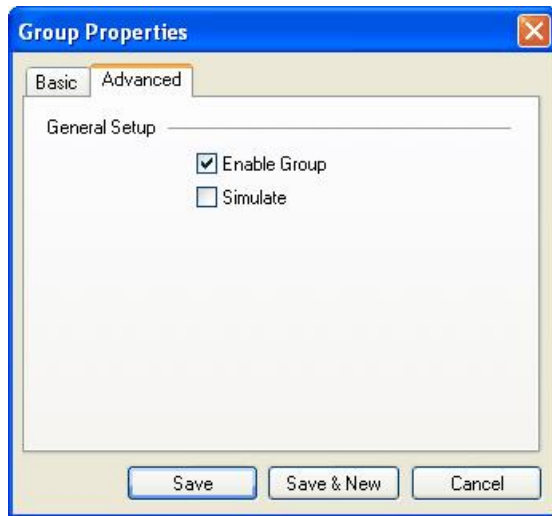
2. The **Basic** tab of the **Group Properties** dialog box appears, as shown in the figure below.



**Configuring Group Properties**

3. In the **Name** field, type a name for the new group, and type a description for the group (optional).

4. Click on the **Advanced** tab. Check the **Enable Group** check box to activate the group.

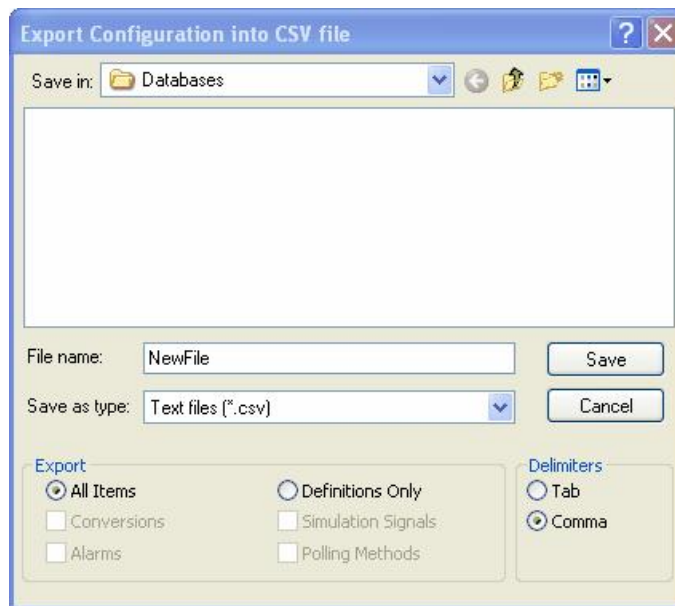


**Group Properties: Advanced Tab**

### 3.6 Exporting Configuration Data to a CSV File

The Configurator offers the flexibility of exporting data from your configuration database to a text (.txt) file or a Microsoft Excel (.csv) file. To export data, select **CSV Export** from the **File** menu. You can export all data, or you can choose to export only alarms and simulation signal data. This opens the **Export Configuration into CSV File** dialog box, as shown in the figure below. By default **All Items** are selected for export, but you can also select to export **Definitions Only** (Conversions, Alarms, Simulation Signals, or Polling Methods).

You can then specify the delimiters for exporting the data. Unless you specify delimiters in the **Export Configuration Data to File** dialog box, the file uses **Commas** as delimiters by default. Each group contains headings and columns that provide information about each item, such as descriptions and associated translations and expressions. It also provides the "tree" pathway for each item. Choose the directory to which you want to export the data from your database. In the **Save As Type** field, choose the file type (.txt or .csv) that you would like to save.

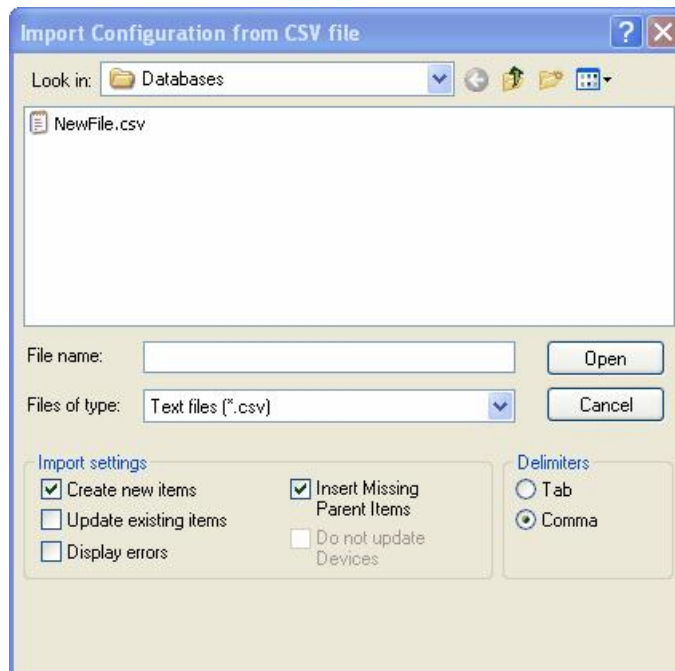


Exporting Configuration Data

### 3.7 Importing Configuration Data from a CSV File

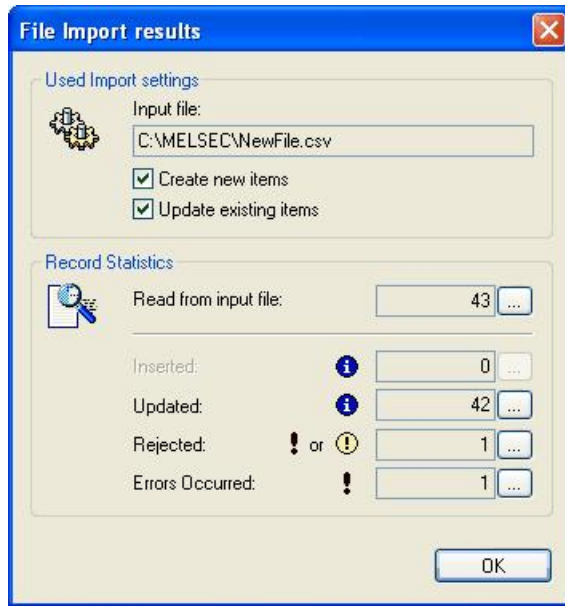
The Configurator offers the flexibility of importing data from a text (.txt) file or a Microsoft Excel (.csv) file to your configuration database. To import data, select **CSV Import** from the **File** menu. This opens the **Import Configuration From CSV File** dialog box, shown below. You can then specify the delimiters and choose from the following import settings:

- **Create new items:** When the import file contains items that are not yet in the configuration database, then it creates them. Otherwise it skips these items.
- **Update existing items:** When the import file contains items that are in the configuration database, then it updates them using data from the import file. Otherwise it skips these items.  
**Note:** Either **Create new items** or **Update existing items** must be selected. Otherwise there is nothing to import.
- **Display errors:** When this item is checked, the Configurator shows a dialog box if an error occurs, and then asks you if you want to proceed with the import. When it is not checked, it skips all items where an error occurred.
- **Insert Missing Parent Items:** When the import file contains parent items that are not yet in the configuration database, then it creates them with the tree control structure of the database. This item is checked by default.



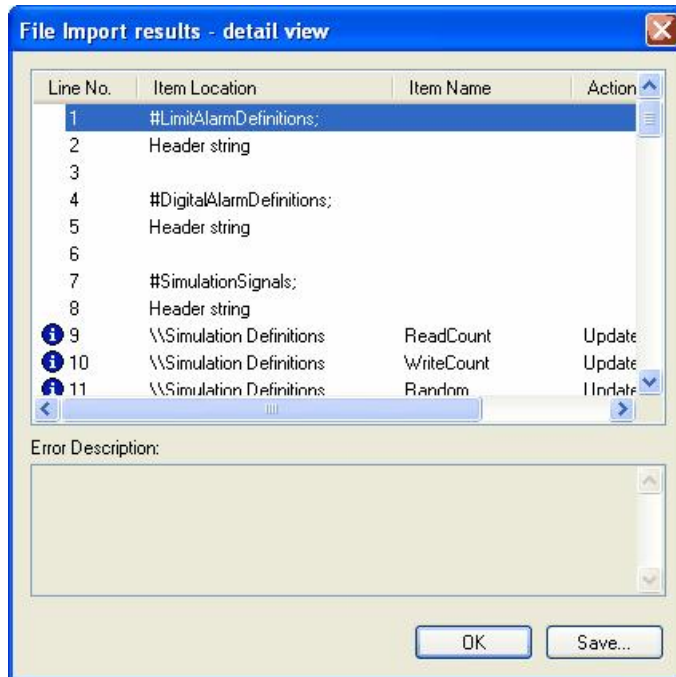
Importing Configuration Data

When you have selected a file to import, click **Open**. When the import is completed, the **File Import Results** dialog box opens, as shown below. This shows the import settings, including the input file name. It also provides a summary of the import, including how many items were inserted, updated, or rejected, and shows how many errors occurred.



**File Import Results Dialog Box**

Click the ... button to the right of each field to get the details view of the import results, as shown in the figure below. This view shows the specific items that were inserted, updated, or rejected, as well as a description of any errors that occurred.



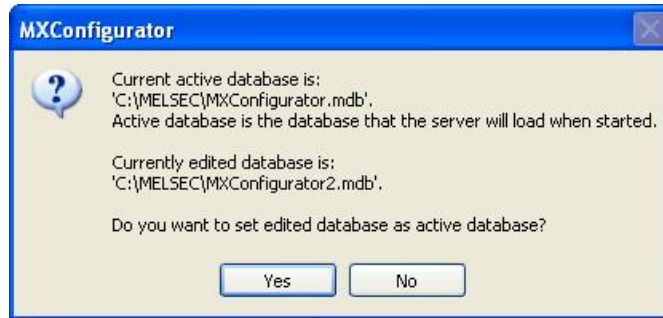
**File Import Results Details**

### 3.8 Activating the Database

Once your configuration is complete, you need to make sure that it is the active database. The database that is currently active is the one that the server uses. To make the current database active, click the **Make Active** button (light bulb icon) on the **Standard Toolbar**.

**Note:** If the button is depressed and the light bulb is yellow, then the current database is already the active database.

A dialog box appears showing both the current active database and database that is currently being edited, as shown in the figure below. To set the edited database as the active database, click the **Yes** button. Next time the server goes into runtime, it will use this active database for all of its operations.



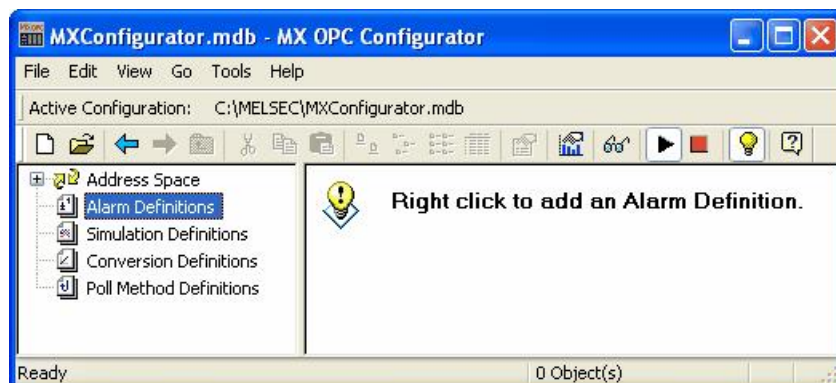
Activating the Database

## 4 Configuring Alarm Definitions

The MX OPC Server is an OPC Alarm and Events server that supports the following alarm types:

- **Limit** (High High, High, Low, Low Low)
- **Digital**

Alarm parameters are set up in the Configurator under the **Alarm Definitions** tree control, shown in the figure below, which is divided into two alarm template types: **Digital** alarm definitions and **Limit** (analog) alarm definitions. Digital alarms can be defined for a data item of "BOOL" type only, while Limit alarms can be defined for all other types except for the "String" data type. Default alarm definitions are provided in the "MXConfigurator.mdb" database.



Alarm Definitions Tree Control

## 4.1 Limit Alarm Definitions

A limit alarm sets the values for four levels of alarms: **LoLo**, **Lo**, **Hi**, and **HiHi**. Limit alarm parameters can have subranges within the data item amplitude. Every subrange definition includes a **Message Body** that will be appended to the alarm message, the **Severity** of the alarm, and the **Req. Ack.** flag for alarm acknowledgement.

### 4.1.1 Basic Limit Alarm Properties

In the **Basic** tab of the **Limit Alarm Properties** dialog box, shown below, configure the following settings:

Limit:	Value:	Message Body:	Severity:	Req. Ack.:
<input checked="" type="checkbox"/> HiHi	100	HiHi Level Alarm	850	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> Hi	90	Hi Level Alarm	500	<input checked="" type="checkbox"/>
<input type="checkbox"/> Return to normal		Return to Normal		
<input checked="" type="checkbox"/> Lo	10	Lo Level Alarm	500	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> LoLo	0	LoLo Level Alarm	850	<input checked="" type="checkbox"/>

**Limit Alarm Properties: Basic Tab**

- In the **Name** field, type a name for the new limit alarm.
- The **Value** field is used to calculate the state of the input fields. For example, a value of 10 for LoLo is compared with the value of the input to determine if the alarm is in LoLo state.
- In the **Message Body** field, enter the warning message that will appear when the alarm is sent. The message can be any text string.
- In the **Return to Normal** field, check the check box and then enter the text that will appear when the alarm is taken care of (e.g. has been acknowledged). The message can be any text string.

**Note:** It is not necessary to enter a message text or a base text. The Server will default to the OPC subcondition name and the OPC condition name. For example, a LoLo alarm will post a description of LoLo limit.

- **Severity** is the OPC-defined value for alarm Priority. The valid OPC severity range is 0 (lowest) to 1000 (highest).
- The **Requires Ack** field is used for OPC condition alarms to determine whether the alarm needs a user acknowledgement. If the **Requires Ack** field is checked, then the alarm requires a user acknowledgement. If the value is not checked, then the alarm is posted as already acknowledged.

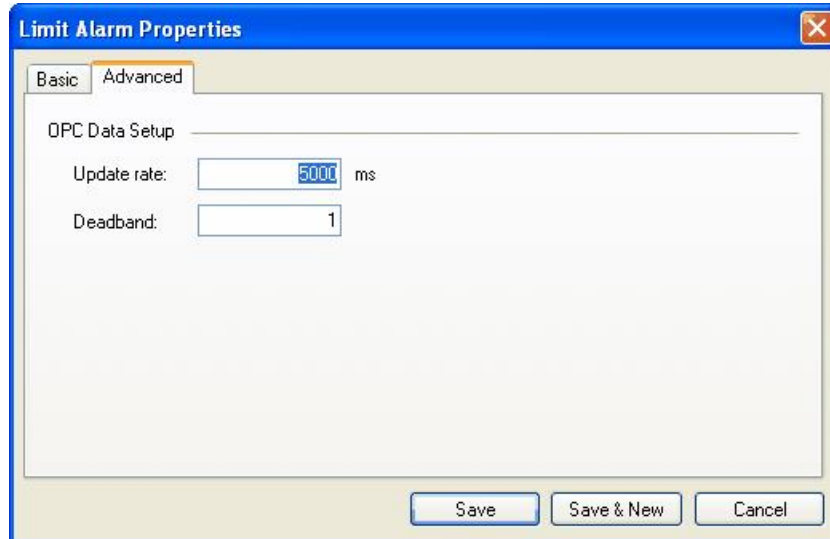
**Note:** Changes to the alarm property fields (HiHi, LoLo, Hi, Lo, Message Text, etc.) in runtime through an OPC tag update will be automatically saved to the database, over-writing any values specified in configuration mode.

- **Save:** Saves all changes specified in the properties dialog box. The alarm definition appears in the list view.
- **Save & New:** Saves all changes specified in the properties dialog box and immediately starts configuration of a new alarm definition.

- **Cancel:** Closes the properties dialog box.

### 4.1.2 Advanced Limit Alarm Properties

In the **Advanced** tab of the **Limit Alarm Properties** dialog box, shown below, configure the following settings:



**Limit Alarm Properties: Advanced Tab**

- **Update Rate:** Defines the frequency of checking the data item value (in milliseconds), and possibly responding by sending the alarm message.
- **Deadband:** Prevents the server from generating a huge amount of alarm messages and overloading the clients when the signal oscillates around one of the limits specified. The deadband value extends the limit zone. It results in sending only one alarm message even if the signal oscillates. Deadband indicates the deadband value to apply to the converted analog values. The deadband value is required and is calculated on borderline alarming limit values to prevent repeated alarm cycles.
- **Save:** Saves all changes specified in the properties dialog box. The alarm definition appears in the list view.
- **Save & New:** Saves all changes specified in the properties dialog box and immediately starts configuration of a new alarm definition.
- **Cancel:** Closes the properties dialog box.



### 4.1.3 Creating a New Limit Alarm Definition

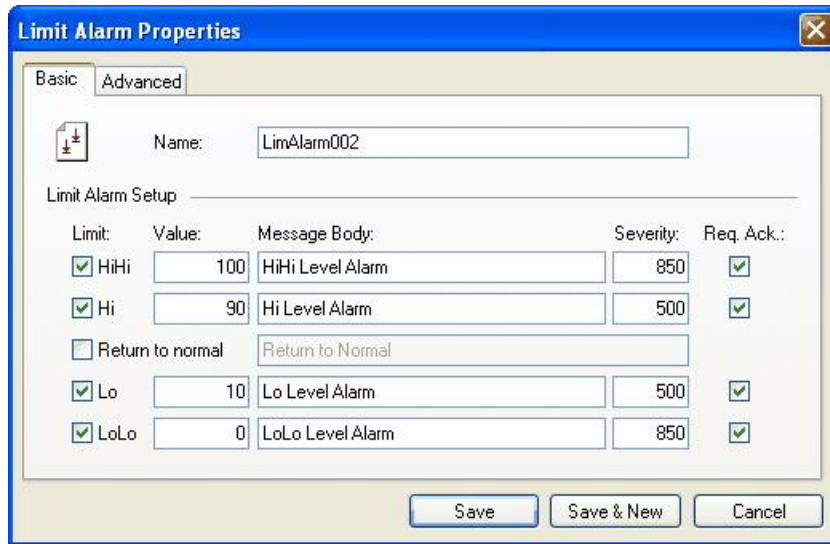
To create a new limit alarm definition:

1. Right-click the **Alarm Definitions** folder on the tree control of the Configurator screen and select **New Limit Alarm Definition** from the pop-up menu, as shown in the figure below.



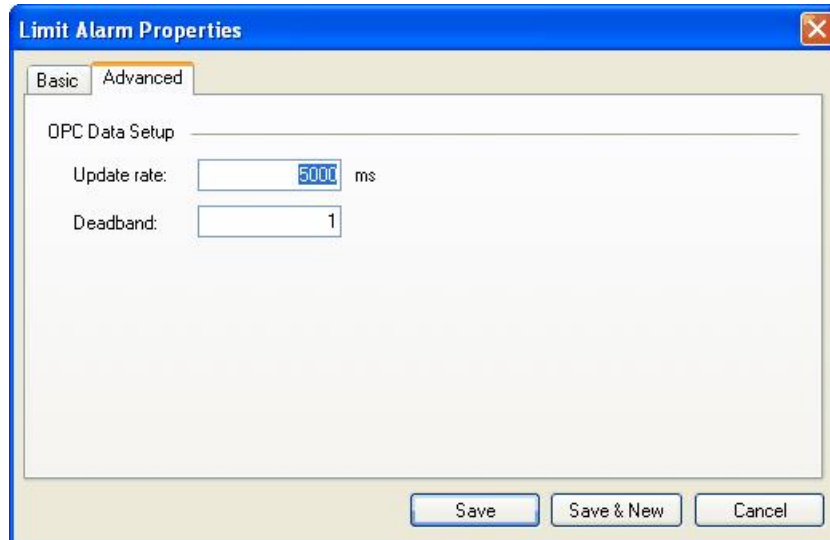
**Creating a New Limit Alarm Definition**

2. The **Basic** tab of the **Limit Alarm Properties** dialog box appears, as shown in the figure below.



**Limit Alarm Properties: Basic Tab**

3. In the **Name** field, type a name for the new alarm definition.
4. Specify the limits, values, and severity levels for the alarm.
5. Click on the **Advanced** tab, as shown in the figure below. Specify a server **Update Rate** (in milliseconds) as well as a **Deadband** value.



**Limit Alarm Properties: Advanced Tab**

6. When you have finished configuring the alarm definition properties, click the **Save** button. The new configuration appears under the **Alarm Definitions** tree control.

## 4.2 Digital Alarm Definitions

A digital alarm sets an alarm if the comparison between the **Alarm State Value** and the input state is TRUE.

### 4.2.1 Basic Digital Alarm Properties

In the **Basic** tab of the **Digital Alarm Properties** dialog box, shown below, configure the following settings:

- In the **Name** field, type a name for the new digital alarm.
- The **Limit** check box enables or disables alarm condition checking.
- Specify a **Value** for the digital alarm (True or False).
- In the **Message Body** field, enter the warning message that will appear when the alarm is sent. The message can be any text string.
- In the **Return to Normal** field, check the check box and then enter the text that will appear when the alarm is taken care of (e.g. has gone back to its normal value). The message can be any text string.

**Note:** It is not necessary to enter a message text or a base text. The Server will default to the OPC subcondition name and the OPC condition name.

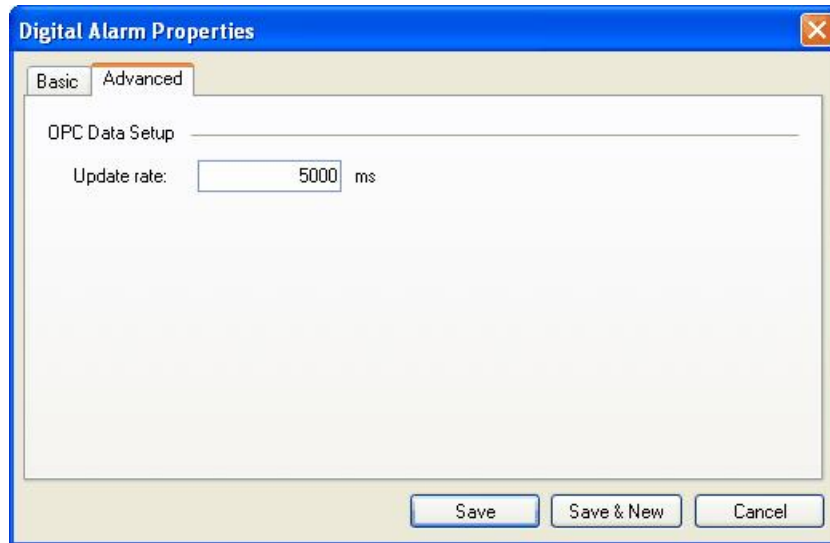
**Digital Alarm Properties: Basic Tab**

- **Severity** is the OPC-defined value for alarm Priority. The valid OPC severity range is 0 (lowest) to 1000 (highest).
- The **Requires Ack** field is used for OPC condition alarms to determine whether the alarm needs a user acknowledgement.
- **Save:** Saves all changes specified in the properties dialog box. The alarm definition appears in the list view.
- **Save & New:** Saves all changes specified in the properties dialog box and immediately starts configuration of a new alarm definition.
- **Cancel:** Closes the properties dialog box.

### 4.2.2 Advanced Digital Alarm Properties

In the **Advanced** tab of the **Digital Alarm Properties** dialog box, shown below, specify the **Update Rate**, which defines the frequency of checking the data item value (in milliseconds), and possibly responding by sending the alarm message.

- **Save:** Saves all changes specified in the properties dialog box. The alarm definition appears in the list view.
- **Save & New:** Saves all changes specified in the properties dialog box and immediately starts configuration of a new alarm definition.
- **Cancel:** Closes the properties dialog box.

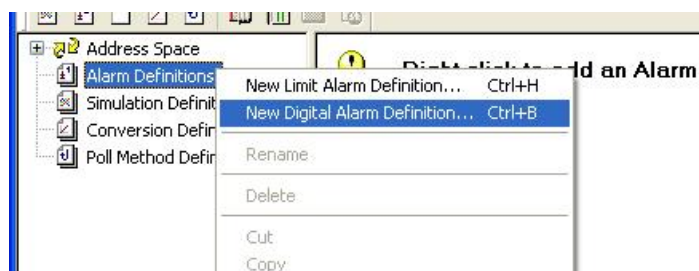


Digital Alarm Properties: Advanced Tab

### 4.2.3 Creating a New Digital Alarm Definition

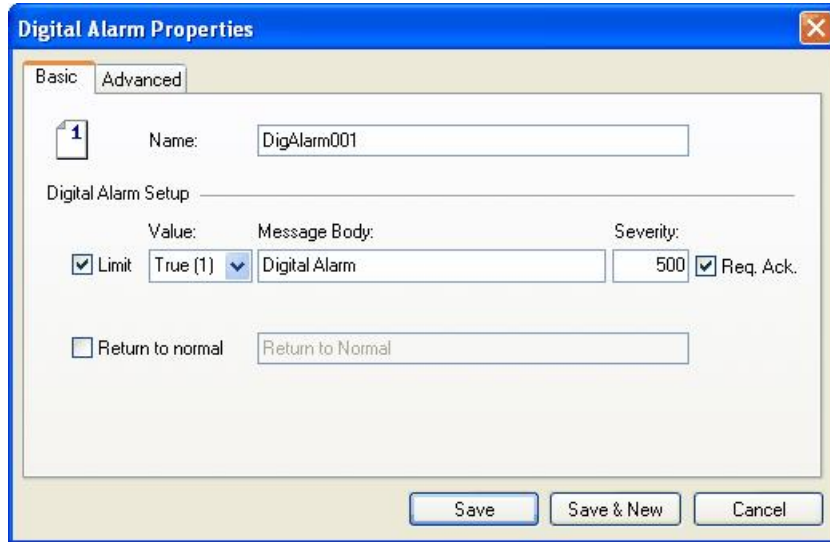
To create a new digital alarm definition:

1. Right-click the **Alarm Definitions** folder on the tree control of the Configurator screen and select **New Digital Alarm Definition** from the pop-up menu, as shown in the figure below.



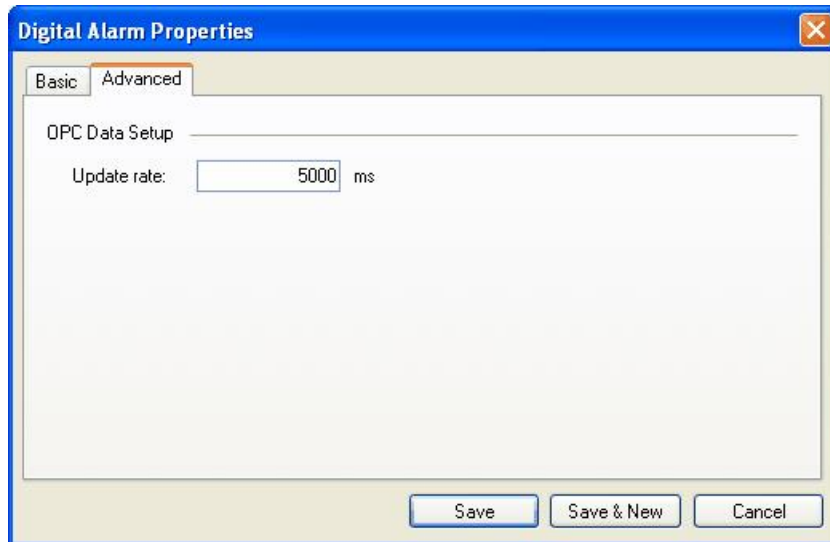
Creating a New Digital Alarm Definition

- The **Basic** tab of the **Digital Alarm Properties** dialog box appears, as shown in the figure below.



Digital Alarm Properties: Basic Tab

- In the **Name** field, type a name for the new alarm definition.
- Specify the value and severity level for the alarm.
- Click on the **Advanced** tab, as shown in the figure below. Specify a server **Update Rate** (in milliseconds).

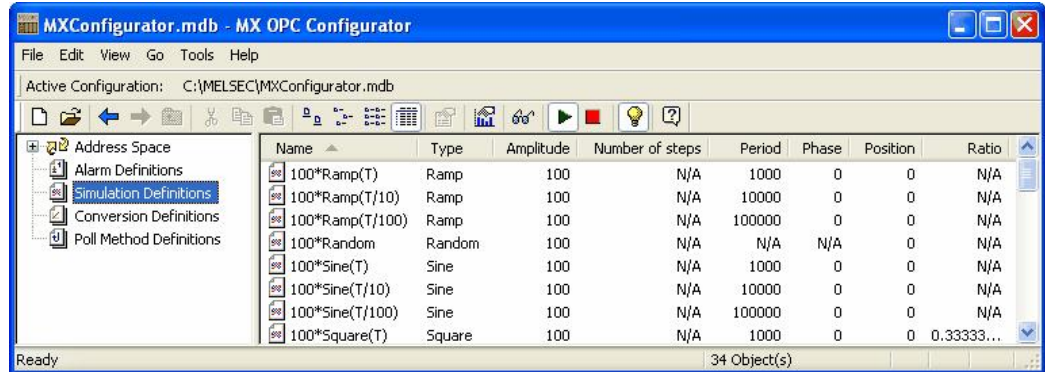


Digital Alarm Properties: Advanced Tab

- When you have finished configuring the alarm definition properties, click the **Save** button. The new configuration appears under the **Alarm Definitions** tree control.

## 5 Configuring Simulation Signals

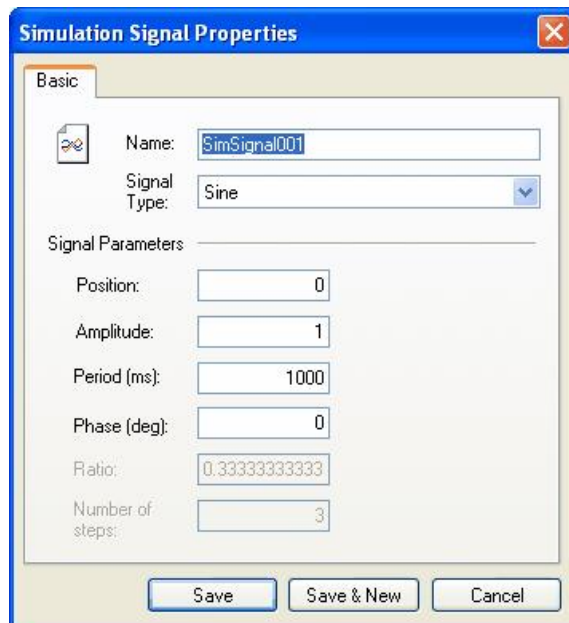
The Configurator offers a wide range of OPC data items in the **Simulation Definitions** tree control, as shown in the figure below.



Simulation Definitions Tree Control

### 5.1 Basic Simulation Signal Properties

In the **Basic** tab of the **Simulation Signal Properties** dialog box, shown below, configure the following settings:



Simulation Signal Properties: Basic Tab

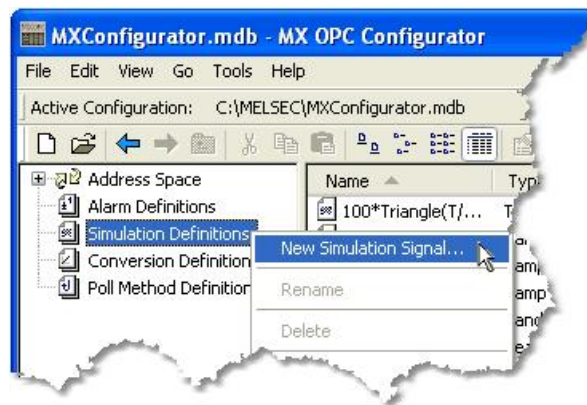
- **Name:** Specifies the name of the selected simulation signal. The name can be up to 50 alphanumeric characters, including underscores ( \_ ) and hyphens ( - ).

- **Signal Type:** For each signal, you can select one of the following signal types from the drop-down list:
  - **Read Count** is incremented by one when any item using the same simulation signal definition is read.
  - **Write Count** increments when any item using the same simulation signal definition is written.
  - **Random** generates random value within the **Amplitude** range starting with **Position**.
  - **Ramp, Sine, Square, Triangle** and **Step** are periodical signals. Their time behavior is influenced by **Period** and **Phase** parameters. **Period** specifies the signal frequency (in milliseconds), while **Phase** moves the signal origin on the time axis (in degrees).
  - **Square** and **Triangle** have one additional parameter: **Ratio**. Ratio defines Triangle signal steepness, or Square signal H/L proportions.
- The **Number of Steps** parameter of the **Step** signal defines the number of steps into which the signal amplitude will be divided.
- **Save:** Saves all changes specified in the properties dialog box. The simulation signal appears in the list view.
- **Save & New:** Saves all changes specified in the properties dialog box and immediately starts configuration of a new simulation signal.
- **Cancel:** Closes the properties dialog box.

## 5.2 Creating a New Simulation Signal

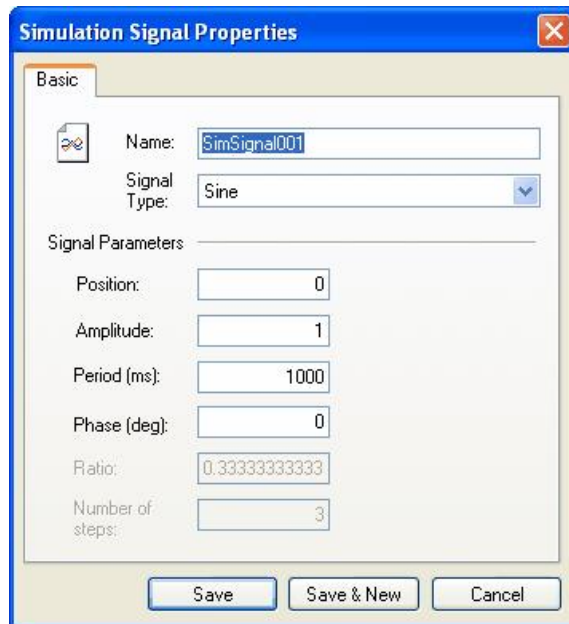
To create a new simulation signal:

1. Right-click the **Simulation Definitions** tree control and select **New Simulation Signal** from the pop-up menu, as shown in the figure below.



Creating a New Simulation Signal

- The properties dialog box for the new configuration appears in the right-hand pane of the Configurator, as shown in the figure below.



**Setting the Properties for the New Simulation Signal**

- In the **Name** field, type a name for the new signal.
- Choose a **Signal Type** from the drop-down list.
- When you have finished configuring the simulation signal properties, click the **Save** button. The new configuration appears under the **Simulation Definitions** tree control.



## 6 MX OPC Server Runtime Module

The MX OPC Server is the actual runtime module. The OPC Server is responsible for accessing the data on the OPC clients' requests. The OPC server has no user interface. It will be started automatically when any OPC client connects to the server.

The runtime module operates directly with the database. Read and write operations are performed directly with no intermediate cache level. It has the effect that every client works with accurate data. In normal situations, the data in the database will be accessed (read/write) by OPC clients. The OPC Server refreshes (with a configurable delay) its tags in order to inform the OPC clients of changes through some external application.

One of the basic concepts of the OPC server is that monitored data are relatively stable in time. By default, the runtime database is located in the same directory in which the runtime module resides.

**Note:** Please ensure that the runtime database exists prior to connecting OPC clients to the OPC server.

The I/O Server is the I/O driver core. The I/O Server contains objects and interfaces that:

- Maintain the I/O driver configuration.
- Read and write process hardware data.

The MX OPC Configurator is a client of the MX OPC Runtime that accesses the I/O Server and lets you view and modify properties for devices and tags. The **Address Space** tree control in the Configurator sets the properties and connection parameters for the following hierarchy of server objects:

- **Devices:** A device is a hardware device or station that uses the I/O driver to communicate with a client PC. The device item contains the properties and methods that govern the behavior of a device. A device is visible to the OPC client.
- **Data tags:** A data tag is an object that makes device data accessible to OPC clients. Data tags can be logically organized into **groups** (folders).

## 6.1 OPC DataSpy

With the introduction of the OPC Foundation Plug and Play communications standard called OPC Data Access, and recently the OPC Alarm and Events Specifications, a universal diagnostics and analysis utility was needed by industry. This need is filled with OPC DataSpy.

### Features of OPC DataSpy

OPC DataSpy includes the following key standard features:

- Internet OPC over TCP/IP tag browsing.
- GenBroker Monitor.
- Simple-to-use OPC test client inspection and diagnostics utility.
- Determination of whether server is OPC compliant.
- Available as a stand-alone OPC test client.
- OPC Data Access (DA) test client.
- OPC Alarm and Events (AE) test client.
- OPC Data Access troubleshooting tool.
- OPC Alarm and Events troubleshooting tool.
- OPC-compliance testing of third-party servers.
- Determination of OPC standards for OPC servers.
- OPC loading and OPC traffic analysis.

The main concept behind the OPC DataSpy is to provide an easy-to-use OPC Test Client to test, diagnose, and troubleshoot industrial applications using the OPC Foundations, OPC Data Access, and Alarm and Events specifications.

The OPC DataSpy tree control capability consists of three main parts:

1. OPC Item Browsing
2. OPC Data and Alarm Monitors
3. GenBroker Monitor

The OPC item browser mode allows you to browse for OPC-compliant Data Access and OPC Alarm and Event servers, as well as OPC Historical Data Access servers, and to provide basic information allowing users to view, test, and troubleshoot any OPC third-party servers. The OPC monitors provide real-time monitoring of OPC server data.

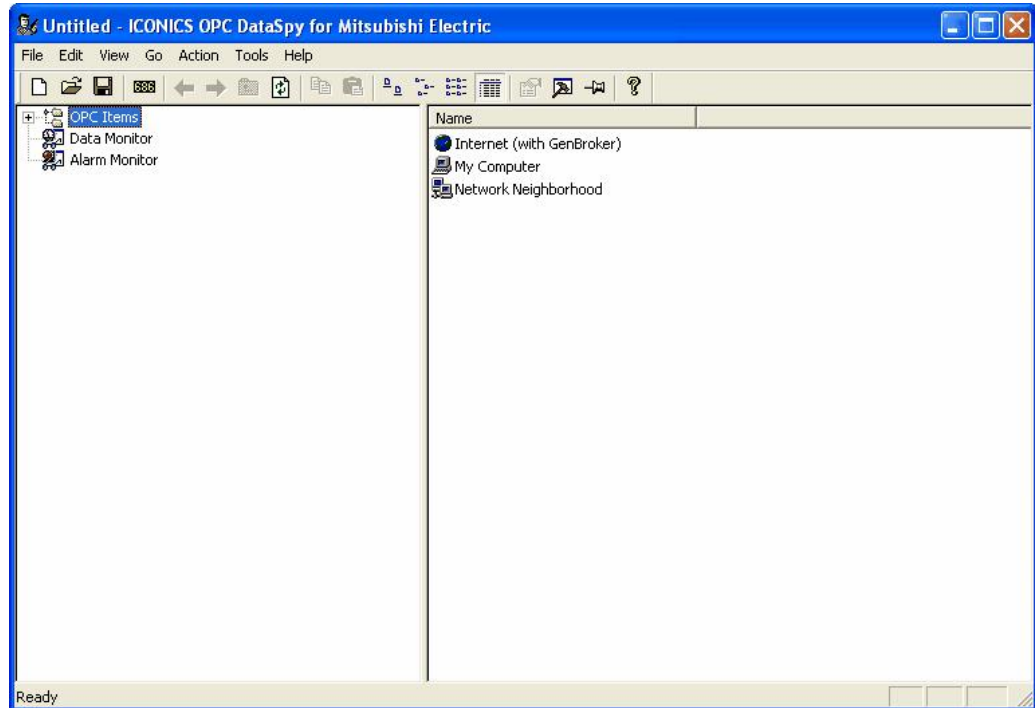
The OPC DataSpy user interface consists of the following elements:

- OPC Item browser
- OPC Data and Alarm Monitors
- Diagnostics and statistics and logging module (output window)

Once the MX OPC Runtime driver is running, you can connect to the MX OPC Server via OPC DataSpy. For this Quick Start, you will use OPC DataSpy provided with the MX OPC Server installation:

1. Open OPC DataSpy from the Windows **Start** menu by selecting **[All] Programs > MELSOFT Application > MX OPC Server 6.05 > OPC Data Spy > DataSpy.exe**. This opens OPC DataSpy, as shown in the figure below.

**Note:** The name of the program folder may vary depending on your local settings (e.g. language settings).

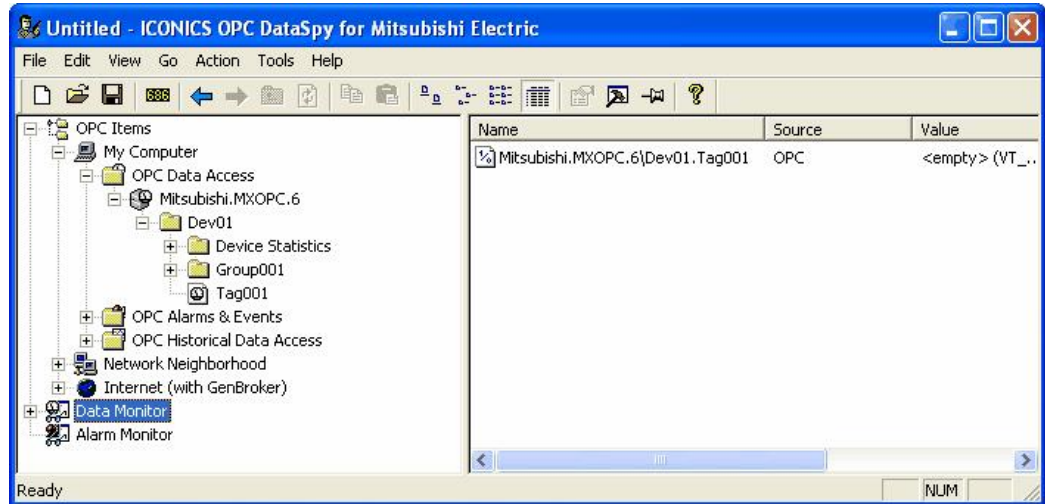


OPC DataSpy Screen

2. From the left navigation pane, expand the **"OPC Items"** item by clicking on the "[+]" symbol to the left of the icon. Expand the **"My Computer"** section then the **"OPC Data Access"** section then the **"Mitsubishi.MXOPC.6"** section.

Users will then see all configured devices listed underneath **"Mitsubishi.MXOPC.6"** and can browse groups and data tags accordingly.

- Right click on any selected tag listed beneath "**Mitsubishi.MXOPC.6**" and click on "**Monitor**". Next, click on "**Data Monitor**" that is listed underneath OPC Items when that section is fully collapsed. In the right navigation pane, users will be able to see the values associated with selected tags.



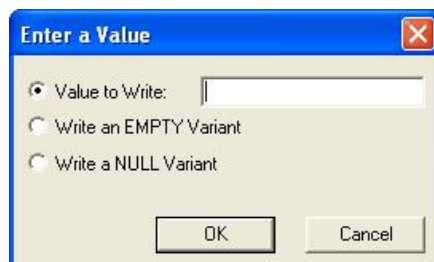
- If your client application does not support reading a single bit within a word or a single item in an array tag, you may still be able to use these features by changing the name of the item that you add.

To access a bit within a word, define a new tag with an address such as D12.1 (bit 1 within the word at address D12, with bit numbering starting a 0 for the least significant bit) and set the type to BOOL. If write access is not required, a client can also address a bit within a word tag using the item syntax "tagname.bitNr" (e.g. tagname.0 for bit 0) - this is a read-only custom "OPC property" of the tag item.

A client can address a single element of an array tag using the item syntax 'arraytag[elemNr]' (e.g. arraytag[2] for element 2).

- To Write to Tags, right-click on the tag name in the right navigation pane. The "Enter a Value" dialog pops up. Users can then enter values in the text entry box.

**Note:** If you are writing a floating point value, the period (.) character is always used as the radix character (e.g. '123.4' not '123,4'). If you are writing to an array item, the values should be separated by commas (,).

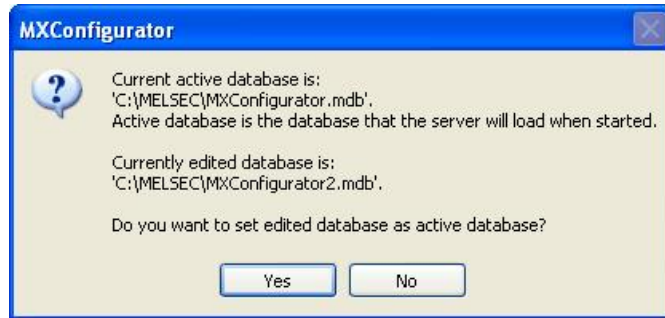


## 6.2 Activating the Database

Once your configuration is complete, you need to make sure that it is the active database. The database that is currently active is the one that the server uses. To make the current database active, click the **Make Active** button (light bulb icon) on the **Standard Toolbar**.

**Note:** If the button is depressed and the light bulb is yellow, then the current database is already the active database.

A dialog box appears showing both the current active database and database that is currently being edited, as shown in the figure below. To set the edited database as the active database, click the **Yes** button. Next time the server goes into runtime, it will use this active database for all of its operations.

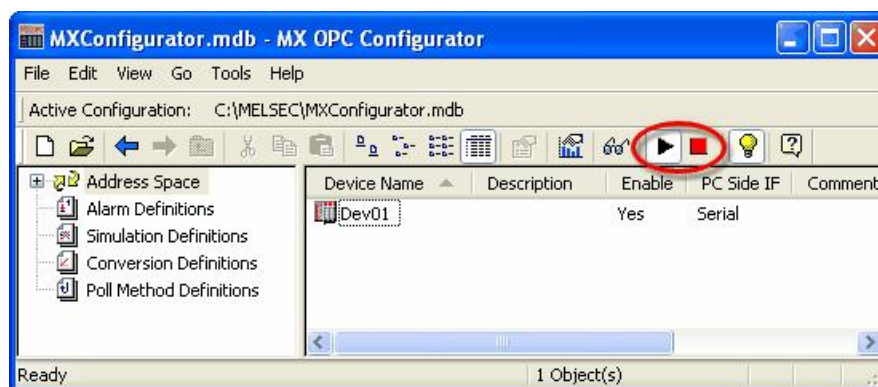


Activating the Database

## 6.3 Starting and Stopping the Driver

The I/O Server maintains the driver's device and data tags, performs all required functions for communicating with the process hardware, and exposes the methods and properties to other applications. This server provides a software layer between all client applications and the Mitsubishi ActiveX Communication support Tool (MXComponent). The MXComponent tool handles all the low-level communication details by providing a set of interfaces that the I/O Server uses.

To enable client/server communication, you must start the driver by clicking the **Start** button (green triangle icon) on the Standard toolbar of the MX OPC Configurator. To stop the driver, click the Stop button (red square icon) on the Standard toolbar, as shown in the figure below.



Starting and Stopping the Driver

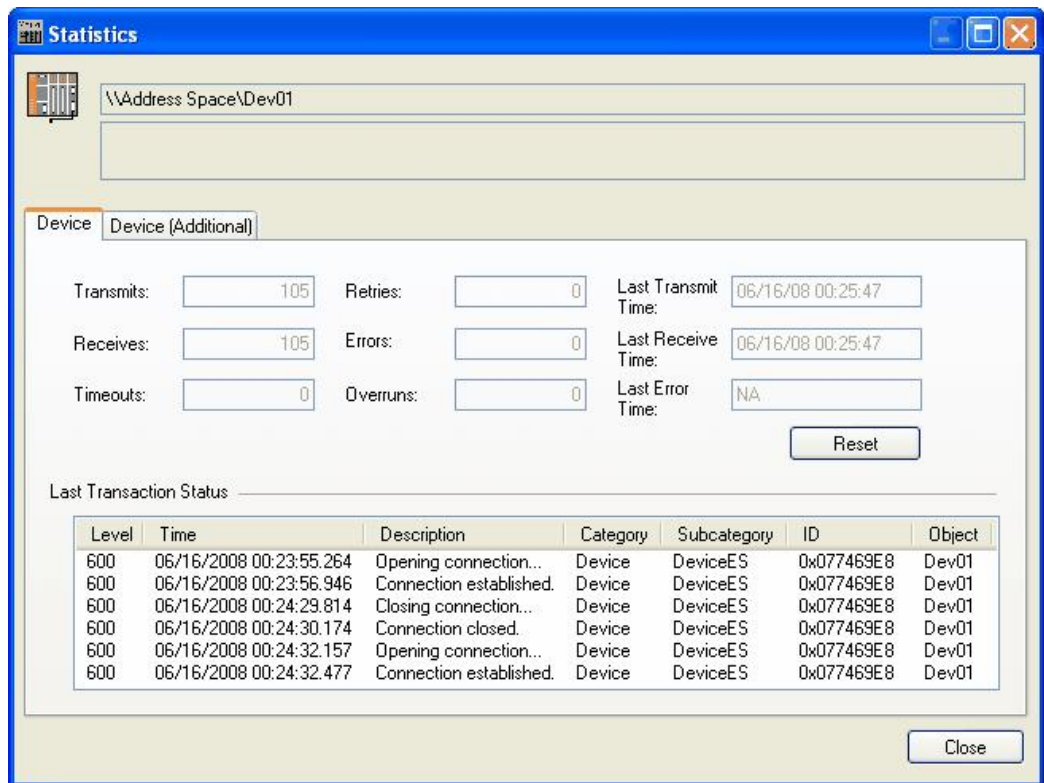
### 6.4 Statistics Mode

The MX OPC Configurator includes a Statistics Mode for viewing the data statistics of your I/O driver while it is running. Statistics are provided for the following:

- Devices
- Data tags

To change to statistics mode:

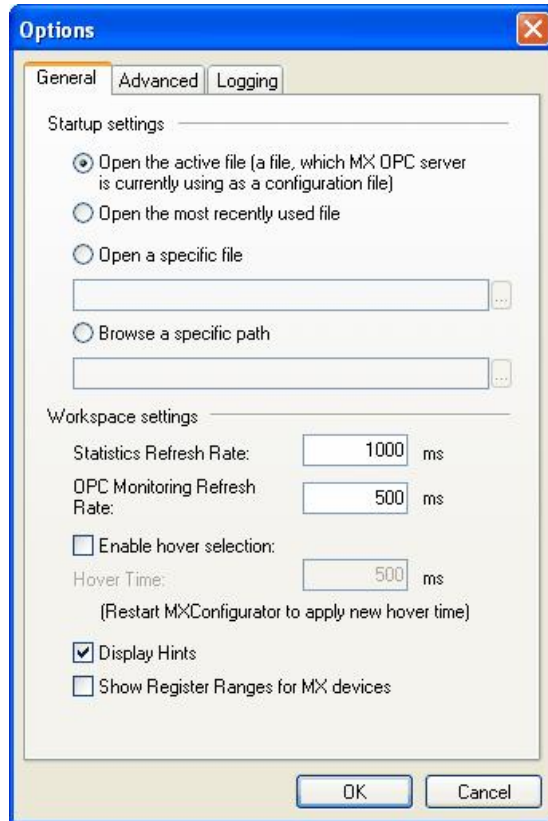
1. Select **Statistics** from the **View** menu.
2. The Configurator switches the current dialog view to data statistics mode, as shown in the figure below.



Statistics Mode View

### 6.4.1 Setting the Statistics Mode Refresh Rate

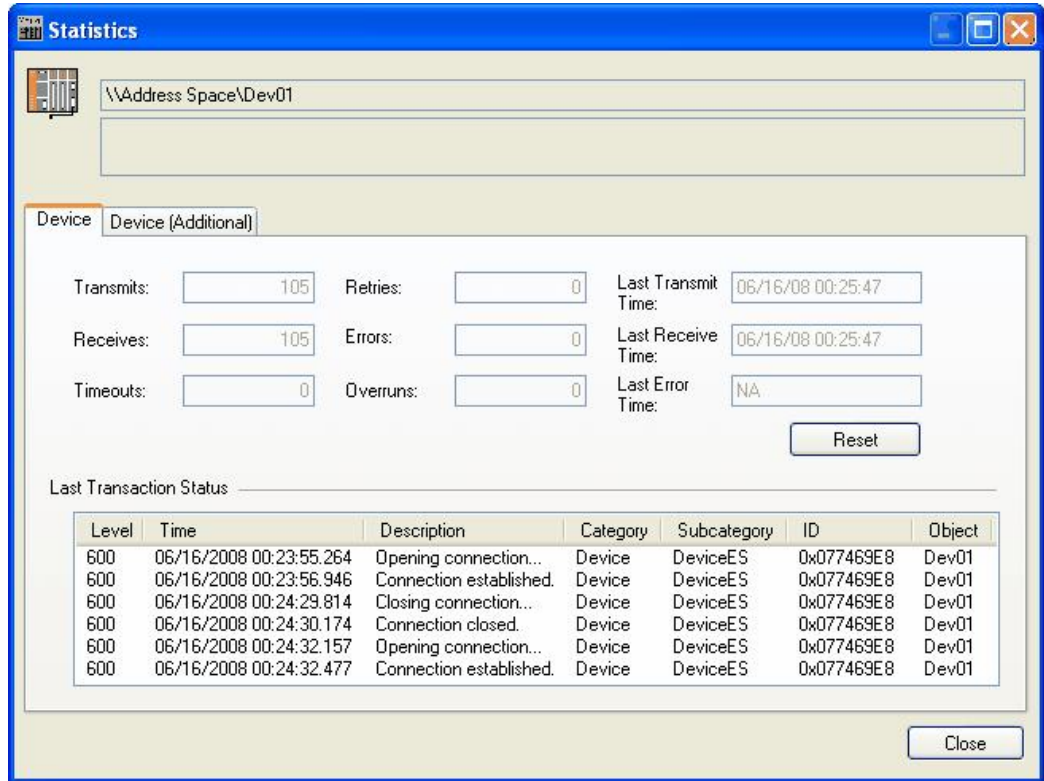
To set the Statistics Mode **Refresh Rate** (in milliseconds), select **Options** from the **Tools** menu and click on the **General** tab of the **Options** dialog box, as shown in the figure below.



Options Dialog Box: General Tab

### 6.4.2 Device Statistics

To view device statistics in statistics mode, click on a device or tag under the **Address Space** tree control, as shown in the figure below.



Viewing Runtime Data Statistics for Devices

**Transmits:** Displays the number of messages sent to the process hardware from the selected device.

**Example**

- Number of data blocks configured for the driver = 8
- Number of messages sent from DataBlock1 = 5
- Number of messages sent from DataBlock2 = 6
- Number of messages sent from DataBlock3 = 3
- Number of messages sent from DataBlock4 = 12
- Number of messages sent from DataBlock5 = 16
- Number of messages sent from DataBlock6 = 6
- Number of messages sent from DataBlock7 = 6
- Number of messages sent from DataBlock8 = 6
- Total number of messages sent = 60
- Transmits field displays: 60

**Receives:** Displays the total number of messages the device received from the process hardware. The **Receives** field includes both valid and invalid messages. Use the Errors field for the number of invalid messages.



**Timeouts:** Displays the total number of messages sent to the process hardware from the selected device that did not receive a reply. Timeouts result from the device property **Reply Timeout**. This property specifies the length of time that the I/O driver waits for a response from the process hardware. If the driver sends a message to the hardware and does not receive a response within the specified Reply Timeout value, a timeout occurs and increments the **Timeouts** field.

Modem transmissions are particularly vulnerable to timeouts because of the length of time it takes to establish communication. If you have a high timeout count:

- Try increasing the value for the Reply Timeout property.
- Check your cable. You may have a bad connection.
- Verify that your channel properties match the process hardware. If you don't know your hardware settings, refer to your process hardware documentation.

**Retries:** Displays the total number of messages resent to the process hardware because of errors. A value in the **Retries** field for a device does not indicate a communication problem. It may, for example, indicate that the process hardware is slow replying to driver requests.

Retries are a timing property that you set for each device. The driver uses the value you enter to determine how many times to resend a request after a timeout occurs. Possible reasons for the driver timing out include:

- Checksum errors from communication line interference.
- Busy PLCs. PLCs may not reply if they are involved in another task.
- Hardware problems such as a faulted PLC. The hardware cannot recover unless you clear the fault.

**Errors:** Displays the total number of protocol errors that were sent from the process hardware and received by the selected device. Protocol errors occur when:

- The process hardware receives a message with a checksum error. Typically, interference or a loose connection can cause checksum errors.
- The driver requests data from a register that does not exist in the process hardware.

For more information on the cause of the error, check the **Last Error** field for all data blocks configured for the driver.

**Overruns:** Displays the total number of overruns for the device. An overrun occurs when the driver sends more messages to a data block than it can process at one time. A high number in the Overruns field indicates that your driver cannot poll all of the enabled data blocks at the rate specified. It also indicates that your data is not updating at the specified poll rate.

**Note:** Setting the poll rate to zero forces the driver to run at its maximum rate. Overruns are disabled for poll rates equal to 0. Set the poll rate equal to 0 if it is more important that the driver run at its maximum speed than detect overruns.

**Last Transmit Time:** The last time data was sent to the device.

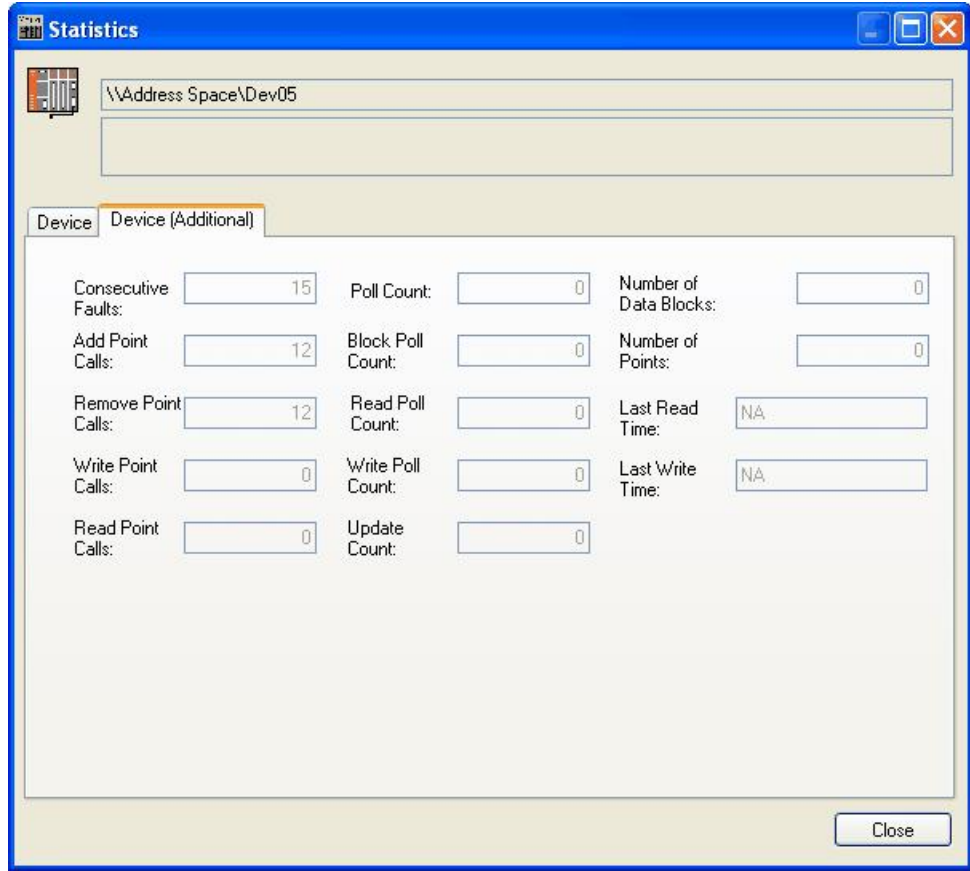
**Last Receive Time:** The last time data was successfully received from the device.

**Last Error Time:** The last time an error was observed on the device, including connection errors.

**Reset:** Resets all data statistics.

### 6.4.3 Device Statistics (Additional)

To view additional device statistics in statistics mode, click on a device or tag under the **Address Space** tree control, and then select the **"Device (additional)"** tab, as shown in the picture below.



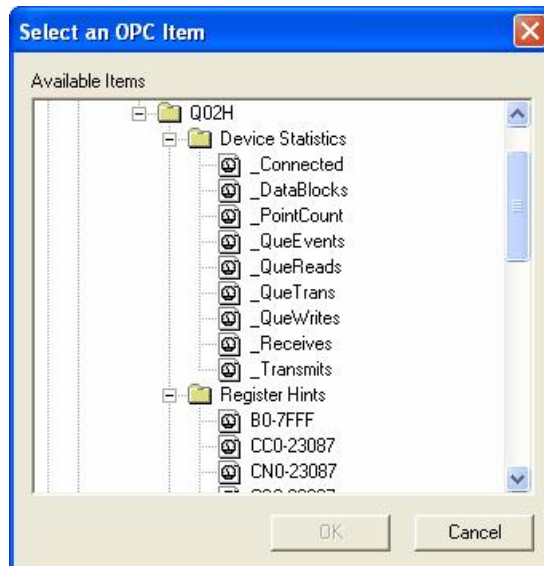
Viewing Runtime Data Statistics for Devices (Additional)

The additional statistics are:

- **Consecutive Faults** – Displays the total number of faults without an interposing success on this device.
- **Add Point Calls** – The total number of points added during the lifetime of this device. Note that the device lifetime is not the same as the running time of the server, as the server may release the device altogether, if no points are requested.
- **Remove Point Calls** – Similar to **Add Point Calls**
- **Write Point Calls** – The count of sync/async write calls; similar to **Add Point Calls**
- **Read Point Calls** – Count of sync read calls; similar to **Add Point Calls**
- **Poll Count** – How many times the internal calendar triggered polling of blocks. One or more blocks can be polled per one Poll Count increase (see Block Poll Count).
- **Block Poll Count** – How many blocks were polled during regular update. Extra read requests and write requests do not increase this counter.
- **Write Poll Count** – Counts the blocks created during writes to the device.
- **Update Count** – The total number of tags that were processed in the Block Polls, Read Polls and Write Polls.
- **Number of Data Blocks** – The total number of data blocks this device uses. Data blocks are allocated automatically by the runtime module depending on the values set for the [advanced device properties](#)<sup>[69]</sup>.
- **Number of Points** – The total number of tags that the device manages currently.
- **Last Read Time** – The last time any data was explicitly read from the device.
- **Last Write Time** – The last time data was written to the device.

#### 6.4.4 Statistical Tags in OPC Address Space

In a similar way to the "Hints", the MX Runtime can optionally expose statistics tags in a group called "Device Statistics", as seen on the next screenshot:



Which tags are shown will depend on the settings in the **Advanced** tab of the [Options dialog](#)<sup>[58]</sup>. The possible settings are:

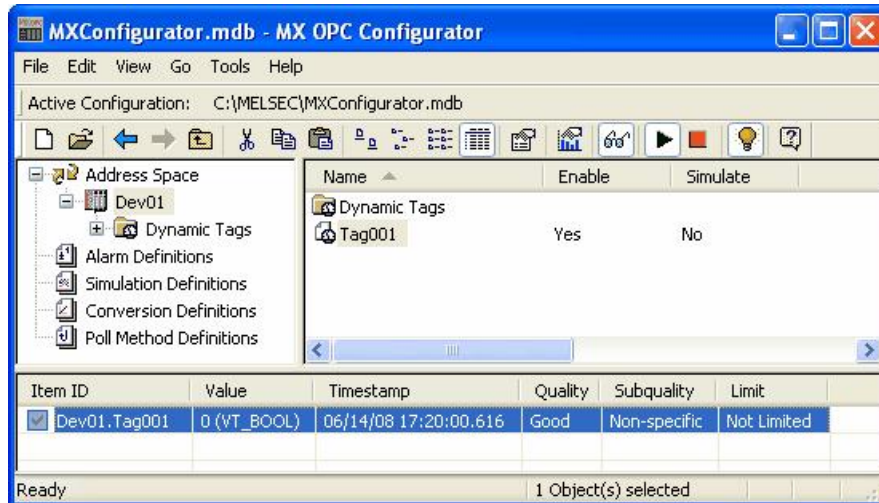
Setting	Tags shown
Off	None - the 'Device statistics' branch will not appear.
Normal	<ul style="list-style-type: none"> <li>• <b>Connected:</b> Returns a number indicating whether the runtime is connected to the device or not. 1 means connected, 0 not connected.</li> <li>• <b>Receives:</b> Number of receives. The same as the statistical counter "Receives" in the statistics dialog.</li> <li>• <b>Transmits:</b> Number of transmits. The same as the statistical counter "Transmits" in the statistics dialog.</li> </ul>
Extended	<ul style="list-style-type: none"> <li>• <b>Connected:</b> Returns a number indicating whether the runtime is connected to the device or not. 1 means connected, 0 not connected.</li> <li>• <b>Receives:</b> Number of receives. The same as the statistical counter "Receives" in the statistics dialog.</li> <li>• <b>Transmits:</b> Number of transmits. The same as the statistical counter "Transmits" in the statistics dialog.</li> <li>• <b>DataBlocks:</b> Number of data blocks. The total number of data blocks this device uses. Data blocks are configured automatically by the Runtime module depending on the <a href="#">advanced settings</a><sup>[69]</sup> for the device.</li> <li>• <b>PointCount:</b> Number of points. The total number of tags that the device manages currently.</li> <li>• <b>QueTrans:</b> Queued transactions. The number of transactions queued.</li> <li>• <b>QueReads:</b> Queued reads. The number of reads remaining in the queue.</li> <li>• <b>QueWrites:</b> Queued writes. The number of writes remaining in the queue.</li> <li>• <b>QueEvents:</b> Queued events. The number of future events scheduled for the queue.</li> </ul>

## 6.5 Monitor View

Monitoring is enabled for each item with a check mark next to it. To enable/disable monitoring for an item, you can click on the box to the left of the item. A check mark inside the box means the item is enabled for monitoring. If there is no check mark, then the item is disabled.

The monitoring view of the Configurator has been optimized for a smoother, better user experience. Monitoring of thousands of tags will not hinder operators from using the Configurator.

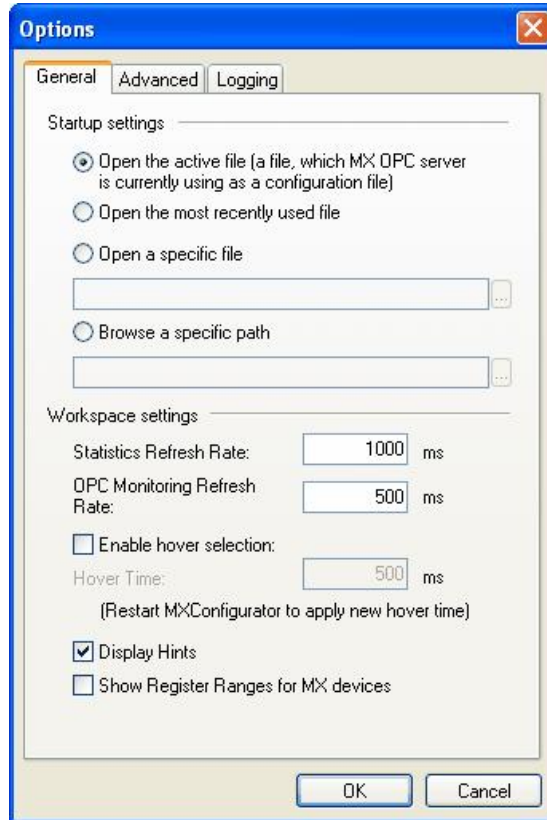
The MX OPC Configurator includes a runtime monitor for viewing server data. To change to the monitor view, select **Monitor View** from the **View** menu. The runtime monitor appears in the bottom pane of the Configurator screen, as shown in the figure below. During runtime, the monitor scans the server and displays the tag values and other statistics such as date, time, and quality.



Runtime Monitor View

### 6.5.1 Setting the Monitor View Refresh Rate

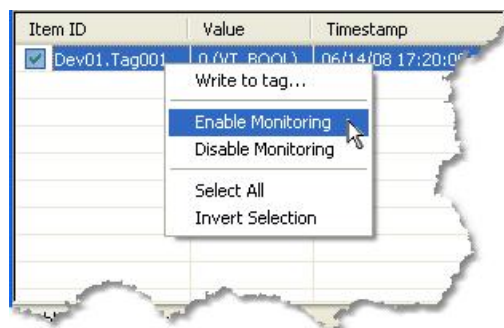
To set the **OPC Monitoring Refresh Rate** (in milliseconds), select **Options** from the **Tools** menu and click on the **General** tab of the **Options** dialog box, as shown in the figure below.



Options Dialog Box: General Tab

### 6.5.2 Enable Monitoring

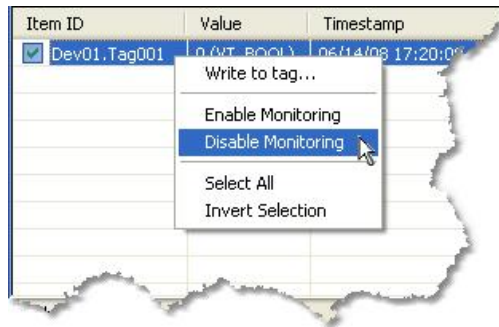
Monitoring is enabled for each item with a check mark next to it. To enable/disable monitoring for an item, you can click on the box to the left of the item. A check mark inside the box means the item is enabled for monitoring. If there is no check mark, then the item is disabled. To enable monitoring for an unchecked item in the monitor view, you can also right-click on the item and select **Enable Monitoring** from the pop-up menu, as shown in the figure below.



Enabling Monitoring for an Item

### 6.5.3 Disable Monitoring

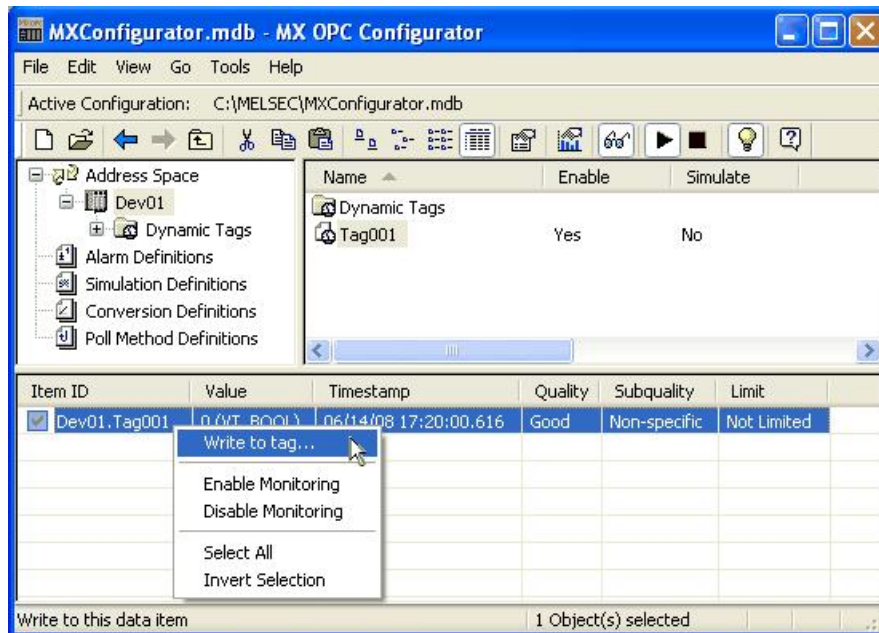
To disable monitoring for a checked item in the monitor view, you can also right-click on the item and select **Disable Monitoring** from the pop-up menu, as shown in the figure below.



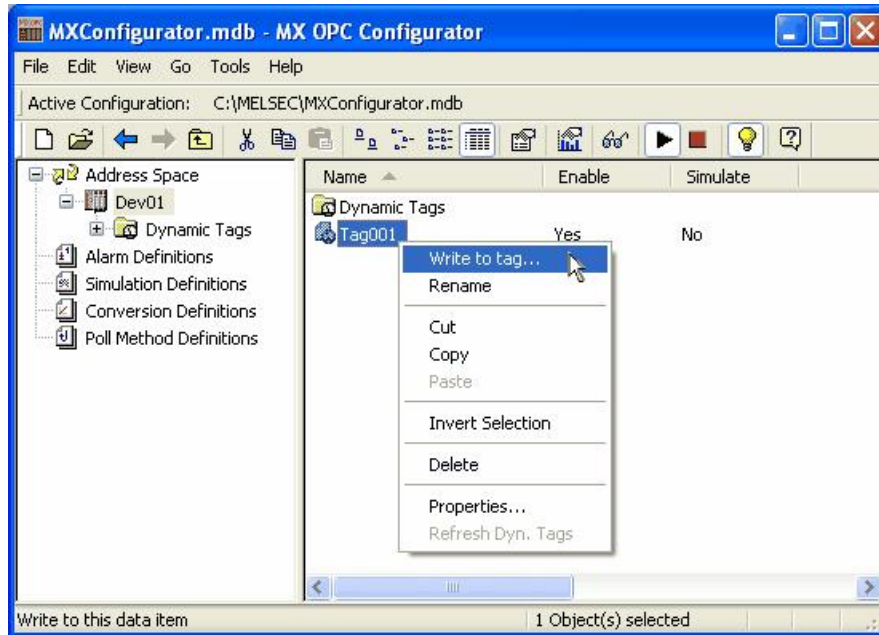
Disabling Monitoring for an Item

## 6.6 Writing to Tags

Right clicking on a tag in the Monitor view or List view will open a context menu offering the possibility to write into the tag:

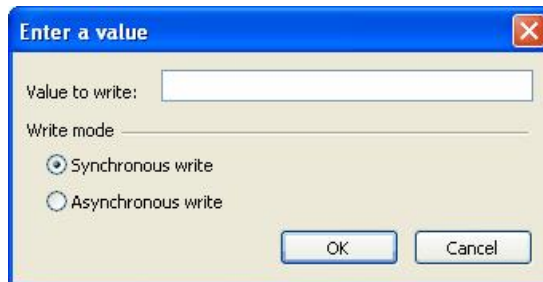


The "Write to tag" menu item is enabled only if exactly one writable item is selected, that is when an MX or Modbus data tag is selected in the Monitor view or the List view.



Clicking the new menu item will open a dialog box, as seen below.

The values accepted depend on the value type of the tag, that the value is written into. For all numeric data types, the value of the number is checked if it is in range of the tag's data type. The radix character for real numbers is a dot, regardless of the current locale.





To write data to arrays, simply separate the elements by commas or semicolons, e.g. writing to an array of floats: 0.1, 0.2, 0.3 or 0.1; 0.2; 0.3. When a semicolon is found in the input value, the configurator assumes that values are separated by semicolons; otherwise it assumes that commas are used as a separator.

Writing to string tags is simple; write the string directly to the edit box. The length of the string is not constrained.

When any of the previous formatting requirements are not met, a dialog box appears showing the error, as in the example below.



## 6.7 Special item names

As well as allowing defined tags to be read, the MX OPC Runtime module can read other details using special item names.

### 6.7.1 Accessing array items

If your client application does not support reading single array elements, you may still be able to read them by changing the name of the item that you add.

A client can address a single element of an array tag using the item syntax:

'arraytag[elemNr]'

e.g. arraytag[2] for element 2 of arraytag.

If the item is of "Array" type, single items representing the array elements will be also created. These items cannot be browsed (they are hidden for browsing), but can be monitored (added to OPC client).

Let's say we have an array item:

TagArray

array of WORDS

Number of Elements: 20

LowerBound: 10.

Then, in the server, aside from the 'TagArray' item that is visible, will also exist the hidden items 'TagArray[10]', 'TagArray[11]', ..., 'TagArray[29]'. The tags can be added from an OPC client, if the OPC client knows about its existence (other than via browsing).

### 6.7.2 Accessing bits

If your client application does not support reading a single bit within a word, you may still be able to use this feature by changing the name of the item that you add.

To read a single bit within a larger type (such as INT, WORD etc.), the easiest way is to define a new tag for the item in MX OPC Configurator. For instance, if you set the address of a new tag as 'D0.2' and the type as BOOL, the tag will read/write bit 2 from address D0. The bits are numbered starting from 0 for the least significant bit. Note that these bit numbers are always in decimal - this is different from GX IEC Developer and GX Developer, which both use hexadecimal bit numbering. For example, GX IEC Developer address D123.B would be entered as D123.11 in MX OPC Configurator.

Alternatively, most clients will allow you to read bits from other types using dot notation, with the syntax:

```
"tagname.bitNr"
```

e.g. tagname.0 for bit 0 of tagname.

With this "Bit access" feature, if the item is of a type other than BOOL, STRING or REAL, you can access its single bits. They are implemented as custom "OPC Properties", and also can be accessed that way. However, it is not possible to write bit values this way as the properties are read-only. If you need to write to the bit, define a new tag for it (as described above).

Examples:

```
TagD0, type WORD
```

You can access 'TagD0', but also its properties:

```
'TagD0.0', 'TagD0.0', ..., 'TagD0.15'
```

## 6.8 Dynamic Tag Support

The MX OPC Configurator has a special folder "Dynamic Tags" in each MX Device where these dynamic tags show up. This folder cannot be modified and is present all the time, even if no dynamic tags are present.

Dynamic tags are like normal tags, but as the name suggests, they are added dynamically and can be used without having to first define them in the configurator database. When an OPC client application requests a dynamic tag, it will be added temporarily and removed again when there are no longer any OPC client applications using it. This can be useful when (for example) setting up a display screen in a SCADA package which the user will only view occasionally. By using dynamic tags to read the values on the screen, the tags will only be created and polled while the screen is in use. When the screen is not in use, there is no additional load on the system.

Dynamic tags have a fixed name format, which MX OPC Server uses to determine the device, address and data type of each tag. Each tag has a name such as:

```
<Device>[\<Group>]\Item
```

The backslash ('\') separator is used to identify dynamic tags – parts of other item IDs are separated by period (.) characters. If you are writing your own client application, this means that when programmatically browsing the OPC server hierarchy for dynamic tags (for example, in Visual Basic – See "[Using OPC from Visual Basic](#)"<sup>[36]</sup>), the item path and item ID can be different, for example, the path "Dev01.Dynamic Tags.D0.I.4" would have item ID "Dev01\D0.I.4".

The '<Device>' part must match the name of an MX device which is already configured in the database; devices cannot be created dynamically.

The '<Group>' section is optional, and works in a similar way to the normal OPC server groups, although only one level is allowed for dynamic tags.

The '<Item>' part can be in one of two formats, either:

```
<Register><StartAddr>[.<bit>[.<Type>[.<ArraySize>]]]
```

which specifies "bit-within-word" addressing to read starting from the bit number specified (decimal format), similar to addressing bits within a tag using the 'Tag.1' syntax, or the alternative format:

```
<Register><StartAddr>[.<Type>[.<ArraySize>]]
```

The <Register> part gives the device register within the address, e.g. 'D' for 'D0', 'X' for 'X1A', 'Y' for 'Y5' etc. For devices that contain a backslash ('\') e.g. U3E1\G, the backslash should be removed from the device name to prevent it being used as an item name separator - e.g. specify 'U3E2G' instead of 'U3E2\G'.

The <StartAddr> part gives the numeric element of the address, e.g. '0' for 'D0', '1A' for 'X1A', etc. Depending on the PLC type and register selected, this may be in decimal, hexadecimal or octal – this will match the address entry in the configurator.

The <Type> part is optional, and specifies the data type of the tag (see list below). If this is not specified, a suitable default type will be used based on the device register.

The <ArraySize> part is used to define array tags, and specifies the number of elements in the array (except for strings where it gives the length instead – see note below). The <Type> part must always be supplied as well when an array size is used.

Possible values for the <Type> part and their abbreviations are:

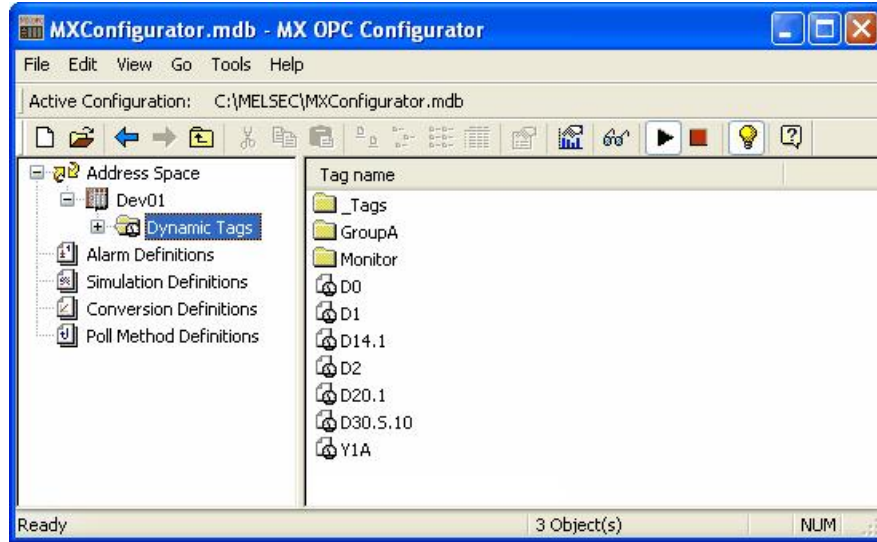
<Type> Part	Abbreviation
Bool	B
Word	W
Dword	DW
Int	I
Dint	DI
Real	R
String	S

For strings, the <ArraySize> must be supplied and gives the string length in characters.

Some example item names for dynamic tags using this format are shown below, for a device called 'PLC'.

PLC\D20	Address D20, default type
PLC\D20.W	Address D20, read as word type
PLC\D20.1	Bit 1 of address D20 (will be Boolean)
PLC\GroupA\D40	Address D40, within a group called 'GroupA'
PLC\D41.Dint	DINT at address D41
PLC\X1A	Address X1A, in a register type that uses hexadecimal addressing on this PLC
PLC\D50.I.10	Array[10] of int
PLC\D100.S.10	String of length 10
PLC\M10.B.5	Array [5] of bool at M10
PLC\D100.12.B.8	Array [8] of bool at D100.12
PLC\U3E1G10000.I	Shared word device in CPU 2 as integer

The "Dynamic Tags" folder can hold tags directly or the tags can be grouped, depending on the way the client created the tags. Only one level of groups is allowed and dynamic tag group names have the same restrictions as normal groups, i.e. the group name must contain only standard letters, numbers, underscores, hyphens and parentheses, and the name must start with a letter, hyphen or underscore. The maximum name length is 32 characters.



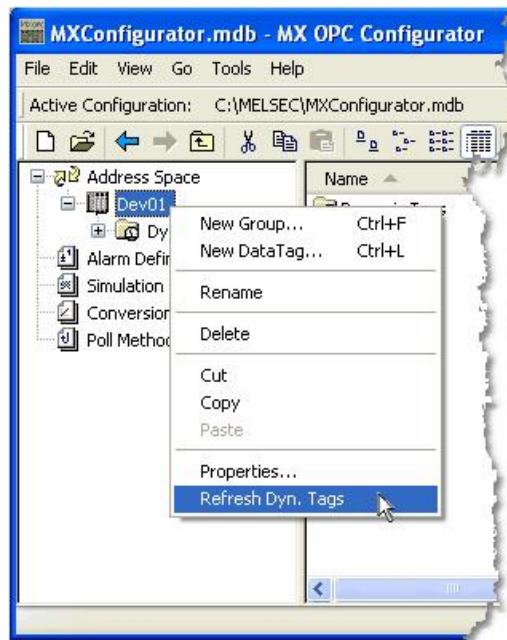
Example.mdb dialog

Dynamic tags will also show in the Monitor view and the Statistics dialog, the Monitor view can be used to write into existing dynamic tags.

The dynamic tags for a device are read from the server when the user clicks on the Dynamic Tags folder for the first time. Also all tags that are in a group (like GroupA) are read in when the user clicks on that group for the first time.

### 6.8.1 Refresh Dynamic Tags

To re-read (or refresh) tags in a particular group, or device, simply right click on that item and choose the new item: **Refresh Dyn. Tags**.



This new item is available for the Address Space, MX Devices and the Dynamic Tags folder in each device, and for individual dynamic tag groups. Refreshing each of the items has the following results:

**Address space** - Refreshes all the tags and all groups in all MX Devices, which were previously read in at least once, that means the user clicked on them at least once in the past.

**MX Device** - Refreshes all tags and groups of this device, which were previously read at least once.

The **Dynamic Tags** folder - Same effect as refreshing the parent MX Device

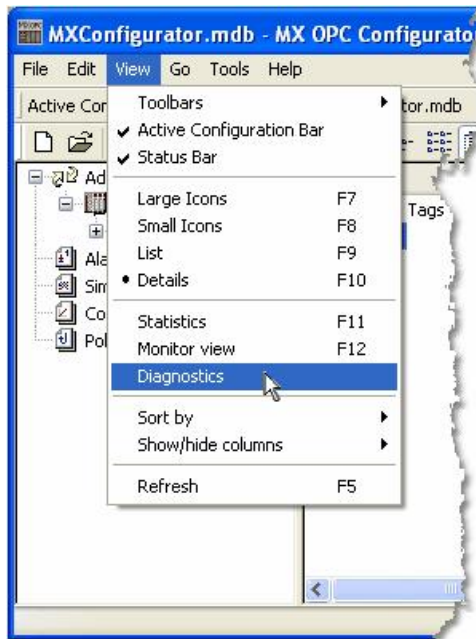
A **Dynamic Group** - Refreshes all tags in this group.

## 6.9 Automatic Backup

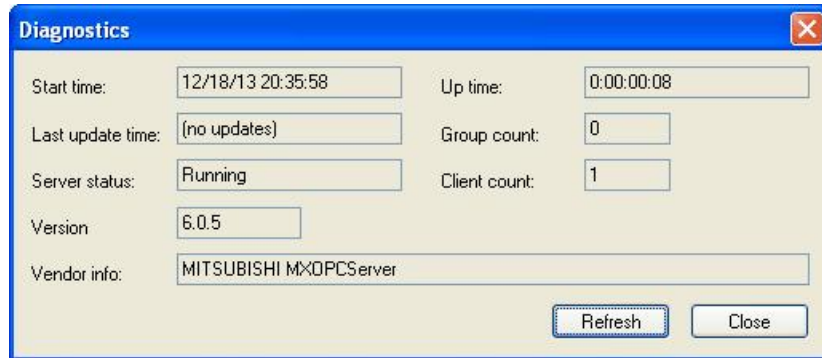
Each time MX OPC Configurator starts up, it now makes a backup copy of the opening database to the same folder where the original resides. The name of the backed-up database is the original database name but with the extension "bak" added. The resulting filename will be for example "MXConfigurator.mdb.bak".

### 6.10 Diagnostics

As shown in the screenshot below, MX OPC Configurator includes an item in the main menu called Diagnostics:



Selecting this item will show a window with diagnostic information from the connected OPC server. You can still use other windows normally while this information is shown.



**Diagnostic information**

- **Start time:** The time the server started in format mm/dd/yy hours:minutes:seconds
- **Up time:** The time the server is running, formatted as days:hours:minutes:seconds
- **Last update time:** The time the server sent the last data value update to this client. This value is maintained on an instance basis. Formatted in the same way as start time.
- **Group count:** The total number of groups being managed by the server.
- **Server status:** Possible server states include Running, Failed, and No configuration
- **Client count:** The connected client count
- **Version:** The version of the server
- **Vendor info:** The vendor info as reported by the server
- **Refresh:** Reload this dialog data from the server
- **Close:** Closes the window

The runtime will also support various diagnostics on various abstraction levels and log these diagnostics using TraceWorX into an XML file. The Configurator will not show or parse these XML files in version 6.05. These options are described in "[Options - Logging Tab](#)"<sup>[60]</sup>.

## 6.11 Showing Register Ranges

The new server runtime can provide the list of possible register ranges of each MX device. This list is provided to the OPC clients as a special group called "Hints" in the address space of the device containing tags.

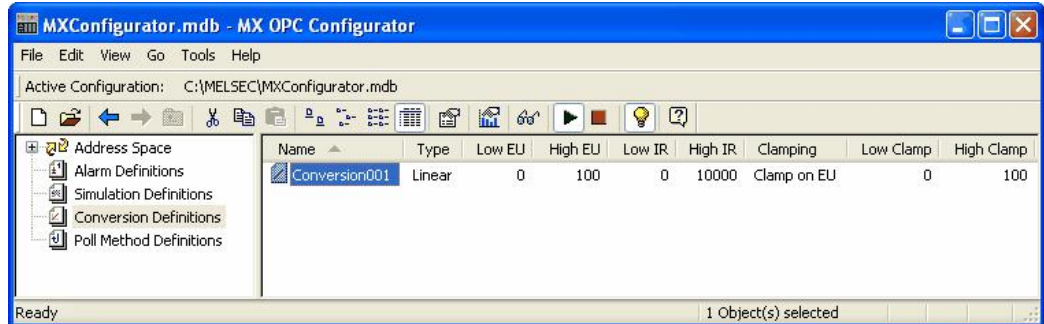
The names of these tags are in the form <register><starting address>-<ending address>, where all three items are in the same form as seen in the **Supported Devices** dialog (clicking Browse from Tag Properties Basic tab). D0.0-7999.15 is a valid example for the FX3U(C) CPU type.

This function can be enabled or disabled through the [Options](#)<sup>[56]</sup> dialog. It is switched off by default.



## 7 Configuring Conversion Definitions

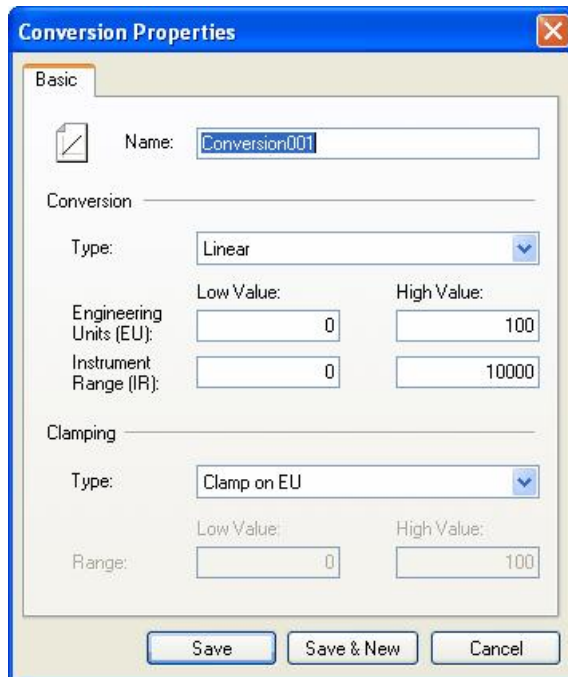
The **Conversion Definitions** tree control, shown in the figure below, allows you to configure conversion settings for data tags. Conversion settings can be associated with a data tag via the tag properties dialog box.



Conversion Definitions Tree Control

### 7.1 Basic Conversion Properties

In the **Basic** tab of the **Conversion Properties** dialog box, shown below, configure the following settings:



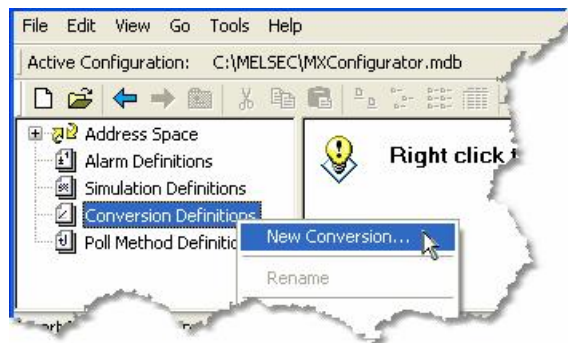
Conversion Properties: Basic Tab

- **Name:** Specifies the name of the conversion definition. The name can be up to 50 alphanumeric characters, including underscores ( \_ ) and hyphens ( - ).
- **Conversion Type:** For each conversion definition, select one of the following conversion types from the drop-down list:
  - **None (make float):** Converts the data into float data type, but does not change the value itself. When this option is selected, the **Engineering Units** and **Instrument Range** fields are disabled.
  - **Linear:** Keeps a linear relation between EU and IR.
  - **Square Root:** Keeps a square root relation between EU and IR.
- **Engineering Units (EU):** Client scale; specify low and high values for the engineering units (if applicable).
- **Instrument Range (IR):** Device scale; specify low and high values for the instrument range (if applicable).
- **Clamping:** If clamping is enabled, the data value will be limited to its High Value/EU High Value when it exceeds the upper limit, and similarly to its Low Value/Low EU Value parameter when it exceeds the bottom limit. Select one of the following clamping types from the drop-down list:
  - **None:** No clamping type is specified.
  - **Clamp on EU:** Clamps on the specified low and high engineering units (EU) values.
  - **As Specified:** Clamps on a specified range of low and high values.
- **Range:** Specify low and high values for the range.
- **Save:** Saves all changes specified in the properties dialog box. The conversion definition appears in the list view.
- **Save & New:** Saves all changes specified in the properties dialog box.
- **Cancel:** Closes the properties dialog box.

## 7.2 Creating a New Conversion Definition

To create a new conversion definition:

1. Right-click the **Conversion Definitions** tree control and select **New Conversion** from the pop-up menu, as shown in the figure below.



**Creating a New Conversion Definition**

- The properties dialog box for the new configuration appears, as shown in the figure below.

The screenshot shows the 'Conversion Properties' dialog box with the following settings:

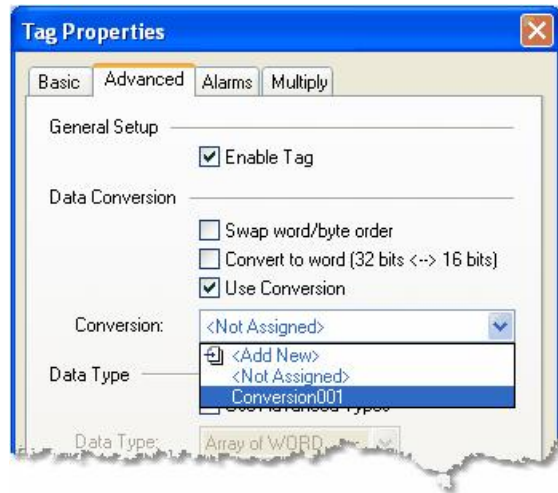
- Name:** Conversion001
- Conversion Type:** Linear
- Engineering Units (EU):** Low Value: 0, High Value: 100
- Instrument Range (IR):** Low Value: 0, High Value: 10000
- Clamping Type:** Clamp on EU
- Range:** Low Value: 0, High Value: 100

#### Setting the Properties for the New Conversion Definition

- In the **Name** field, type a name for the new conversion definition.
- Choose a **Conversion Type** from the drop-down list and specify low and high values for engineering units and instrument range (if applicable).
- Choose a **Clamping Type** from the drop-down list and specify low and high values for the range.
- When you have finished configuring the conversion definition properties, click the **Save** button. The new configuration appears under the **Conversion Definitions** tree control.

## 7.3 Assigning Conversion Definitions to Tags

Conversion settings can be associated with a data tag via the **Advanced** tab of the **Tag Properties** dialog box, as shown in the figure below.



**Data Tag Properties: Advanced Tab**

To get the data value converted to another form, choose one of the predefined or user-defined conversions. MX OPC Server includes the following conversions:

### 1. Swap word/byte order

- WORD conversion B1/B2 <-> B2/B1
- DWORD conversion B1/B2 B3/B4 <-> B4/B3 B2/B1

### 2. Convert to Word (32bits <-> 16bits)

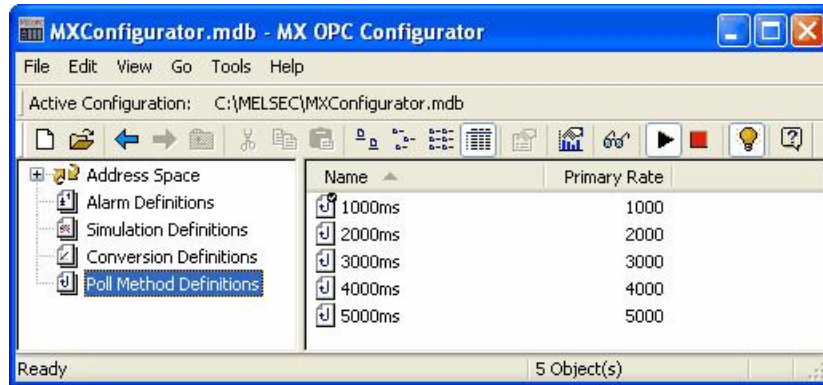
If an OPC client requests a VT\_I4 (DINT), where the tag is defined in the MX OPC Configurator as a VT\_I2 (INT), the MX OPC Server will reduce it to VT\_I2 when writing and expand it to VT\_I4 when reading. The sign of the data will be taken into account.

### 3. Use Conversion

To select a conversion definition, check the **Use Conversion** check box and then select a conversion definition from the drop-down list, which lists all conversion definitions configured in the **Conversion Definitions** tree control of the Configurator.

## 8 Configuring Poll Method Definitions

The **Poll Method Definitions** tree control, shown in the figure below, allows you to configure polling method settings for data tags. Polling methods can be associated with data tags via the tag properties dialog box. The configurator provides several preconfigured polling methods.

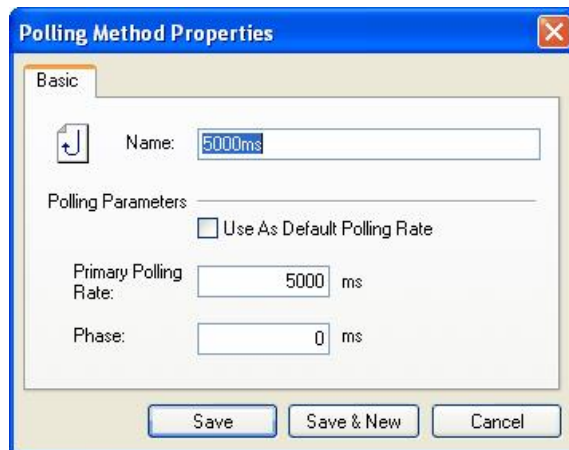


**Poll Method Definitions Tree Control**

## 8.1 Basic Poll Method Properties

In the **Basic** tab of the **Poll Method Properties** dialog box, shown below, configure the following settings:

- **Name:** Specifies the name of the poll method definition. The name can be up to 50 alphanumeric characters, including underscores ( \_ ) and hyphens ( - ).
- **Use As Default Polling Rate:** You can designate one poll method definition as the default polling rate for the configuration database. The default poll method definition appears with a check mark icon in the list view.
- **Primary Polling Rate:** The driver polls the data at the specified primary polling rate (in milliseconds). For example, if you enter 2,000 milliseconds (2 seconds) in the **Primary Polling Rate** field, the driver polls data every 2 seconds.
- **Phase:** The Phase setting can be used to spread the load on the server when reading a large number of tags, by adding an additional delay after the polling time before the tags are read. As an example, if two poll methods are defined as 'Polling rate 10000, phase 10000' and 'Polling rate 10000, phase 5000', these will be read at the same interval, but at different times within the 10 second polling period.
- **Save:** Saves all changes specified in the properties dialog box. The poll method definition appears in the list view.
- **Save & New:** Saves all changes specified in the properties dialog box.
- **Cancel:** Closes the properties dialog box.

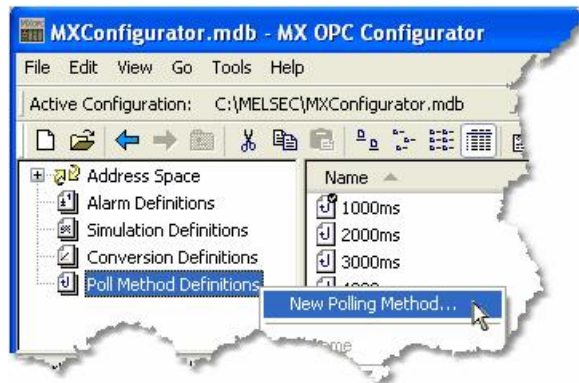


**Polling Method Properties: Basic Tab**

## 8.2 Creating a New Poll Method Definition

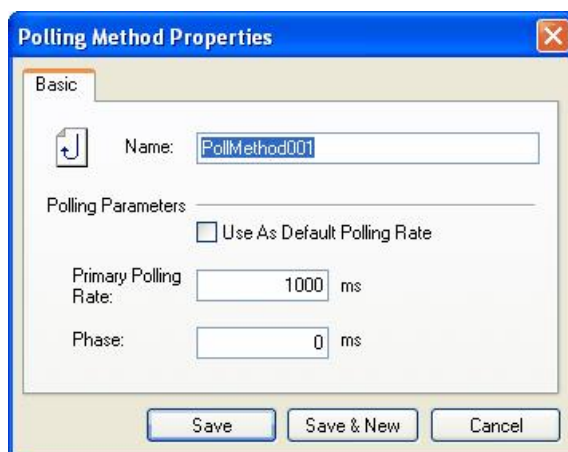
To create a new poll method definition:

1. Right-click the **Poll Method Definitions** tree control and select **New Polling Method** from the pop-up menu, as shown in the figure below.



**Creating a New Poll Method Definition**

2. The properties dialog box for the new configuration appears, as shown in the figure below.

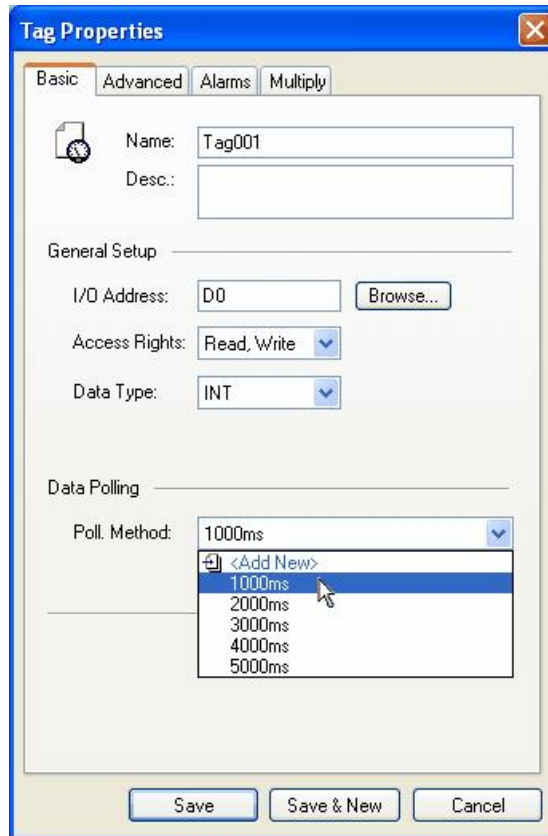


**Setting the Properties for the New Polling Method**

3. In the **Name** field, type a name for the new polling method.
4. Specify a **Primary Polling Rate** (in milliseconds). The driver polls the data at the specified primary polling rate. For example, if you enter 2,000 milliseconds (2 seconds) in the **Primary Polling Rate** field, the driver polls data every 2 seconds.
5. Check the **Enable Primary Rate** check box to activate the polling method.
6. When you have finished configuring the poll method definition properties, click the Save button. The new configuration appears under the **Poll Method Definitions** tree control.

### 8.3 Assigning Poll Method Definitions to Tags

Polling methods can be associated with data tags via the **Basic** tab of the **Tag Properties** dialog box, as shown in the figure below. Select a polling method from the **Polling Method** drop-down list, which lists all polling methods configured in the **Polling Method Definitions** tree control of the Configurator.



Data Tag Properties: Basic Tab



## 9 Using MXChange

MXChange is a concept designed to integrate Mitsubishi's existing automation development tools into a single environment which harmonises programming, supervision, network configuration and simulation components. It is intended to provide an *Integrated Development Environment (IDE)* which integrates existing programming and configuration tools. MXChange provides a way for the configuration packages to co-operate, allowing the definition of I/O references in several programs at the same time. If an I/O address is changed in one package, the change can be propagated to the other packages automatically, reducing the potential for error.

Each MX device in MX OPC Configurator represents a connection to a PLC. Each PLC will have been programmed with a development tool such as GX IEC Developer or GX Developer. When an MX OPC Configurator project is connected to MXChange, each MX device is linked to the corresponding PLC project in the MXChange database. When new tags are added in GX IEC Developer and marked for export to MXChange, they will be automatically imported into MX OPC Configurator with the correct details such as I/O address, tag name and description. Other applications (such as E-Designer) can also share the tag information. When the details of a shared tag are modified (e.g. by changing the I/O address), the new address is sent to all applications using the tag, reducing the risk of configuration errors such as two applications using different addresses for the same tag.

By default, a single instance of MXChange will be installed as a service when MX OPC Configurator is installed.

**Note:** Modbus devices are not imported or exported within MXChange, as there are no other development applications which use Modbus devices.

### 9.1 Connecting to MXChange

MX OPC Configurator can be connected to MXChange either by creating a new MXChange database, or by opening an existing MXChange database. Once you have connected an MX OPC Configurator project to MXChange, it will attempt to reconnect to the MXChange database automatically each time the project file is reopened. If the stored database details do not match the current MXChange database connection, you will be prompted to reconnect to the server and database that were originally used for the opened project, as shown below. Select **Yes** to connect to the previous server (recommended) or **No** to switch to the currently connected server.



Reconnect with previous details

You can tell whether or not there is a connection to MXChange, and if so to which database, from the status indicator shown at the right of the active configuration bar. The connection status is highlighted in the example picture below, where there is a connection to a database called *MXCDB* on a server called *LOCALPC*.



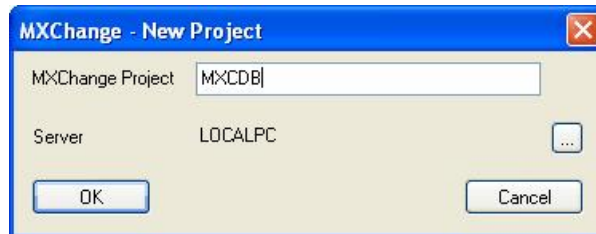
MXChange connection status

If you create a new MX OPC Configurator database while connected to MXChange, it will be linked to the MXChange database automatically.

### 9.1.1 Connecting to a new MXChange database

The steps to create a new MXChange database and connect the current project to it are:

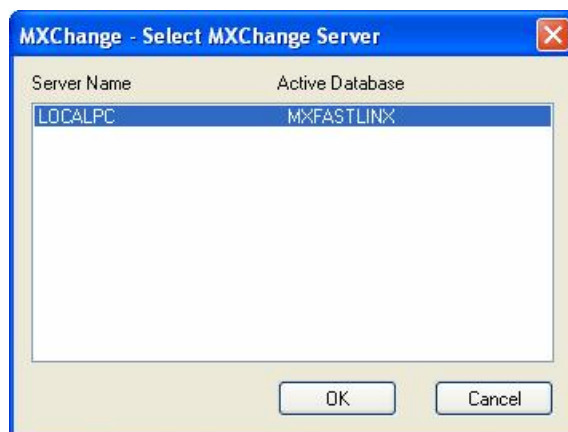
1. From the **File** menu, select **MXChange database actions -> New**. The **New database** dialog will appear as shown below.



**New MXChange database**

Enter the name of the new MXChange database in the **MXChange Project** field.

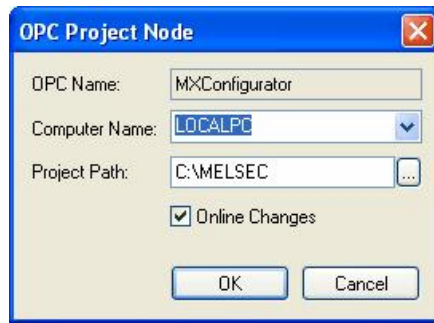
To select a different server, click the ... button to show the server browse dialog:



**Browse for server**

2. On the **MXChange Connection** dialog box, click **OK**. The configurator tries to connect to the MXChange Server and create the new database.

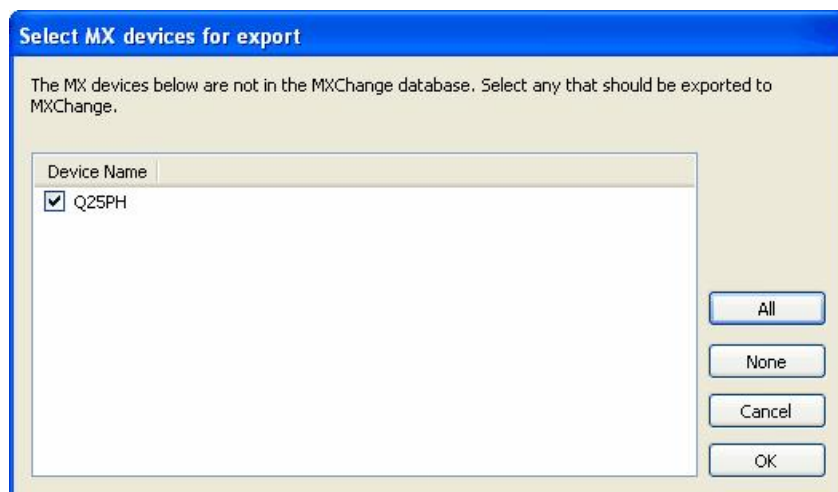
3. The **OPC Project Node** dialog box appears, as shown in the figure below.



**OPC Project Details**

Specify the following properties for the node, and then click **OK**:

- **OPC Name:** Same as the MX OPC Configurator database name (with no .mdb file name extension).
  - **Computer Name:** Name of the computer where the project is stored. You can select a node from the drop-down list.
  - **Project Path:** Folder where the project is stored. Click the ... button to browse for a folder.
  - **Online Changes:** Allows you to see changes that other projects have made to the shared tags as soon as they happen. If you check the check box then you will see a [notification dialog](#)<sup>[186]</sup> when something is changed in the MXChange database on the MXChange Server (i.e. when someone changes, deletes, or adds an item in the MXChange database). If this option is switched off, you will not be notified of changes, although you can still use the [Show changes](#)<sup>[186]</sup> option to check for changes, and they will still be processed when the database is [synchronised](#)<sup>[186]</sup>.
4. If there are already MX devices defined in the MX OPC Configurator project, you will be given the option to export them to the MXChange project:



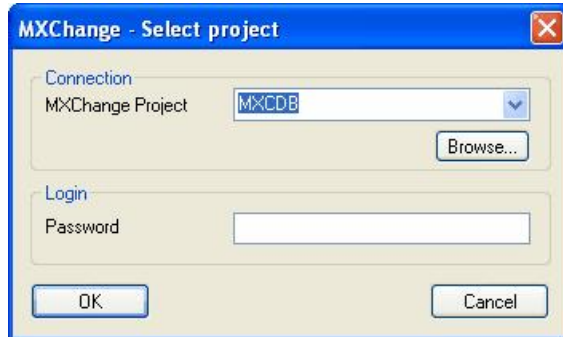
**Select devices for export to MXChange**

For more information about exporting existing devices to MXChange, see [Exporting to MXChange](#)<sup>[182]</sup>.

### 9.1.2 Connecting to an existing MXChange database

To connect the current MX OPC Configurator project to an existing database, the steps to take are:

1. From the **File** menu, select **MXChange database actions -> Open**. The **Open database** dialog will appear as shown below.

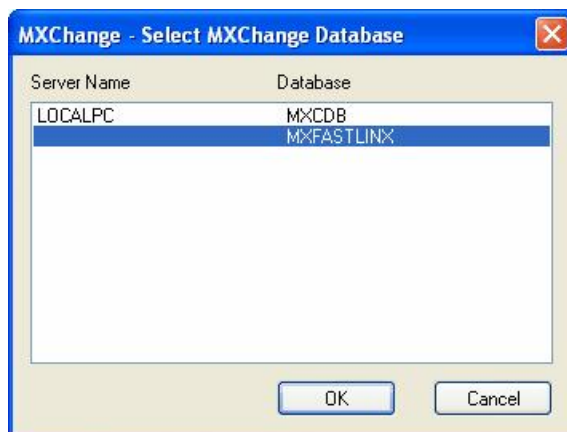


**MXChange open database dialog**

The controls on this dialog are:

- **MXChange project** - the MXChange database to connect to
- **Browse** - browse the local network for a different MXChange server and database (see below)
- **Password** - if the database requires a password, enter it here
- **OK** - try to connect
- **Cancel** - cancel the connection attempt

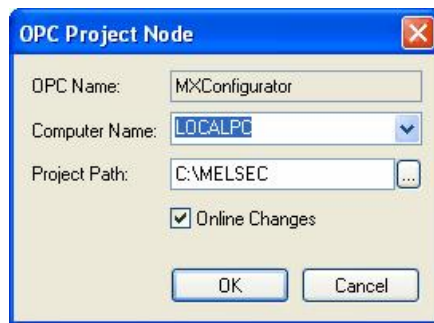
The Browse button will show the dialog below, after a short delay while it searches the network for available MXChange servers. This dialog allows a server and database to be selected. Each server is listed in turn along with its available databases.



**MXChange browse database dialog**

*Note:* Depending on the configuration of your network router and switches, MXChange server browsing may only be able to detect servers on the same subnet. If you know the name of the server to browse / connect to, you can enter a server name or address in the 'MXChange project' field above in the format '*Server:Database*' to connect to it directly. When a server name has been provided in this way before 'Browse' is clicked, the named server will be checked directly - this will often allow browsing of server information across subnets.

2. On the **MXChange Connection** dialog box, click **OK**. The configurator tries to connect to the MXChange Server and login as administrator with the provided password. If the login fails, you will be informed by an error dialog.
3. After connecting to the server, MX OPC Configurator will check the database to see if there is already an MX OPC Configurator project with the same name as the currently edited configurator database. If not, you will be prompted to create one using the **OPC Project Node** dialog box, as shown in the figure below.

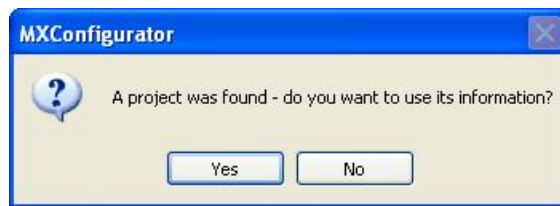


**OPC Project Details**

Specify the following details for the project, and then click **OK**:

- **OPC Name:** Same as the MX OPC Configurator database name (with no .mdb file name extension).
- **Computer Name:** Name of the computer where the project is stored. You can select a node from the drop-down list.
- **Project Path:** Folder where the project is stored. Click the ... button to browse for a folder.
- **Online Changes:** Allows you to see changes that other projects have made to the shared tags as soon as they happen. If you check the check box then you will see a [notification dialog](#)<sup>[186]</sup> when something is changed in the MXChange database on the MXChange Server (i.e. when someone changes, deletes, or adds an item in the MXChange database). If this option is switched off, you will not be notified of changes, although you can still use the [Show changes](#)<sup>[186]</sup> option to check for changes, and they will still be processed when the database is [synchronised](#)<sup>[186]</sup>.

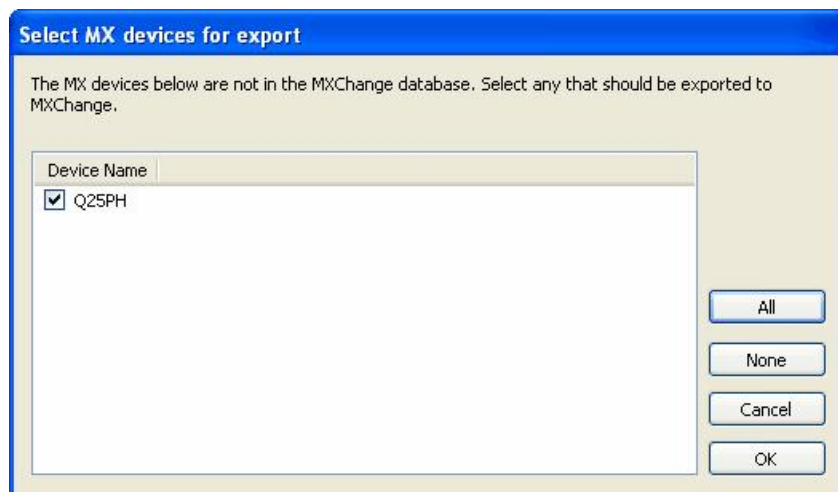
If there was already a matching OPC server project in the database, you will be prompted to reconnect using the same details.



**Reuse prompt**

Select **Yes** to import PLC projects as new MX devices immediately - see [Importing from MXChange](#) [184] for a description of the import procedure.

4. If there are already MX devices defined in the MX OPC Configurator project, you will be given the option to export them to the MXChange project:



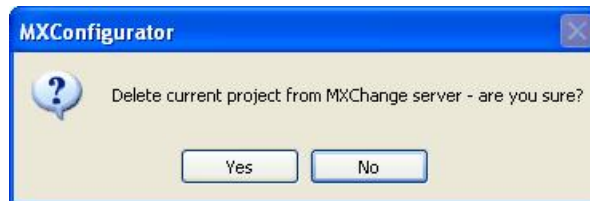
**Select devices for export to MXChange**

For more information about exporting existing devices to MXChange, see [Exporting to MXChange](#) [182].

## 9.2 Detaching from MXChange

When you no longer wish to connect an MX OPC Configurator project to MXChange, select **File -> MXChange database actions -> Close** to disconnect from the database and break the link to MX OPC Configurator. After using this option, MX OPC Configurator will no longer attempt to connect to the database each time the project is loaded. The project will still exist in the MXChange database in case you need to reconnect to it later.

To remove the MX OPC Configurator project from MXChange permanently, select **File -> MXChange project actions -> Delete** to delete the project from the MXChange database. Other application projects in the MXChange database will not be affected. You will be prompted to confirm the deletion as shown below.



Delete project from MXChange prompt

## 9.3 Managing MXChange databases

MX OPC Configurator can be used to manage the current MXChange database, using items on the **MXChange database actions** menu. From MX OPC Configurator, you can:

- [Change the database password](#)<sup>[180]</sup>
- [Compact the database](#)<sup>[180]</sup>
- [Save the database with a new name](#)<sup>[181]</sup>
- [Delete the MXChange database and all the projects within it](#)<sup>[181]</sup>

### 9.3.1 Changing the MXChange password

To change the administrator connection password for the connected MXChange Server:

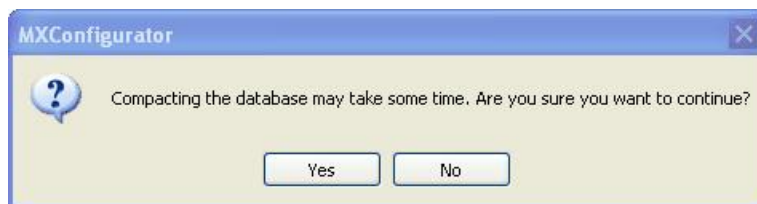
1. Select **MXChange Actions > Change Password** from the **File** menu. (**Note:** You must be connected to the server before you can change the password.)
2. This opens the **Change MXChange Password** dialog box, as shown in the figure below. Enter the **Old Password**, and then type the **New Password** twice to confirm. Click **OK**.



Changing the Password for the MXChange Server

### 9.3.2 Compacting the MXChange database

To compact the MXChange database, use the menu item **File -> MXChange database actions -> Compact**. This reduces the size of the MXChange database by removing any unused empty space that has accumulated over time. At the prompt below, select **Yes** to continue.

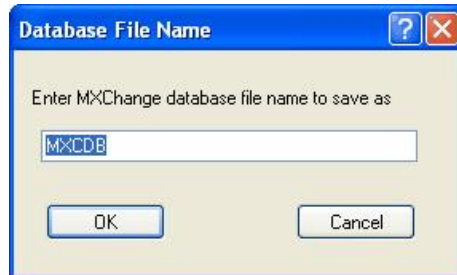


Compact Database



### 9.3.3 Save the MXChange database with a new name

To save the MXChange database with a new name, use the menu item **File -> MXChange database actions -> Save as**. You will be prompted to enter the new database name as shown in the figure below.



**Save database as**

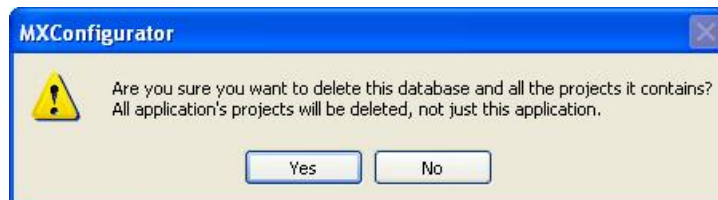
If you intend to use the new database with other MXChange client applications (such as GX IEC Developer and E-Designer), you will need to switch to the new database in those applications as well. Otherwise, they may use their own stored connection details and continue to connect to the old database.

### 9.3.4 Deleting the MXChange database

To delete the MXChange database and *ALL* the projects it contains (even those from other applications), use the menu item **File -> MXChange database actions -> Delete**.

**WARNING:** This destroys *ALL* application data in the MXChange database, not just the MX OPC Server project data. Do not use this option unless you intend to destroy all data in the MXChange server database for *ALL* applications. If you just want to delete the current project from MXChange, use the menu item **File -> MXChange project actions -> Delete** instead.

As this action is dangerous and potentially very destructive, you will be prompted twice to confirm it, with the message shown below. Answer **No** at either prompt to cancel the operation.



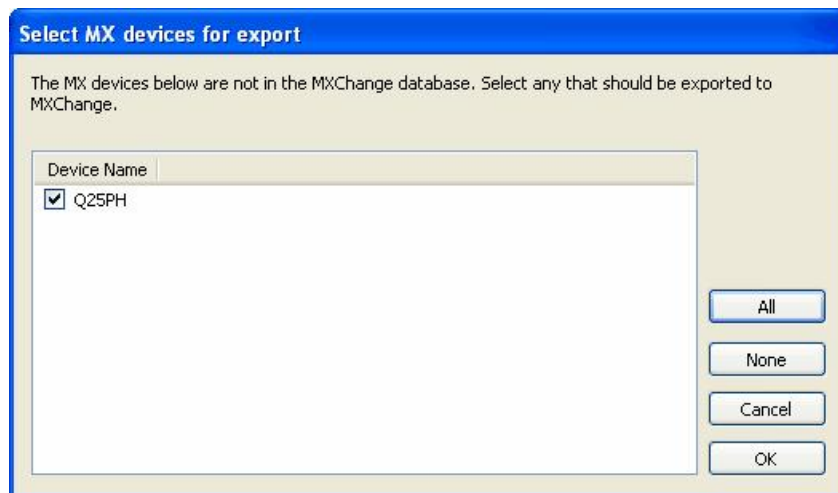
**Delete database prompt**

If any other application still has an open connection to the MXChange database, the deletion will fail with an error.

## 9.4 Exporting to MXChange

When a MX OPC Configurator database is linked to MXChange, each of the MX devices it contains can be linked to a GX IEC Developer or GX Developer project in the MXChange database. The most common way to do this is to start by creating the GX IEC Developer project and variables and then export the details from GX IEC Developer to a new device in MX OPC Configurator. However, if you have already defined a device and tags in MX OPC Configurator, it is still possible to export these to a new GX IEC developer project in MXChange using the **Export devices to GID** option:

1. From the **File** menu, select **MXChange Project Actions -> Export devices to GID**.
2. You will be prompted to select the devices to export to MXChange, as shown in the picture below. Devices which have already been linked to MXChange will not be shown in the list.



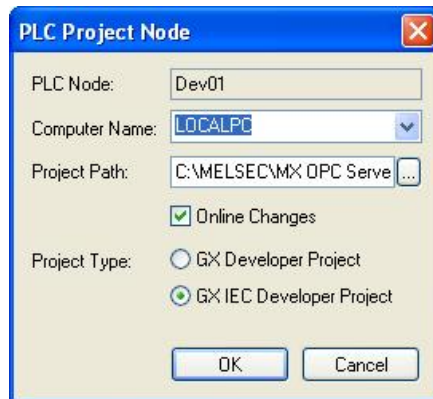
**Select devices for export to MXChange**

To select a device for export, click on the box next to the device name so that it appears checked.

The controls on this dialog are:

- **All** - select all available devices
- **None** - deselect all devices
- **Cancel** - close the dialog without exporting to MXChange
- **OK** - export the selected devices to MXChange

3. If the PLC project does not already exist, the **PLC Project Node** dialog box appears as shown in the figure below. Specify the following properties for the node, and then click **OK** to create the new PLC project.



**PLC Project Node Properties**

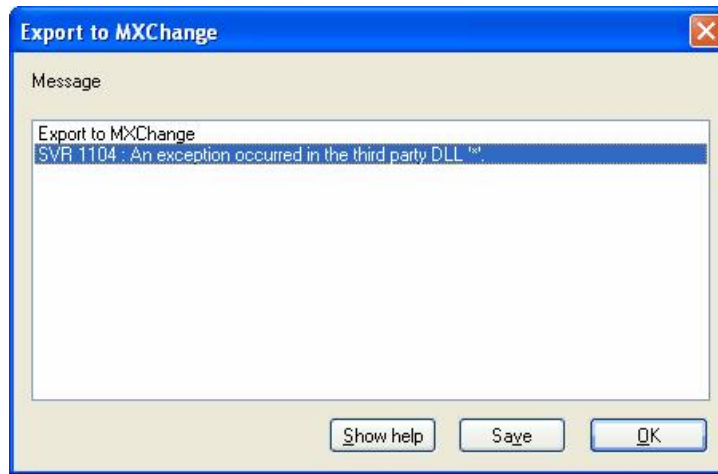
The controls on this dialog are:

- **PLC Node:** Name of the PLC project node in the MXChange database. This always matches the device name.
  - **Computer Name:** Name of the computer where the PLC project file is stored. Either type in the computer name, or select a name from the drop-down list.
  - **Project Path:** Folder where the PLC project file is stored. Click the ... button to browse for a folder (on the local computer only).
  - **Online Changes:** If this box is checked, you will see a notification dialog when something is changed in the MXChange database on the MXChange Server (i.e. when someone changes, deletes, or adds a node in the MXChange database).
  - **Project Type:** Specifies which application will be used to edit the PLC project. There are two applications normally used for creating PLC projects: **GX Developer** and **GX IEC Developer**.
4. After clicking **OK** on the previous dialog, the export will begin. A new project will be created in the MXChange server, and the MX OPC Configurator tags on the device will be exported to it. If everything goes well, the following message box will appear:



**Export complete without errors**

5. If the export is unsuccessful, then you will see a list of problems encountered during export similar to the one shown in the figure below. To save the results, click the **Save** button and specify a name for the file. To show help on an MXChange error code (such as SVR 1104 in the example below), click on the line in the results then click **Show help**.



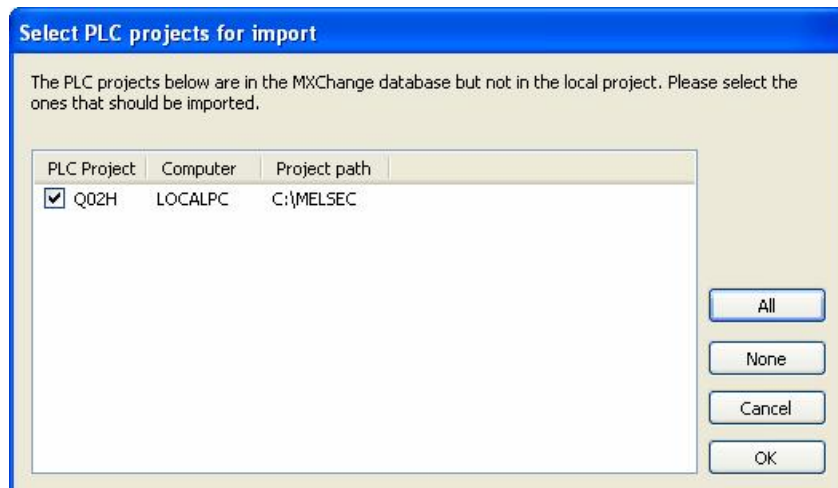
Export Unsuccessful

## 9.5 Importing from MXChange

Once a connection to the MXChange Server has been established, you can also import the data from GX IEC Developer or GX Developer projects into your configuration database from the MXChange Server. When you import data from MXChange, each PLC project on the MXChange Server becomes a new device in the MX OPC Configurator project.

To import data from the server:

1. From the **File** menu, select **MXChange Project Actions -> Import GID projects**.
2. You will be shown a list of GX IEC Developer and GX Developer projects in the database which have not already been linked to the current MX OPC Configurator project, as in the figure below.



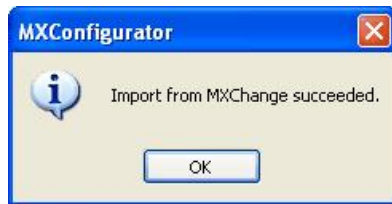
Select PLC projects to import

To select a device for import, click on the box next to the device name so that it appears checked.

*Note:* Since the project will be created as a new device in MX OPC Configurator, you will only be able to select PLC projects with names that would be valid as device names. Device names are limited to 32 characters and cannot contain spaces. To import a PLC project with a name that would be invalid as a device, rename the project before attempting to import it.

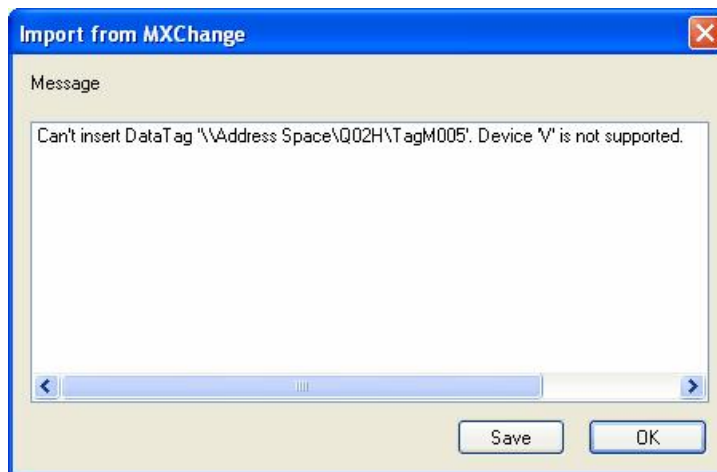
The controls on this dialog are:

- **All** - select all available devices
  - **None** - deselect all devices
  - **Cancel** - close the dialog without importing from MXChange
  - **OK** - import the selected devices from MXChange
3. Each time you import a device, a new [Communication Setting Wizard](#)<sup>[95]</sup> appears to allow the connection details to be set up. Select the link to the PLC (for example, Ethernet connection or serial connection) in the same way as for a new device.
  4. After the devices are created, the tags are imported into the MX OPC Configurator database. On successful completion, the message below will be shown:



**Import Successful**

5. If the import is unsuccessful, then you will see a list of problems (e.g. unsupported register types for the specified device, tag address out of valid range, etc.) encountered during import similar to the one shown in the figure below. To save the results, click the **Save** button and specify a name for the file.



**Import Unsuccessful**

## 9.6 Synchronising with MXChange

When changes (such as address or data type changes) are made to tags linked to MXChange, the new details are notified to other applications using those tags. This applies when MX OPC server modifies tags, or when another application modifies tags that MX OPC Configurator has imported from MXChange.

### Notification of changed tags in MXChange Server

If online changes are enabled and a tag is modified in the MXChange Server, a change notification message like the one below will appear:



**Notification of Changed Tags in MXChange Server**

Select either:

- **Yes** - to see the details of the changes immediately, or
- **No** - to deal with the changes later. If you select this option, no further changes will be shown until after the next synchronisation with MXChange.

If you select Yes, details of the changes will be shown as in the example picture below.

You can show changes at any time using the **File -> MXChange Project Actions -> Show Changes** menu option. This is the only way to view changes if online changes have not been enabled for the project.



**Details of changes**

The options on this dialog are:

- **Synchronise now** - synchronise with the MXChange server immediately (see below)
- **Save** - Save the list of changes to a text file
- **Close** - close the dialog without synchronising.

If you select **Synchronise now**, the MX OPC server and MXChange tag lists will be synchronised, and all the changes will be processed.

The tag lists can be synchronised at any time using the menu option **File -> MXChange Project Actions -> Synchronise**.

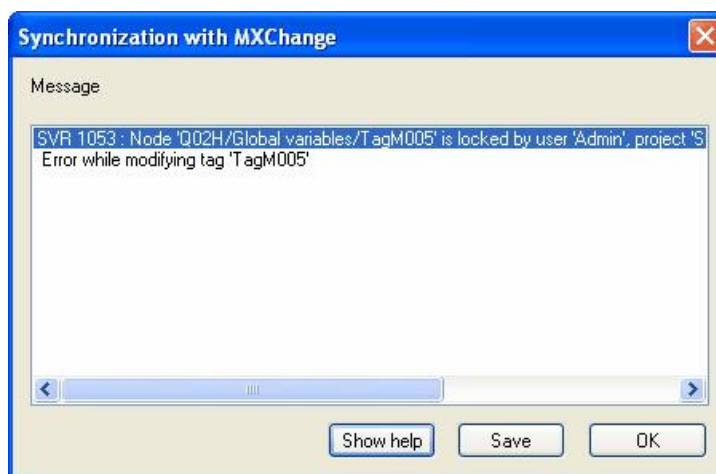
If MX OPC Configurator and a GX IEC Developer project have both made changes to the same tag, the changes made by GX IEC Developer will take priority.

If the synchronisation succeeds, the message below will appear:



**Details of changes**

If the synchronisation fails, you will be shown the details of the errors as shown below:



**Synchronisation errors**

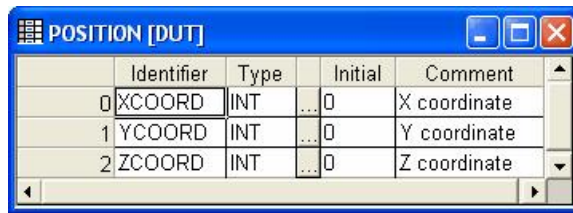
On this dialog, you can select:

- **Show help** - if you click on a line containing an MXChange error code (such as SVR-1053 in the example), the **Show help** button will show an explanation of the error and advice on how to fix it.
- **Save** - save the error messages to a text file
- **OK** - close the dialog

## 9.7 DUT tag support

From version 6.00 onwards, MX OPC Configurator can import tags defined in GX IEC Developer as *Data Unit Types (DUTs)*. These are structured data types, where the structure is defined once as a data type consisting of sub-elements of simple data types (INT, BOOL etc.), and then many different variables can be created using the type definition. Each new variable will inherit all the sub-elements of the type. This is helpful to users in that less data entry is required when defining objects with compound data, reducing the risk of typing errors and helping to maintain a logical structure to the program.

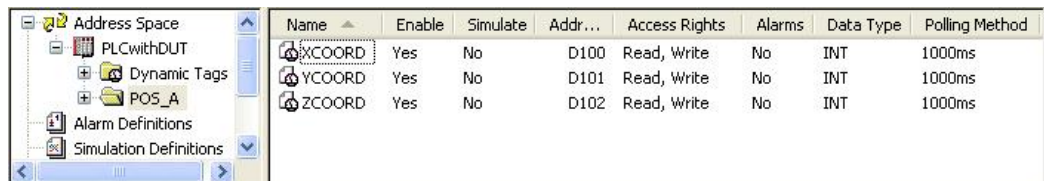
As an example, a 'position' structure could be defined with three integer sub-elements XCOORD, YCOORD and ZCOORD for the position coordinates, and then variables could be defined using the 'position' type - the DUT definition in GX IEC Developer would look like the figure below. Each variable will then have an 'XCOORD', 'YCOORD', and 'ZCOORD' element to store a whole position. This saves time compared with the alternative of defining variables such as e.g. Pos1X, Pos1Y, Pos1Z, Pos2X, Pos2Y, Pos2Z etc. to store the same data.



DUT definition in GX IEC Developer

In MX OPC Configurator, DUT variables are stored using the group / folder structure. The DUT variable is shown as a top-level folder containing one variable for each of the sub-elements (from the DUT type definition). Each element may have its address assigned independently in GX IEC Developer. The sub-elements can be read from an OPC client in the same way as any other tag. Individual elements of a DUT variable are referenced using dot notation e.g. DUTvariable.DUTELEMENT. The sub-element part has the same 32-character limit as the variable name.

A variable 'POS\_A' defined with the above type would appear as in the picture below.



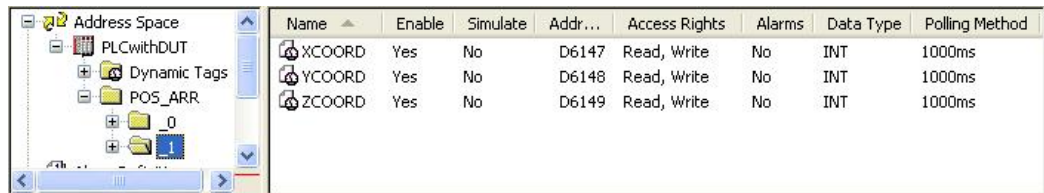
DUT variable in MX OPC Configurator



Arrays of DUT variables are handled slightly differently. These are represented in MX OPC Configurator as a top-level folder for the DUT tag itself, with another folder for each of the array elements underneath it. Finally, under each of these array element folders will be one variable for each of the elements (from the DUT type definition). Storing the tags in this way means that OPC clients that do not support arrays or structures can still reference single elements within a DUT array variable, as long as they have the ability to browse hierarchically for tags.

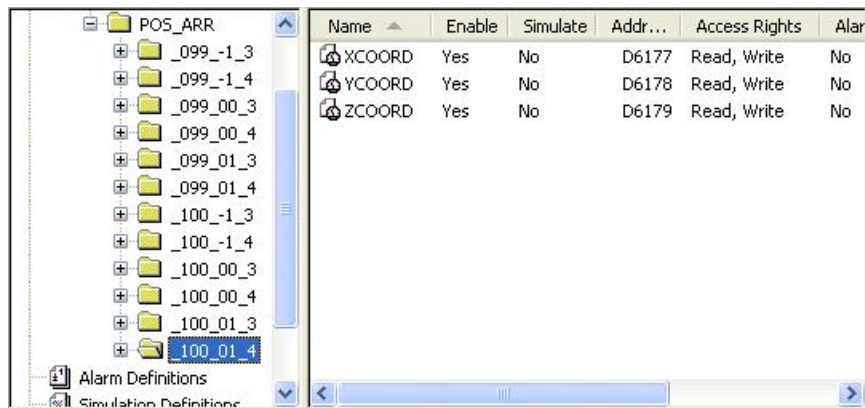
The name of the main DUT group node matches the DUT variable name in GX IEC developer. For DUT array tags, the DUT element group node will be given a name representing the array index e.g. '\_0' (for [0]), '\_-1' (for [-1]) or '\_0\_1' (for [0,1]). This format should be acceptable in most client applications' naming conventions. To ensure correct sorting of array elements, leading zeroes will be added to the indices to pad each index value up to the maximum width for that dimension, e.g. for an array with dimensions [0..1,0..100,0..50] an index value of [5,2,8] would be padded to Element\_5\_002\_08.

A simple example of an array of DUT variables in MX OPC Configurator is shown below. This is a variable 'POS\_ARR' consisting of an array [0..1] using the POSITION type as above.



Array[0..1] of POSITION variables

A more complicated example follows. This is an array of variables using the POSITION type as above but with three dimensions and some negative index values.



Array[99..100,-1..1, 3..4] of POSITION variables

### 9.7.1 Restrictions on DUT variables

As MX OPC Configurator is a less specialised programming tool than GX IEC Developer, it does not store the full range of configuration information which is available within GX IEC Developer. For example, it does not store DUT type information or details of the ranges of system memory that can be used for new variables. As a result, certain operations on DUT variables cannot be performed safely from MX OPC Configurator, and should instead always be performed from GX IEC Developer.

MX OPC Configurator will not be permitted to make changes which modify the DUT structure. The modifications that can be made by applications other than GX IEC Developer are limited to 'safe' operations that do not attempt to change the structure (inherited from the DUT type definition).

MX OPC Configurator *is* permitted to:

- Modify the addresses of individual elements in a non-array DUT variable
- Delete entire DUT variables (from the highest level folder downwards)

MX OPC Configurator is *not* permitted to:

- Create new DUT tags
- Modify the addresses of array DUT tags
- Delete part (but not all) of a DUT variable – this would be equivalent to changing the structure

### 9.7.2 Unallocated addresses

In GX IEC Developer, sub-elements of a DUT variable do not always have to be allocated a fixed address in PLC memory. Instead, the user can leave one or more addresses for the tag 'blank' at design time. When the project is next built, these addresses are replaced with new addresses allocated from the system memory range. These addresses can change later as other variables are added or removed.

If the GX IEC Developer user saves the project without first building it, these 'blank' (unallocated) addresses will be sent to other applications such as MX OPC Configurator, where they are displayed as the word 'Unallocated'. The runtime module will treat these addresses as badly-configured tags, because they do not have an address that can be used to read them.

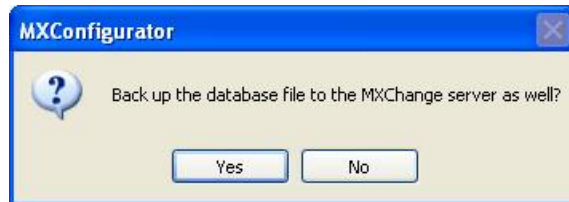
If you see an 'Unallocated' address in the configurator, it means that the GX IEC Developer user needs to build, download and save the project in GX IEC Developer so that the addresses are updated in MX OPC Configurator to match.

If the GX IEC Developer project is stored on the same computer as MX OPC Configurator, you can use the [Start GX IEC Developer](#)<sup>[53]</sup> context menu item to run GX IEC Developer to make this change. Alternatively, you may prefer to use fixed addresses for all DUT variables in GX IEC Developer, so that unallocated addresses are never passed to other client applications.

## 9.8 Back up / restore from MXChange

In a similar way to the options to Backup and Restore to/from a file, the MX OPC Configurator project can be backed up to or restored from the MXChange server. A copy of the MX OPC Configurator can be saved to the MXChange server database along with the MX OPC project information.

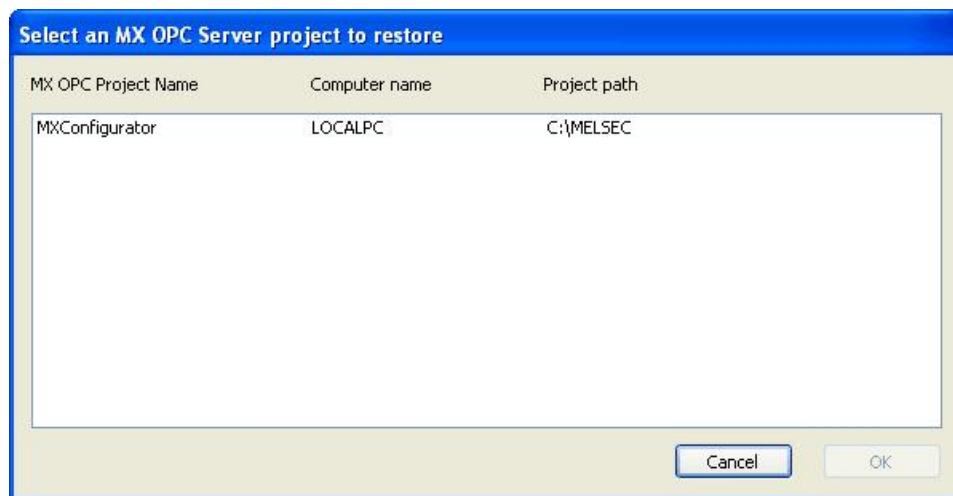
When you are backing up the database using the **File -> Backup to file** menu item and have a connection to an MXChange database, you will be prompted to back up the file to the MXChange server as well:



**Backup to MXChange prompt**

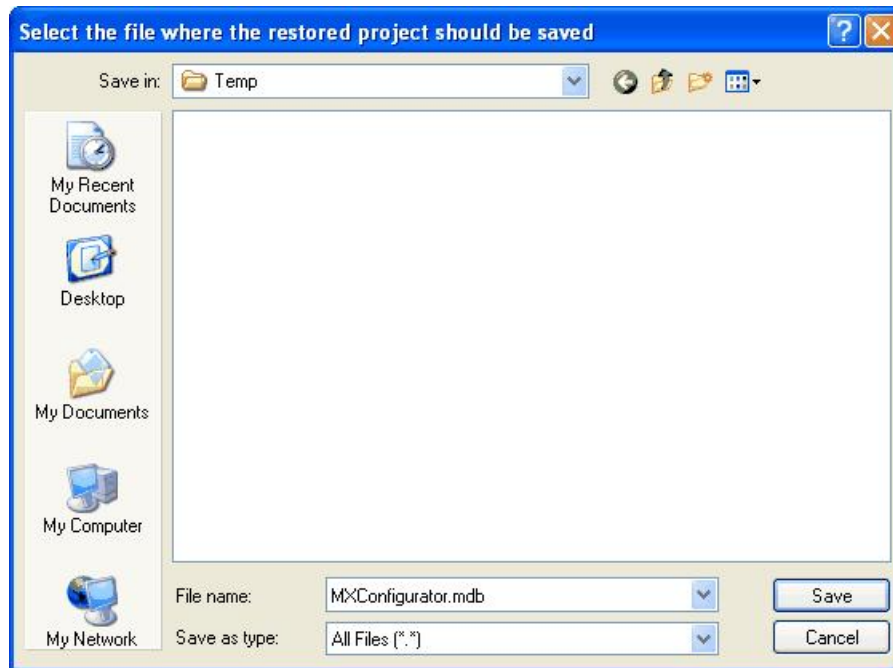
Select **Yes** to store a separate copy of the project file in the MXChange server.

To restore a saved database from the MXChange database, connect to the MXChange database, and then select **File -> MXChange Project Actions -> Load project from MXChange**. You will be prompted to select the MX OPC Configurator project to restore:



**Restore from MXChange**

After selecting the project to restore, you will be prompted to select the directory where the restored file should be written:



**Select name for restored file**

On clicking **Save**, the file will be restored from the server and opened as the current configuration.

## 10 Using iQ Works / MXChange2

iQ Works is an integrated suite of automation development tools, which includes configuration tools such as GX Works2 for PLC programming.

The iQ Works Navigator application can be used to create a *Workspace*, which contains all projects in a single system, and network diagrams showing how the projects are connected. iQ Works allows the user to define *System labels* which can be shared between the different projects in the workspace. When the address of a system label is changed, the change can be propagated to the other projects automatically, reducing the potential for configuration errors such as two applications using different addresses for the same label. Labels can be grouped into separate *System label lists*.

In iQ Works, projects created with applications like GX Works2 are known as *assignment projects*. System labels can be *assigned* to these projects, to give the project ownership of the label. The assigned project is responsible for maintaining the details of the label such as its I/O address. Other projects can *reference* the label, but cannot change details such as the address. Addresses can only be modified by the owning assignment project.

Each MX device in MX OPC Configurator represents a connection to a PLC, programmed with a development tool such as GX Works2. When an MX OPC Configurator project is connected to iQ Works (using the MXChange2 interface), each MX device is linked to the corresponding PLC project in the iQ Works workspace. When new labels are added in GX Works2 and marked for export to iQ Works, they will be automatically imported into MX OPC Configurator with the correct details such as I/O address, label name and description. When labels are deleted or unassigned from the linked GX Works2 project, they are deleted from MX OPC Configurator.

MX OPC Configurator is a *reference* project in iQ Works, so after an MX device has been linked to an iQ Works project, it is no longer possible to change the details (such as the I/O addresses) of existing labels from within MX OPC Configurator, or to delete linked labels - this is described in more detail in [Restrictions on editing labels](#) [219]. The details must be modified in the assigned project (usually GX Works2) instead. New labels can be added (see [Adding iQ Works labels](#) [217]) but cannot be modified after they have been synchronised with iQ Works. The restriction on modifying labels is only removed after the MX device is [unlinked](#) [220] from iQ Works.

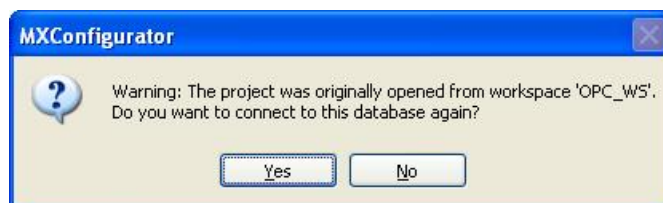
MX OPC Configurator projects do not appear on the network diagrams in Navigator.

**Note:** Modbus devices are not imported or exported within iQ Works, as there are no other development applications which use Modbus devices.

### 10.1 Connecting to iQ Works

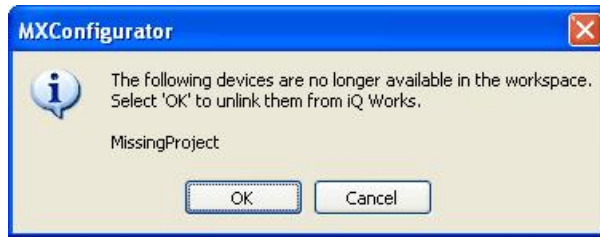
MX OPC Configurator can be connected to iQ Works by opening an existing iQ Works workspace, as described in [Connecting to a local workspace](#) [195] and [Connecting to a remote workspace](#) [198].

Once you have connected an MX OPC Configurator project to iQ Works, it will attempt to reconnect to the same iQ Works workspace automatically each time the project file is reopened. If the stored database details do not match the current workspace connection, you will be prompted to reconnect to the workspace that was originally used for the opened project, as shown below. Select **Yes** to connect to the previous server (recommended) or **No** to switch to the currently connected server.



Reconnect with previous details

On reconnection, MX OPC Configurator will check that all the GX Works2 projects that are linked to MX devices are still available in the iQ Works workspace. If any of the projects are missing or no longer contain labels, the warning message below will be shown:

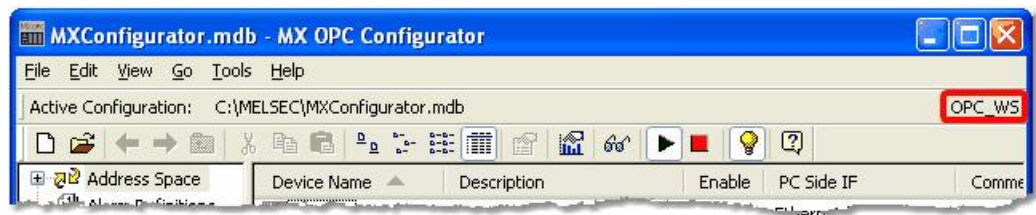


**Missing projects warning**

Select either:

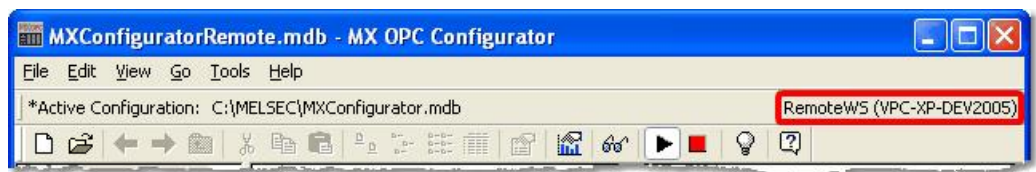
- **OK** - continue, and unlink the missing projects from the MX devices (equivalent to using the [Unlink device](#) menu item)
- **Cancel** - do not attempt to unlink the devices. This should be selected if (for example) the connection is not to the expected iQ Works workspace.

You can tell whether or not there is a connection to iQ Works, and if so to which workspace, from the status indicator shown at the right of the active configuration bar. The connection status is highlighted in the example picture below, where there is a connection to a workspace called *OPC\_WS*.



**iQ Works connection status (local)**

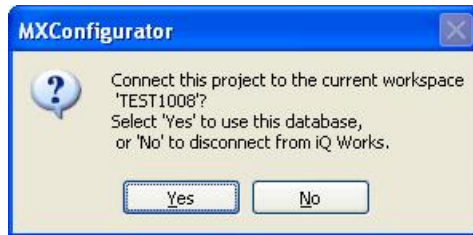
Where there is a connection to an iQ Works workspace on another computer, the remote computer name will be shown in brackets. In the example below, there is a connection to a workspace called *RemoteWS* on a computer called *VPC-XP-DEV2005*.



**iQ Works connection status (remote)**

If you create a new MX OPC Configurator database while connected to iQ Works, it will be linked to the iQ Works workspace automatically.

While there is a connection to an iQ Works workspace, if you open another configuration that is not linked to iQ Works, you will be prompted to link the opened configuration to the currently-connected workspace:



**Connect project prompt**

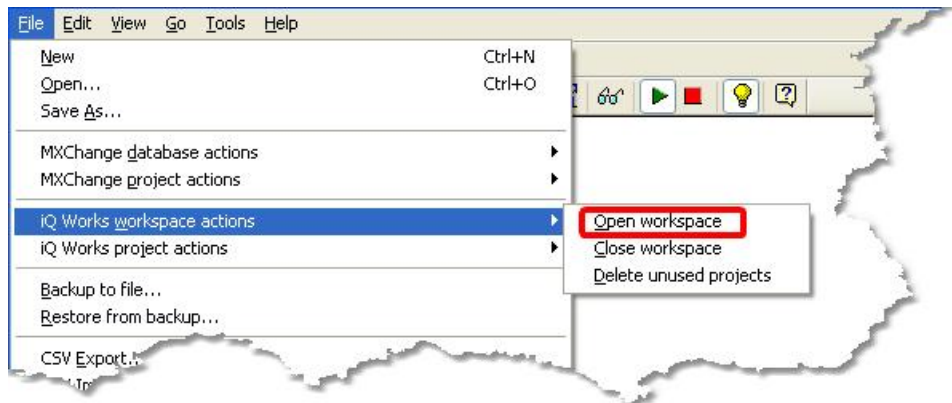
Select either:

- **Yes** - continue, and link the opened project to the currently-connected iQ Works workspace.
- **No** - do not link to opened project to the currently-connected iQ Works workspace. Instead, disconnect from iQ Works before opening the project.

### 10.1.1 Connecting to a local workspace

The steps to connect to a local workspace are:

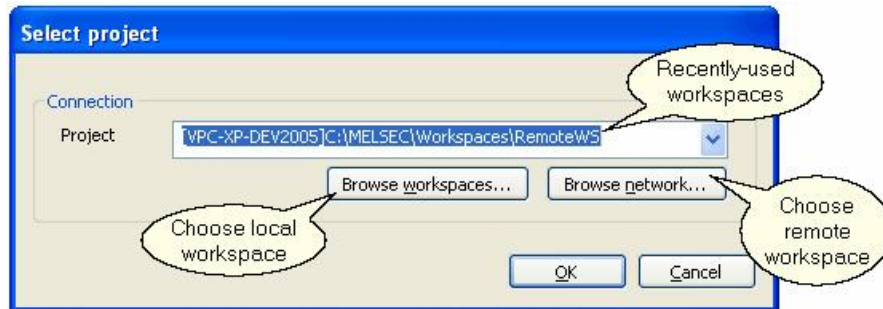
1. From the **File** menu, select **iQ Works workspace actions -> Open workspace**.



**Open iQ Works workspace**



2. From the **File** menu, select **iQ Works workspace actions -> Open workspace**. The **Open workspace** dialog will appear as shown below.

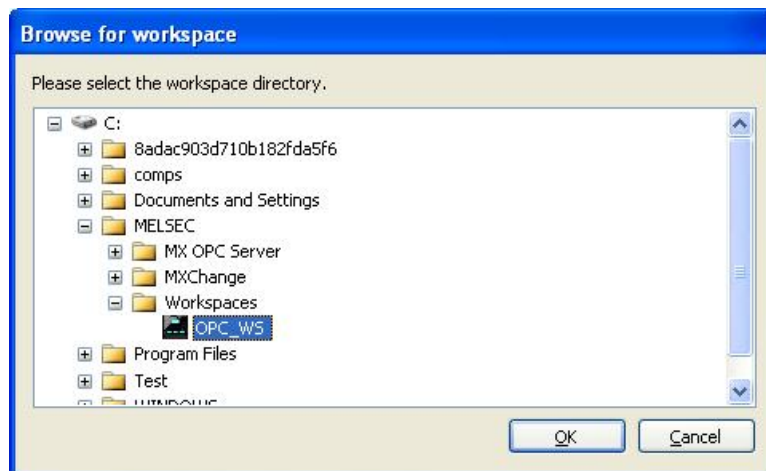


**Open workspace dialog**

The controls on this dialog are:

- **Project** - the selected project. Use the drop down arrow to select a recently-used workspace.
- **Browse workspaces** - browse the local computer for a workspace to use (see below)
- **Browse network** - browse the network for a workspace to use (see [Connecting to a remote workspace](#)<sup>[198]</sup>)
- **OK** - select the workspace
- **Cancel** - cancel and do not try to open a workspace.

3. After clicking **Browse workspaces**, you can select the location of the workspace file on the local computer.

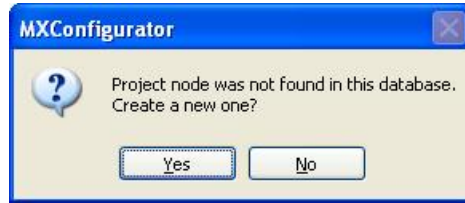


**Browse for local workspaces**



4. Click **OK** to select the workspace folder, then **OK** again on the **Open workspace dialog** to open the workspace.

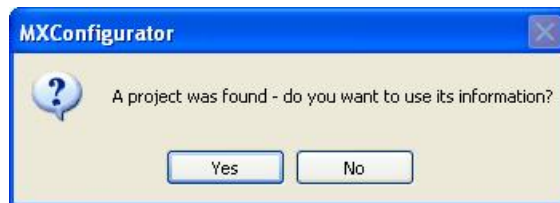
If no matching MX OPC Configurator project is found in the database, you will be prompted to create a new one:



**Create project prompt**

Select **Yes** to create a new MX OPC Configurator project in the workspace. Select **No** to cancel (for example, if a project should have been found in the workspace but was not, because the wrong workspace was opened).

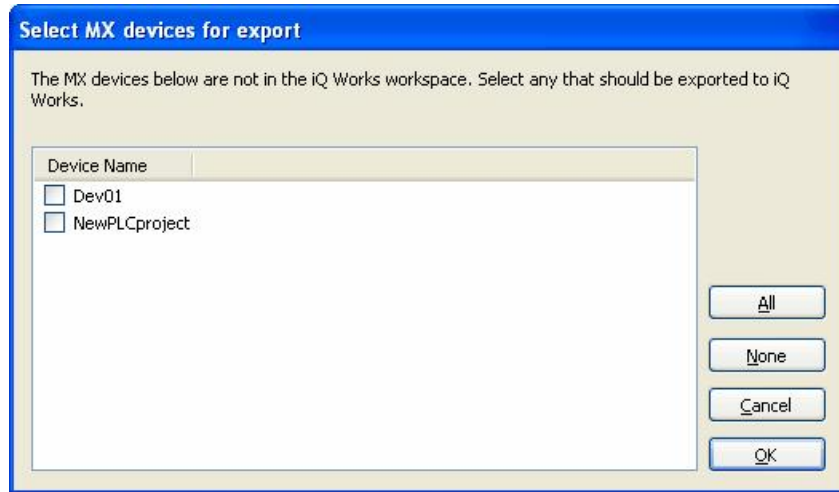
If there was already a matching MX OPC Configurator project in the database, you will be prompted to reconnect using the same details.



**Reuse prompt**

Select **Yes** to import PLC projects as new MX devices immediately - see [Importing from iQ Works](#) [210] for a description of the import procedure.

5. If there are already MX devices defined in the MX OPC Configurator project, you will be given the option to export them to the iQ Works workspace:



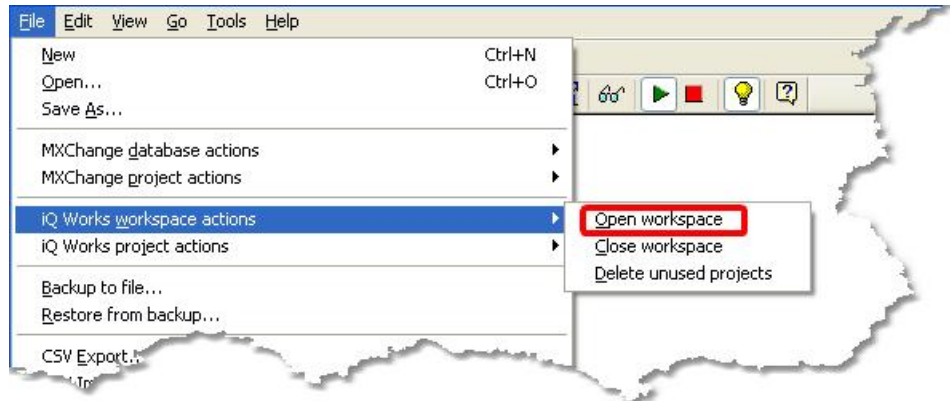
Select devices for export to iQ Works

For more information about exporting existing devices to iQ Works, see [Exporting to iQ Works](#)<sup>[203]</sup>.

### 10.1.2 Connecting to a remote workspace

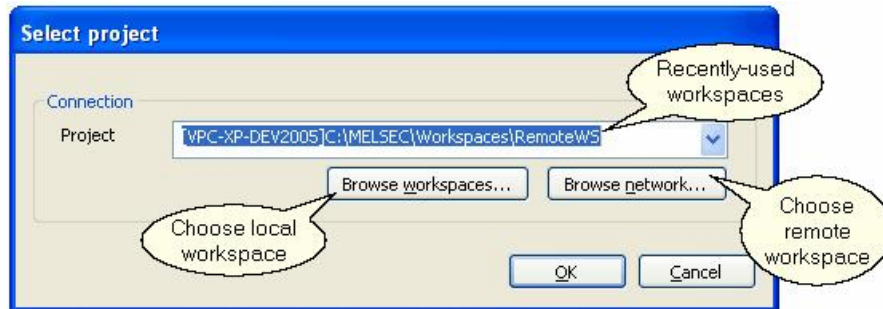
The steps to connect to a remote workspace are:

1. From the **File** menu, select **iQ Works workspace actions -> Open workspace**.



Open iQ Works workspace

2. From the **File** menu, select **iQ Works workspace actions -> Open workspace**. The **Open workspace** dialog will appear as shown below.

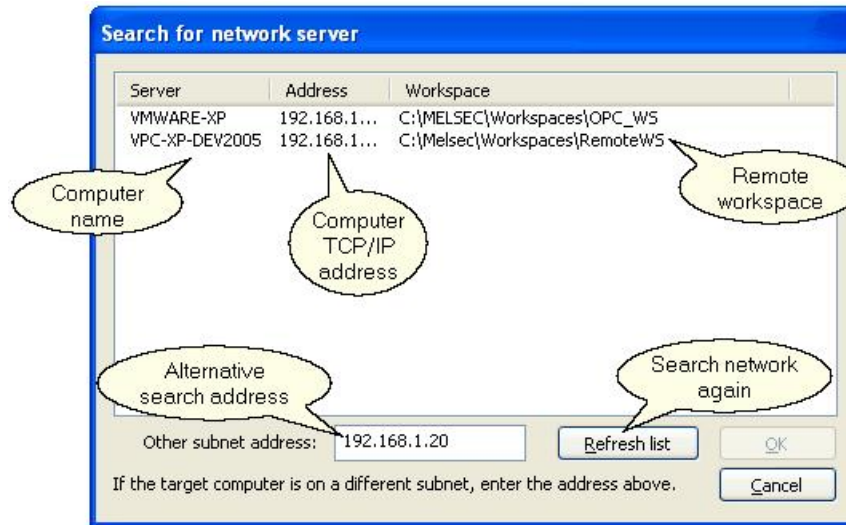


**Open workspace dialog**

The controls on this dialog are:

- **Project** - the selected project. Use the drop down arrow to select a recently-used workspace.
- **Browse workspaces** - browse the local computer for a workspace to use (see [Connecting to a local workspace](#)<sup>[195]</sup>)
- **Browse network** - browse the network for a workspace to use (see below)
- **OK** - select the workspace
- **Cancel** - cancel and do not try to open a workspace.

3. After clicking **Browse network**, there will be a short pause while the network is searched for other workspaces, and then the available workspaces will be shown as pictured below.



**Browse for remote workspaces**

The controls are:

- **Workspace list** - this shows the workspaces that were found on the local network, with the computer name, TCP/IP address and workspace path of each one. To select a workspace, click on it in the list.
- **Other subnet address** - depending on the configuration of your network router and switches, the server browsing may only be able to detect computers on the same subnet. If you know the name or address of the computer to browse / connect to, you can enter it here. When **Refresh list** is clicked, the named computer will be checked directly - this will often allow browsing of server information across subnets.
- **Refresh list** - this searches the network again, and will also use the **Other subnet address** value (if provided).
- **OK** - select the workspace
- **Cancel** - cancel without selecting a workspace.

*Note:* For a workspace on a remote computer to appear in the list, it must be actively in use on the target computer. Opening the workspace with Navigator will make sure that it is active. After the workspace has been used once, it can be selected from the recent projects list on the open workspace dialog.

4. Click **OK** to select the workspace folder, then **OK** again on the **Open workspace dialog** to open the workspace.

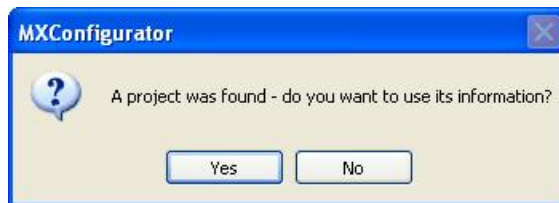
If no matching MX OPC Configurator project is found in the database, you will be prompted to create a new one:



**Create project prompt**

Select **Yes** to create a new MX OPC Configurator project in the workspace. Select **No** to cancel (for example, if a project should have been found in the workspace but was not, because the wrong workspace was opened).

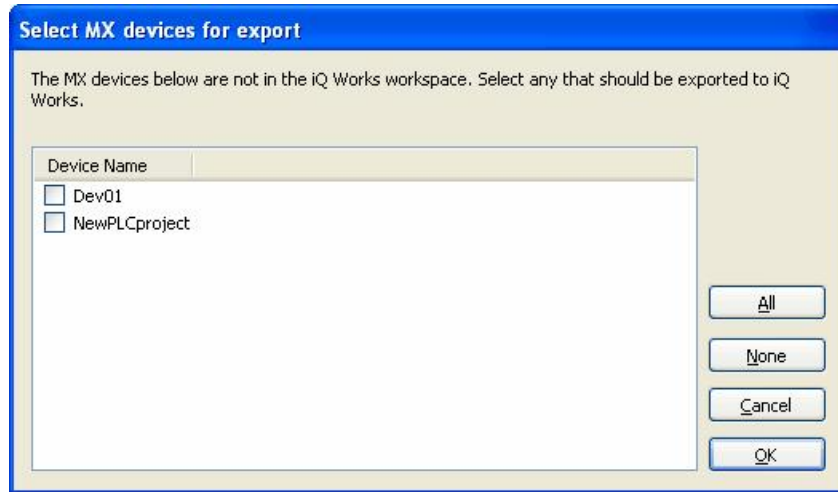
If there was already a matching MX OPC Configurator project in the database, you will be prompted to reconnect using the same details.



**Reuse prompt**

Select **Yes** to import PLC projects as new MX devices immediately - see [Importing from iQ Works](#) [210] for a description of the import procedure.

5. If there are already MX devices defined in the MX OPC Configurator project, you will be given the option to export them to the iQ Works workspace:



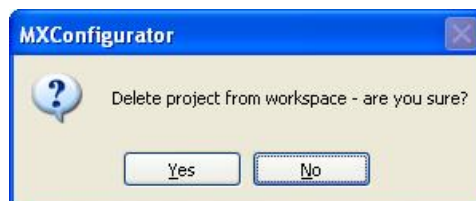
Select devices for export to iQ Works

For more information about exporting existing devices to iQ Works, see [Exporting to iQ Works](#)<sup>[203]</sup>.

## 10.2 Detaching from iQ Works

When you no longer wish to connect an MX OPC Configurator project to iQ Works, select **File -> iQ Works workspace actions -> Close workspace** to disconnect from the workspace and break the link to MX OPC Configurator. After using this option, MX OPC Configurator will no longer attempt to connect to the workspace each time the project is loaded. The project will still exist in the iQ Works workspace in case you need to reconnect to it later.

To remove the MX OPC Configurator project from iQ Works permanently, select **File -> iQ Works project actions -> Delete OPC project** to delete the project from the iQ Works workspace. Other application projects in the iQ Works workspace will not be affected. You will be prompted to confirm the deletion as shown below.



Delete project from iQ Works prompt

### 10.3 Supported labels

It is possible to define system labels in iQ Works that cannot be supported in MX OPC Configurator. For example:

- Labels assigned to address ranges that cannot be used in MX OPC Configurator
- Labels with data types such as 'Pointer' that are not available in MX OPC Configurator
- Array labels of simple types with more than one dimension or a lower bound less than 0.

Unsupported labels will not be saved to MX OPC Configurator when synchronising or importing from iQ Works. However, they are still considered as part of the MX OPC Configurator project in iQ Works, so MX OPC Configurator may still be sent change notifications for these labels. This is described in more detail in [Synchronising with iQ Works](#)<sup>[219]</sup>.

There are also some labels that can be defined in MX OPC Configurator which cannot be used in iQ Works. These include:

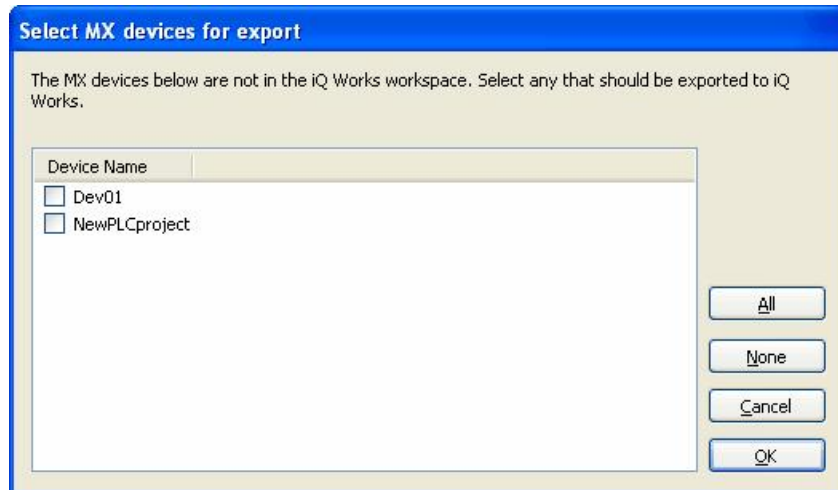
- Labels with device addresses beginning 'EG...', 'WR...' and 'WW...'

These labels will not be submitted to iQ Works when synchronising or exporting.

### 10.4 Exporting to iQ Works

When a MX OPC Configurator database is linked to iQ Works, each of the MX devices it contains can be linked to a GX Works2 project in the iQ Works workspace. The most common way to do this is to start by creating the GX Works2 project and variables and then export the details from GX Works2 to a new device in MX OPC Configurator. However, if you have already defined a device and tags in MX OPC Configurator, it is still possible to export these to a new GX Works2 project in iQ Works using the **Export devices to GX Works2** option:

1. From the **File** menu, select **iQ Works project Actions > Export devices to GX Works2**.
2. You will be prompted to select the devices to export to iQ Works, as shown in the picture below. Devices which have already been linked to iQ Works will not be shown in the list.



Select devices for export to iQ Works

To select a device for export, click on the box next to the device name so that it appears checked.

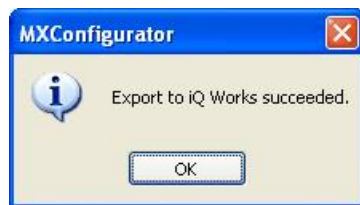
The controls on this dialog are:

- **All** - select all available devices
  - **None** - deselect all devices
  - **Cancel** - close the dialog without exporting to iQ Works
  - **OK** - export the selected devices to iQ Works
3. If the workspace contains more than one *System label list*, you will be prompted to select which list should be used for the labels exported into the GX Works2 project. All labels will be exported into the same list. After selecting a list, click **OK** to continue.



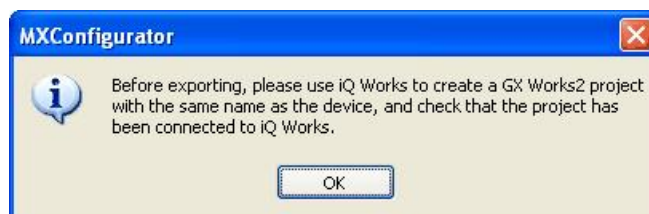
**Select system label list**

4. The MX OPC Configurator labels on the device will be exported to the project. If everything goes well, the following message box will appear:



**Export complete without errors**

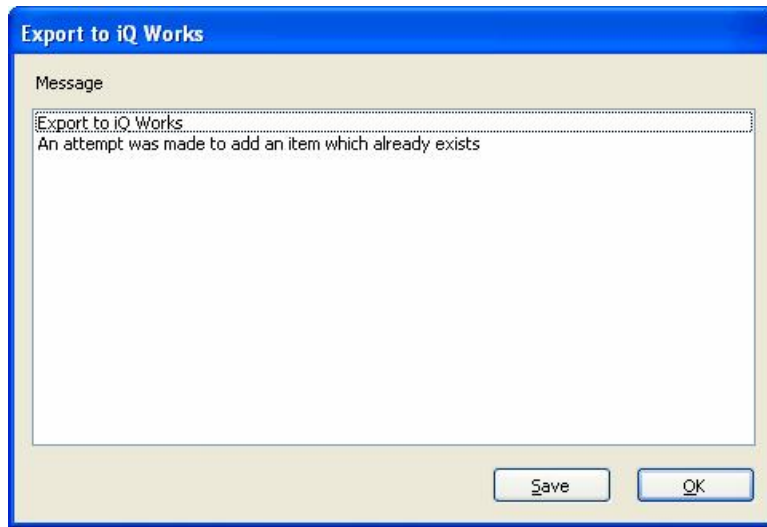
If the message below is shown, MX OPC Configurator was unable to find a GX Works2 project with the same name as the device. Make sure that a GX Works2 project with the same name as the device has been created in iQ Works. The project must also have been used with iQ Works at least once to make it visible to MX OPC Configurator. One way to do this is to open the GX Works2 project, open the global variable list, and select 'Import' then 'Cancel'.



**GX Works2 project not found**

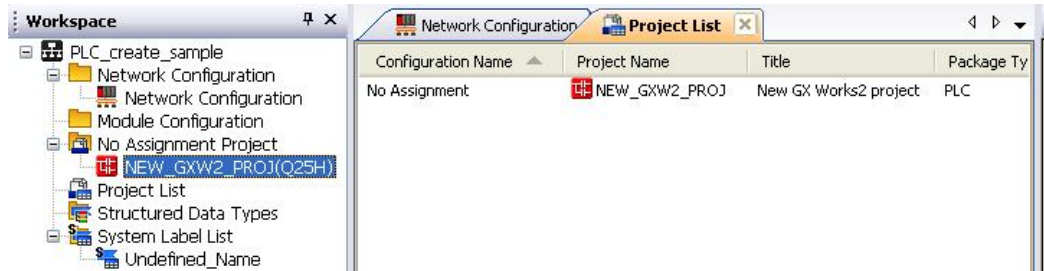


5. If the export is unsuccessful, then you will see a list of problems encountered during export similar to the one shown in the figure below. To save the results, click the **Save** button and specify a name for the file.



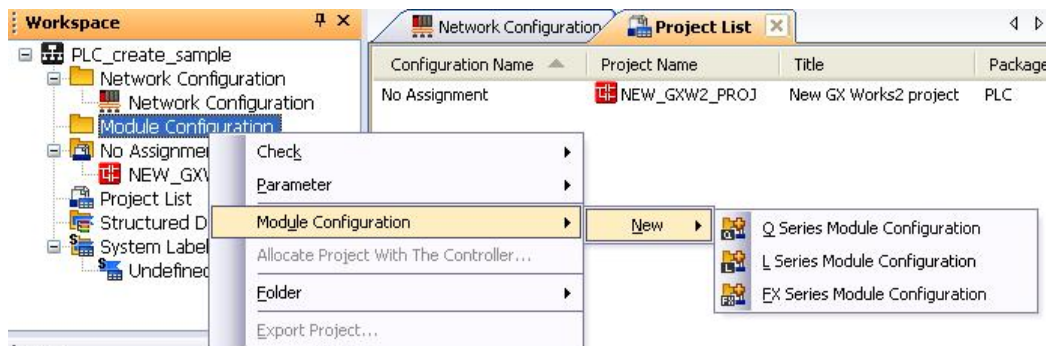
Export unsuccessful

6. After the MX device has been exported as a GX Works2 project, the new project can be set up in Navigator. When you start Navigator, the project will initially appear in the 'Unassigned projects' list:



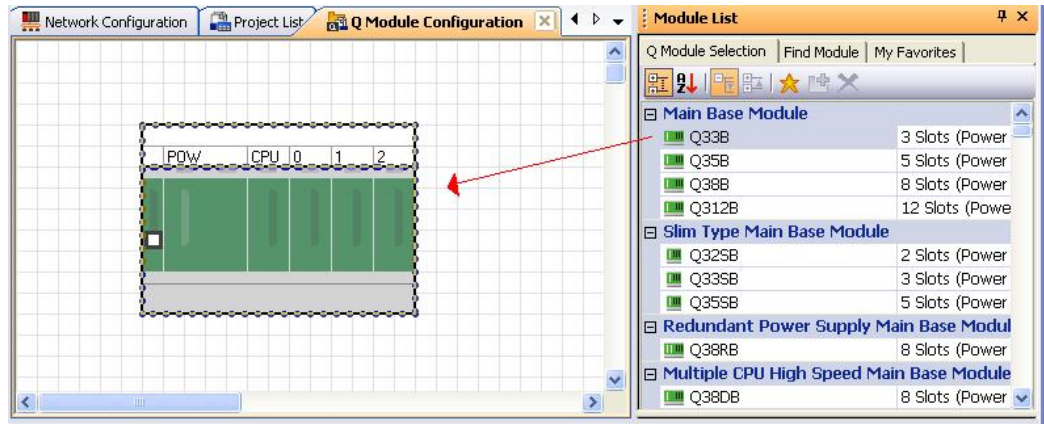
Exported project in the 'unassigned projects' list

A new module configuration will need to be created for the project, selecting the correct PLC type:



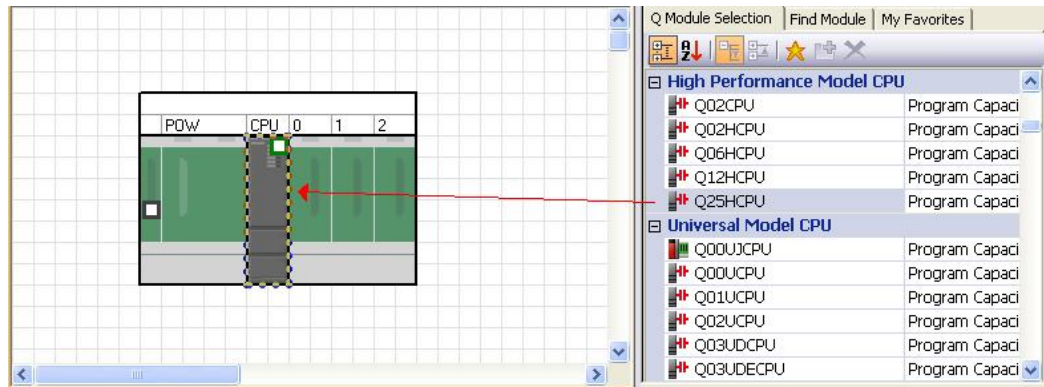
New module configuration menu

In the module configuration diagram, start by adding a suitable base unit for the PLC using drag and drop:



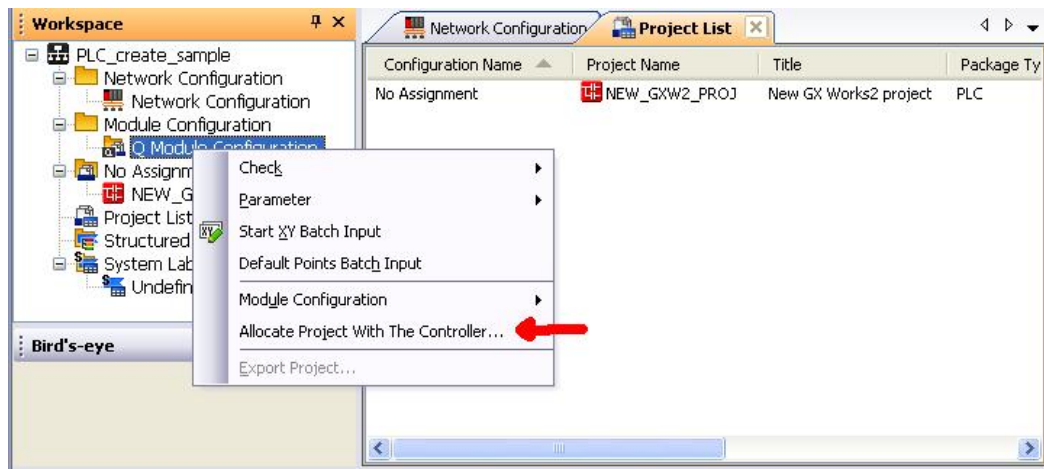
Add base unit

Next, add the right CPU to the base unit by dragging and dropping it onto the CPU slot. Items such as power supply modules and I/O cards can also be added in this way.



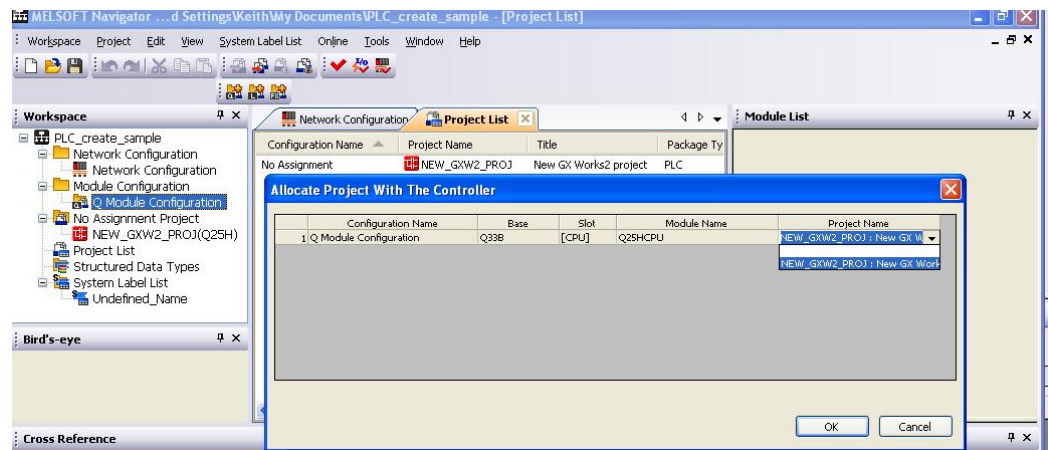
Add CPU

The new GX Works2 project must be *allocated* to the module configuration using the 'Allocate project with the controller' menu:



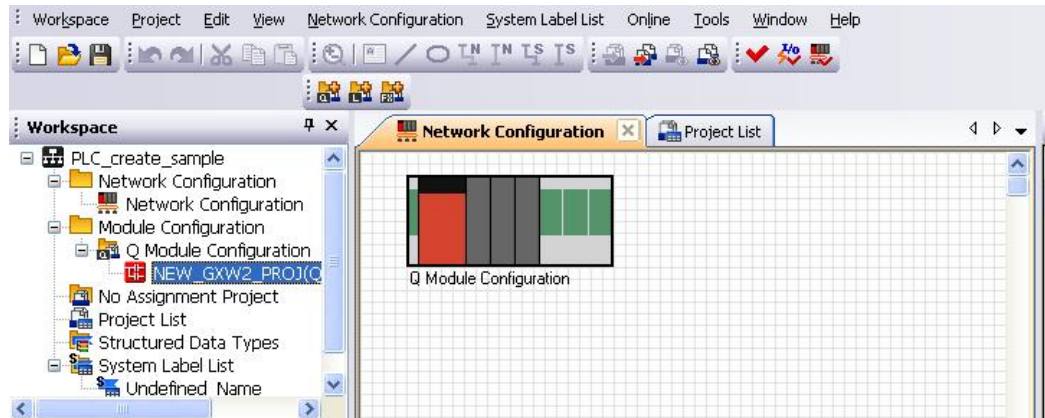
Allocate project with the controller - menu

In the dialog that appears, select the new GX Works2 project to use with the module configuration diagram.



Allocate project with the controller - dialog

The new project can now be used in Navigator, and will appear in the network diagram.

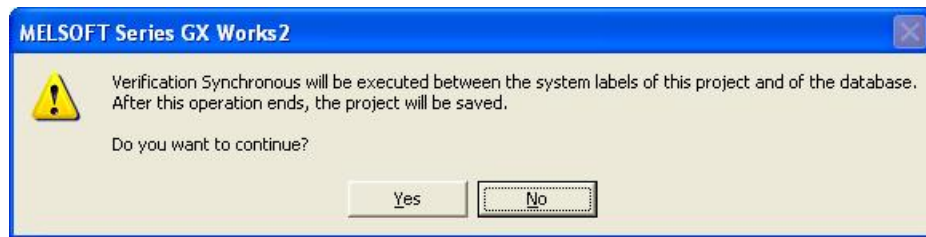


**New project on network diagram**

- 7. To use the exported labels in GX Works2, double click on the new project name in Navigator to open GX Works2, then select:

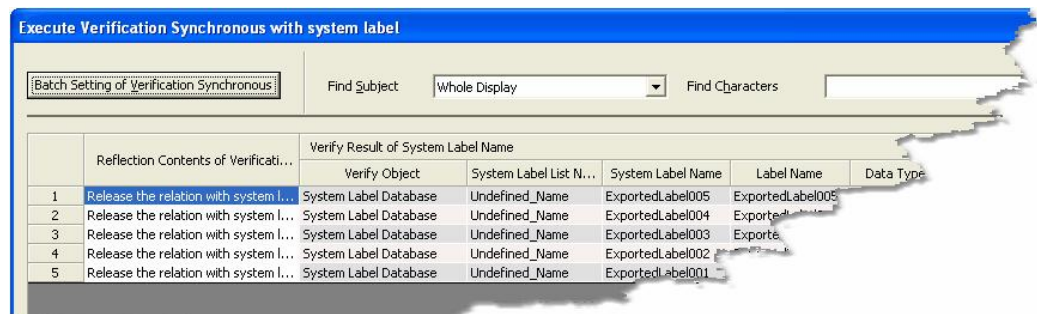
**Edit -> System Label -> Execute Verification Synchronous with System Label.**

At the prompt (shown below), select **Yes**.



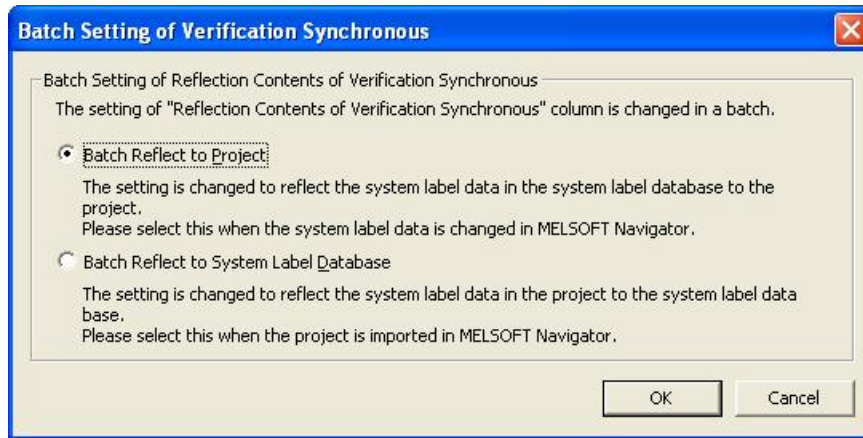
**Confirm verification synchronous**

The *verification synchronous* dialog will appear, as shown in the example picture below where five new labels are being added. The action to take is shown in the column on the left, which will initially release the labels from the project by default.



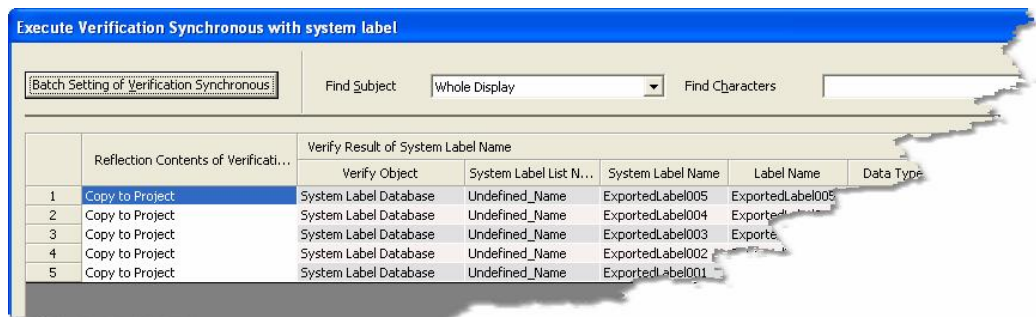
**Verification synchronous dialog**

Click on the **Batch Setting of Verification Synchronous** button at the top left of the dialog. Select the batch setting **Batch Reflect to Project** as shown in the figure below, then click **OK**.



**Batch setting of verification synchronous**

On returning to the *verification synchronous* dialog, the left column will have changed to show **Copy to Project** for each item.



**Verification synchronous dialog**

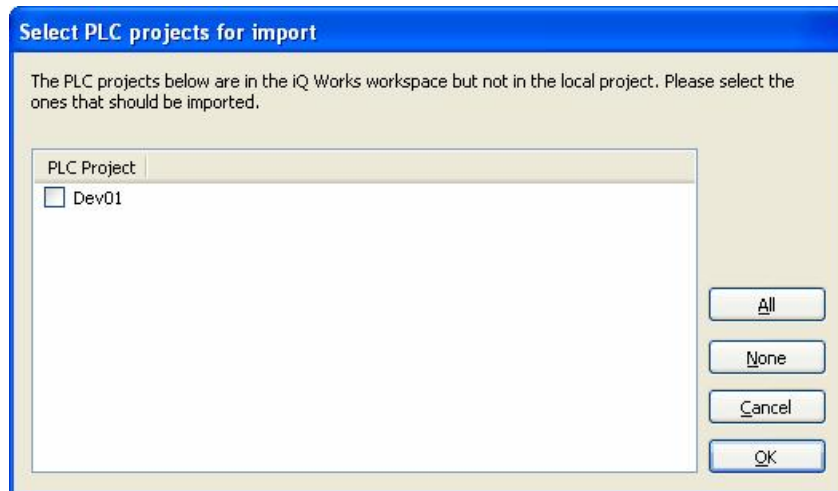
Click on the **Reflection** button to copy the exported labels into the GX Works2 project.

## 10.5 Importing from iQ Works

Once a connection to the iQ Works workspace has been established, you can also import the data from GX Works2 projects into your configuration database from the iQ Works workspace. When you import data from iQ Works, each GX Works2 project in the workspace can be imported as a new device in the MX OPC Configurator project.

To import a project from the workspace:

1. From the **File** menu, select **iQ Works project Actions > Import GX Works2 project**.
2. You will be shown a list of GX Works2 projects in the workspace which have not already been linked to the current MX OPC Configurator project, as in the figure below.



Select PLC projects to import

To select a device for import, click on the box next to the device name so that it appears checked.

*Note:* Since the project will be created as a new device in MX OPC Configurator, you will only be able to select PLC projects with names that would be valid as device names. Device names are limited to 32 characters and cannot contain spaces. To import a PLC project with a name that would be invalid as a device, rename the project before attempting to import it.

The controls on this dialog are:

- **All** - select all available devices
- **None** - deselect all devices
- **Cancel** - close the dialog without importing from iQ Works
- **OK** - import the selected devices from iQ Works



3. Each time you import a device, a new [Communication Setting Wizard](#)<sup>[95]</sup> appears to allow the connection details to be set up. Select the link to the PLC (for example, Ethernet connection or serial connection) in the same way as for a new device.
4. After the devices are created, all [supported labels](#)<sup>[203]</sup> will be imported into the MX OPC Configurator database. On successful completion, the message below will be shown:



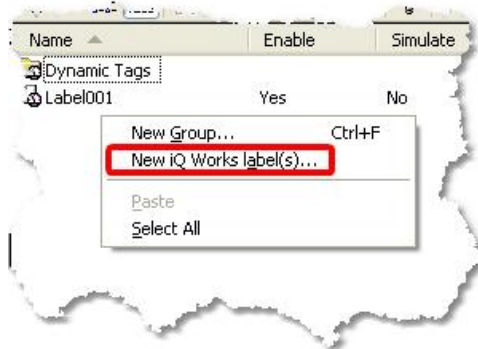
**Import Successful**

## 10.6 Adding iQ Works labels

In MX devices that are linked to GX Works2 projects in iQ Works, the **New DataTag** menu item is replaced with **New iQ Works label(s)...**, which allows new labels to be created for use in iQ Works.

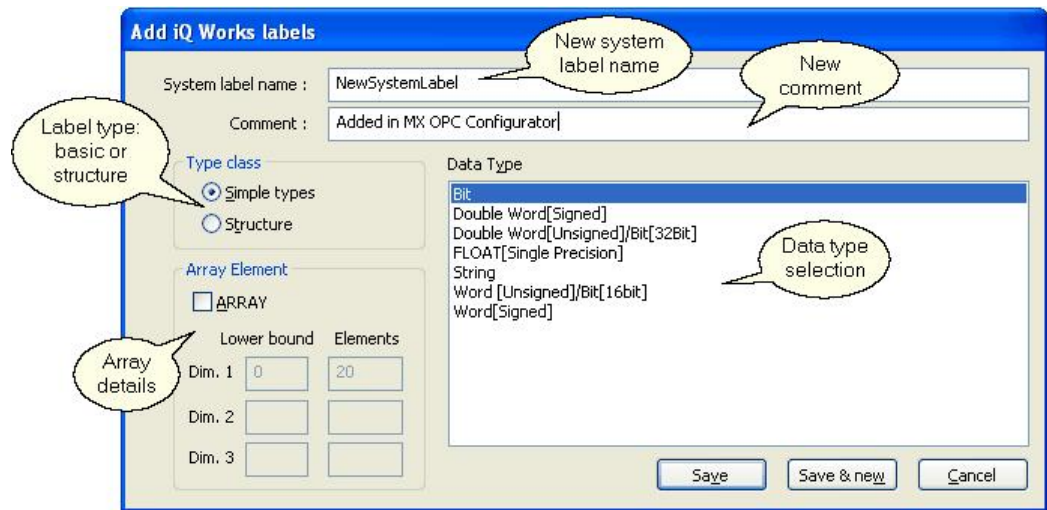
To add labels to a connected iQ Works project:

1. Right click on the background of the labels window and select **New iQ Works label(s)...** from the context menu, as shown in the figure below.



**New label context menu**

2. The **Add iQ Works labels** dialog will be shown as in the figure below.



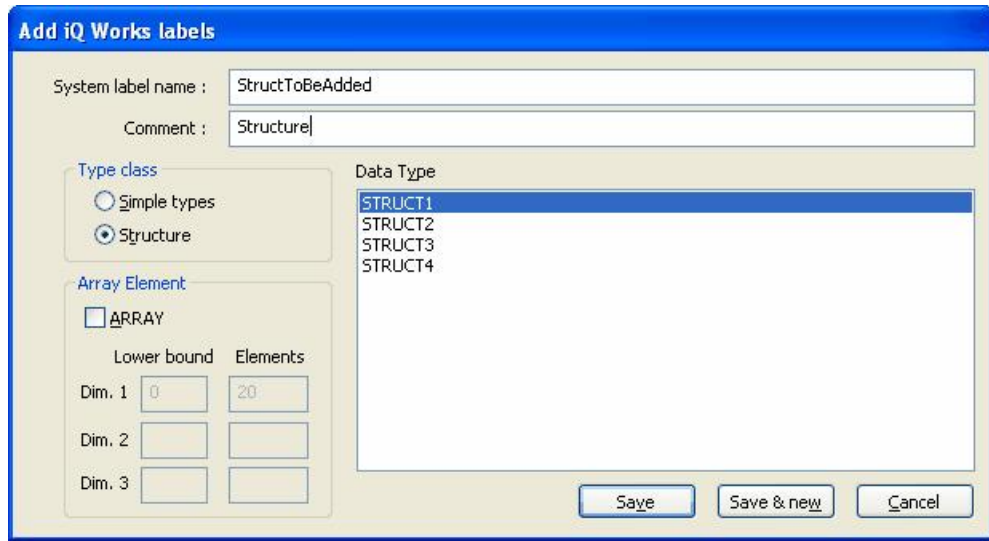
**Add iQ Works labels**

The controls on this dialog are:

- **System label name** - name to use for the new system label. This cannot be blank or the same as any existing label name.
- **Comment** - the comment for the new system label name
- **Type class** - whether to create a simple data type (e.g. Word) or a structured data type (see below)
- **Data type** - the data type for the new label. Depending on the selection in **Type class**, this may be a simple data type or a structured data type. The list of simple data types is limited to types that can be supported in iQ Works and in MX OPC Configurator.
- **Array element: ARRAY** - check this box to create an array variable. For simple types, only one-dimensional arrays are permitted. Arrays of strings cannot be created.
- **Array element: Lower bound** - for arrays, enter the lower bound of the array (usually 0). Note that negative values are not permitted for simple label types, as although iQ Works supports them, MX OPC Configurator does not. Arrays of structures can have negative lower bounds.
- **Array element: Elements** - for arrays, the number of elements in the array. For example, items with type array(0..19) or array(10..29) would have 20 elements.
- **Save** - save the new label in MX OPC Configurator and close the dialog.
- **Save and new** - save the new label in MX OPC Configurator, but leave the dialog open to add more new labels.
- **Cancel** - close the dialog without importing from iQ Works



If the iQ Works workspace contains structure definitions, the type class can be set to 'Structure' to create a structure label (see [Structure support](#)<sup>[227]</sup>). The data types list will change to show the available structure types.



**Add iQ Works labels (structure type)**

For structures, arrays of up to three dimensions are permitted, and the lower bounds can be negative. The controls for each dimension are enabled when the dimension before it has been filled in. To disable a dimension, clear the 'Lower bound' and 'Elements' values.

3. After a label is saved, it will appear in the label list in MX OPC Configurator with default details and no assigned address (the address will appear as 'Unallocated'). The label can be edited further (e.g. to set addresses) up until the next synchronisation with iQ Works. Unless there is a reason to set a specific address, it is safe to leave the address blank. GX Works2 will assign an address when the project is compiled and saved, and the new address will be *reflected* back to MX OPC Configurator during synchronisation. The addresses of arrays of structures cannot be set in MX OPC Configurator, and should always be set in GX Works2 instead.

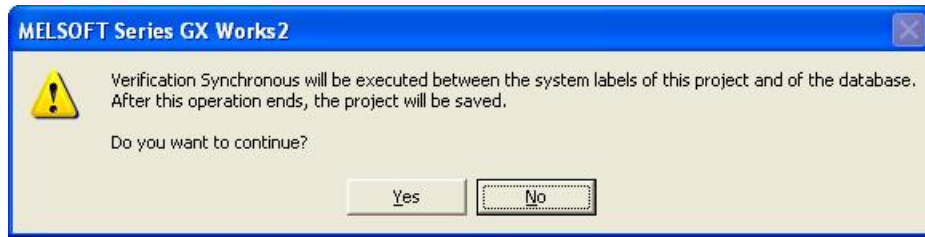
New labels can also be deleted from MX OPC Configurator if they have not yet been synchronised. After synchronisation, they must be deleted from GX Works2 instead.

4. To write the new labels to the workspace, it is necessary to synchronise the workspace as described in [Synchronising with iQ Works](#)<sup>[219]</sup>, by selecting **File -> iQ Works project actions -> Synchronise**. If there is more than one system label list in the iQ Works workspace, you will be prompted to select the system label list that should be used for the new system labels:



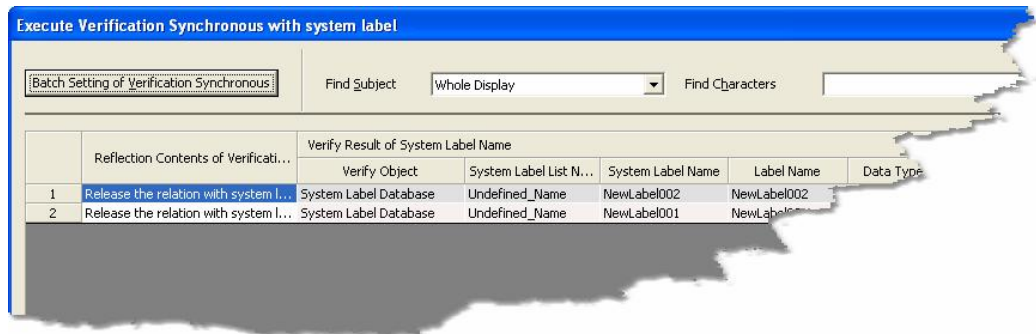
**Import Successful**

- To use the new labels in GX Works2, start GX Works2, and select:  
**Edit -> System Label -> Execute Verification Synchronous with System Label.** At the prompt (shown below), select **Yes**.



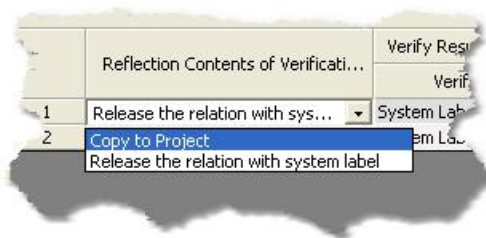
**Confirm verification synchronous**

The *verification synchronous* dialog will appear, as shown in the example picture below where two new labels are being added. The action to take is shown in the column on the left, which will initially release the labels from the project by default.



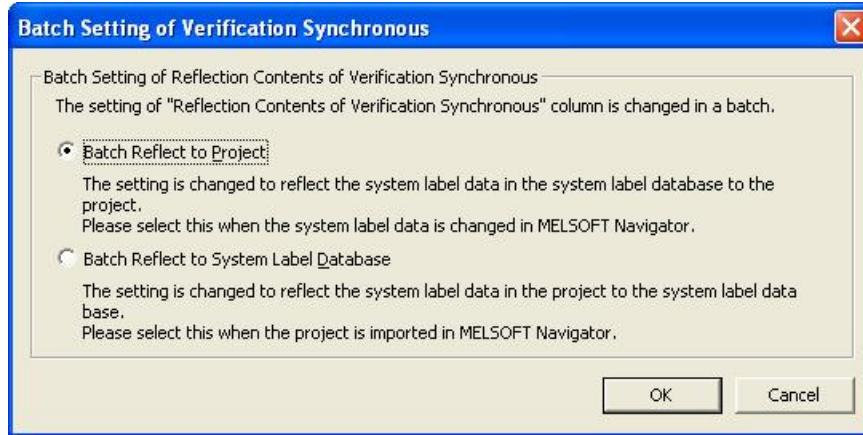
**Verification synchronous dialog**

For each new label, click on the 'Reflection contents...' text and select 'Copy to Project' from the list.



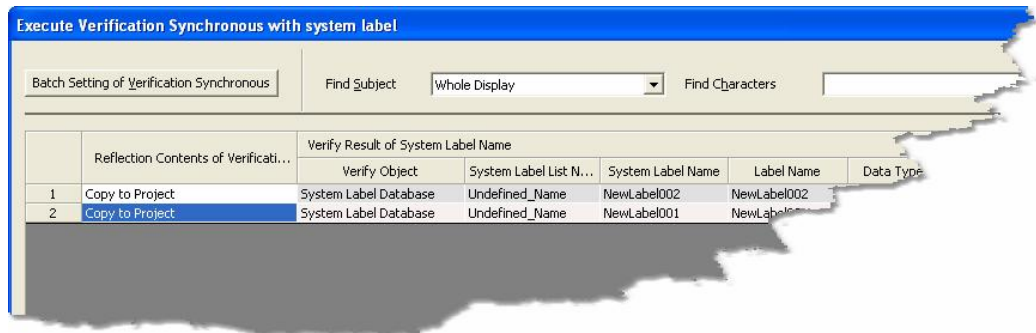
**Selecting 'Copy to Project'**

Alternatively, if the list only shows new labels, the 'Copy to project' setting can be selected for all items at once. Click on the **Batch Setting of Verification Synchronous** button at the top left of the dialog. Select the batch setting **Batch Reflect to Project** as shown in the figure below, then click **OK**.



**Batch setting of verification synchronous**

On returning to the *verification synchronous* dialog, the left column will have changed to show **Copy to Project** for each item.



**Verification synchronous dialog**

Click on the **Reflection** button to copy the new labels into the GX Works2 project.

## 10.7 Restrictions on editing labels

MX OPC Configurator is a *reference* project in iQ Works. It imports labels from *assignment* projects such as GX Works2. System labels are owned by the assignment project, and although other applications such as MX OPC Configurator can read the labels, modifications are not permitted in other projects.

System labels can still be modified or deleted, but the changes must be made in the project to which they are assigned (usually GX Works2).

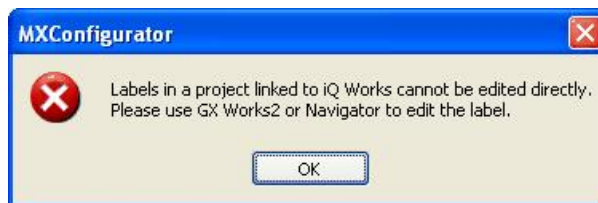
MX OPC Configurator *is* permitted to:

- Modify the addresses of individual elements in a structure variable (except arrays of structures), up until the point where it is synchronised with iQ Works
- Delete entire structure variables (from the highest level folder downwards), unless they have been synchronised with iQ Works

MX OPC Configurator is *not* permitted to:

- Modify the addresses of labels that have been synchronised with iQ Works
- Delete labels that are linked to iQ Works
- Add or delete part (but not all) of any structure variable – this would be equivalent to changing the structure

If an attempt is made to modify a system label in a way that is not permitted, the warning message below will be shown:



**Edit not permitted warning**

There is a restriction on the number of labels that can be added to a single system label list in iQ Works. This limit is 1000 labels in the current version. If this limit is likely to be exceeded when exporting or synchronising, the message below will be shown and the action will be cancelled. Labels should be divided between system label lists so that no individual list exceeds the maximum number of labels.



**Too many labels in list warning**

## 10.8 Change notification

When changes (such as setting a new I/O address) are made to system labels linked to iQ Works, the new details are notified to all other applications that use those labels. This will typically happen when a GX Works2 project that MX OPC Configurator has imported from iQ Works is compiled and saved.

### Notification of changed tags in the iQ Works workspace

When a change to an existing label is detected, the notification message below will appear:



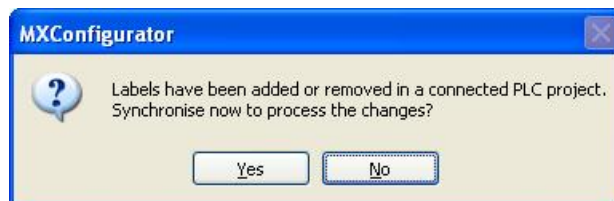
**Notification of changed labels in iQ Works**

Select either:

- **Yes** - to show the details of the changes immediately, equivalent to selecting **File -> iQ Works project actions -> Show changes**, or
- **No** - to deal with the changes later. If you select this option, no further changes will be shown until after the next [synchronisation](#)<sup>[219]</sup> with iQ Works.

### Notification of new labels in a connected project

When new labels are added to (or removed from) a GX Works2 project which is linked to an MX device, the message below will be shown.



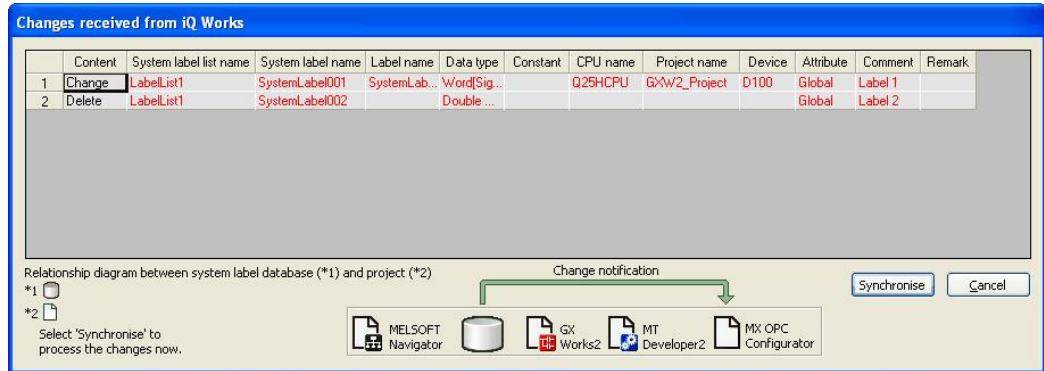
**Notification of new labels in a connected project**

Select either:

- **Yes** - to [synchronise](#)<sup>[219]</sup> immediately, or
- **No** - to deal with the changes later.

**Showing changes**

You can show changes at any time using the **File -> iQ Works project actions -> Show changes** menu option. This will show a notification dialog like the one below, showing which labels have been modified or deleted.



**Details of changes**

The options on this dialog are:

- **Synchronise** - synchronise with the iQ Works workspace immediately (see below)
- **Cancel** - close the dialog without synchronising.

If you select **Synchronise**, the MX OPC Configurator and iQ Works label lists will be synchronised, and all the changes will be processed. This is described in more detail in [Synchronising with iQ Works](#)<sup>[219]</sup>.

*Note:* Labels that have been added in a connected project are not shown in this dialog. They will be imported automatically on the next synchronisation with iQ Works.

*Note:* The notification dialog may sometimes be blank. This can happen if (for example) changes are made to [unsupported labels](#)<sup>[203]</sup> in a connected GX Works2 project.

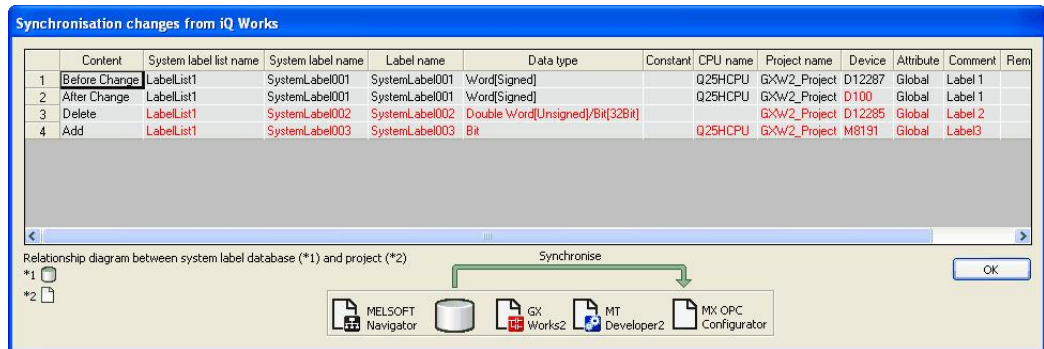
## 10.9 Synchronising with iQ Works

The label lists in MX OPC Configurator and the iQ Works workspace can be synchronised at any time using the menu option:

**File -> iQ Works Project Actions -> Synchronise**

Synchronisation will normally involve MX OPC Configurator accepting changes that have been made in the GX Works2 project - MX OPC Configurator is a reference project in iQ Works and cannot modify system labels directly. The only exception to this is when the [Add iQ Works label\(s\)](#)<sup>[211]</sup> feature has been used to add new labels. After the new labels have been synchronised once, they cannot be modified in MX OPC Configurator again and must be modified in GX Works2 instead.

If the synchronisation succeeds, a summary of the changes will be shown, as in the example below where the address of one label has been modified, another label has been deleted, and a third label has been added.



**Summary of changes**

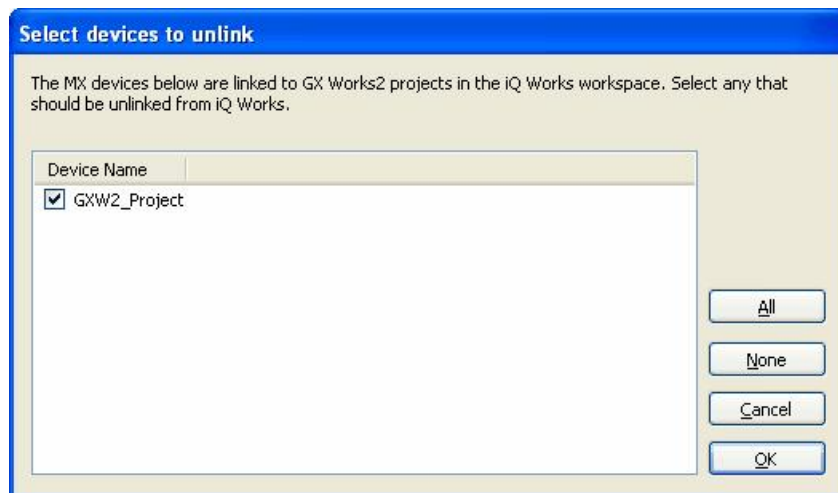
*Note:* It is possible that labels that are marked as 'Deleted' are still in the workspace, but are no longer [supported](#)<sup>[203]</sup> in MX OPC Configurator, or have been reassigned to a GX Works2 project that is not linked to an MX device. The same applies to labels marked as 'Added', which may have been modified so that they are now supported, or may have been assigned to a GX Works2 project linked to an MX device.

## 10.10 Unlinking devices from iQ Works

When an MX device no longer needs to be linked to a GX Works2 project in iQ Works, it can be *unlinked* to disconnect it from iQ Works. Other projects in the workspace are not affected.

To unlink one or more MX devices from iQ Works:

1. From the **File** menu, select **iQ Works project actions -> Unlink device**.
2. You will be shown a list of MX devices in the database which are linked to iQ Works, as in the figure below.



Select MX devices to unlink

To select a device to be unlinked, click on the box next to the device name so that it appears checked.

The controls on this dialog are:

- **All** - select all available devices
  - **None** - deselect all devices
  - **Cancel** - close the dialog without unlinking devices
  - **OK** - unlink the selected devices from iQ Works
3. When you click OK, the selected devices will be unlinked from iQ Works. They will no longer be updated on [synchronisation](#)<sup>[219]</sup>, but it will be possible to modify the label details in MX OPC Configurator.



### 10.11 Structure support

MX OPC Configurator can import labels defined in iQ Works as *structures*. These are structured data types, where the structure is defined once as a data type consisting of sub-elements of simple data types (Word, Bit etc.), and then many different variables can be created using the type definition. Each new variable will inherit all the sub-elements of the type. This is helpful to users in that less data entry is required when defining objects with compound data, reducing the risk of typing errors and helping to maintain a logical structure to the program.

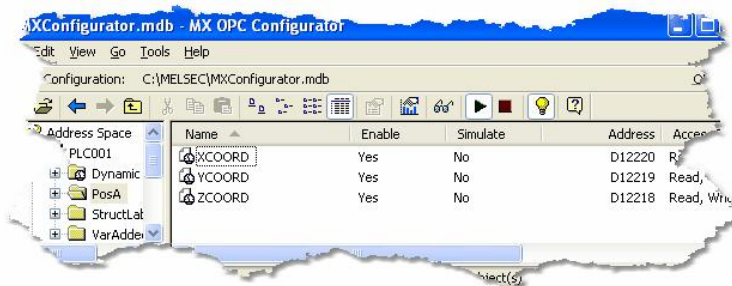
As an example, a 'position' structure could be defined with three integer sub-elements XCOORD, YCOORD and ZCOORD for the position coordinates, and then variables could be defined using the 'position' type - the structure definition in Navigator would look like the figure below. Each variable will then have an 'XCOORD', 'YCOORD', and 'ZCOORD' element to store a whole position. This saves time compared with the alternative of defining variables such as e.g. Pos1X, Pos1Y, Pos1Z, Pos2X, Pos2Y, Pos2Z etc. to store the same data.



Structure definition in Navigator

In MX OPC Configurator, variables of structure type are stored using the group / folder structure. The variable is shown as a top-level folder containing one variable for each of the sub-elements (from the structure definition). Each element may have its address assigned independently in GX Works2. The sub-elements can be read from an OPC client in the same way as any other label. Individual elements of a structure variable are referenced using dot notation e.g. StructVariable.StructElement. The sub-element part has the same 32-character limit as the variable name.

A variable 'PosA' defined with the above type would appear as in the picture below.



Structure variable in MX OPC Configurator

Arrays of structures are handled slightly differently. These are represented in MX OPC Configurator as a top-level folder for the structure label itself, with another folder for each of the array elements underneath it. Finally, under each of these array element folders will be one variable for each of the elements (from the structure definition). Storing the tags in this way means that OPC clients that do not support arrays or structures can still reference single elements within a structure label, as long as they have the ability to browse hierarchically for tags.

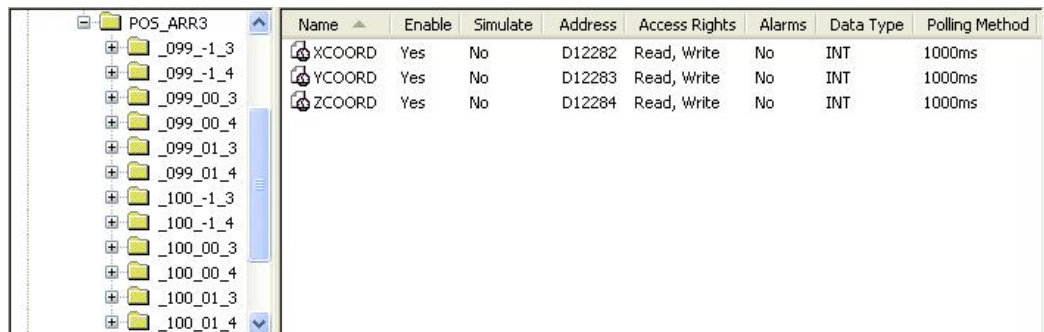
The name of the main structure folder matches the structure label name in GX Works2. For structure array labels, the structure folder will be given a name representing the array index e.g. '\_0' (for [0]), '\_-1' (for [-1]) or '\_0\_1' (for [0,1]). This format should be acceptable in most client applications' naming conventions. To ensure correct sorting of array elements, leading zeroes will be added to the indices to pad each index value up to the maximum width for that dimension, e.g. for an array with dimensions [0..1,0..100,0..50] an index value of [5,2,8] would be padded to Element\_5\_002\_08.

A simple example of an array of structure labels in MX OPC Configurator is shown below. This is a variable 'POS\_ARR' consisting of an array [0..1] using the POSITION type as above.



Array[0..1] of POSITION variables

A more complicated example follows. This is an array of variables using the POSITION type as above but with three dimensions and some negative index values.



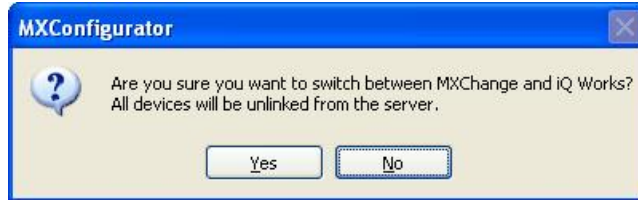
Array[99..100,-1..1, 3..4] of POSITION variables

*Note:* When defining addresses of structure array variables in GX Works2, do not use structure offset values if you intend to use the label with MX OPC Configurator - offsets are not supported.

Like other variables in linked projects in MX OPC Configurator, structure variables cannot be modified directly. An exception to this is if new structure variables are added as described in [Adding iQ Works labels](#)<sup>[211]</sup>. In this case, up until the next synchronisation with iQ Works, the addresses of (non-array) structure elements can be changed and the whole label can be deleted. However, it is not permitted to make changes that would be invalid based on the structure definition, such as deleting individual elements of the structure variable. The addresses of arrays of structures cannot be set in MX OPC Configurator, and should instead be set in GX Works2.

## 10.12 Switching between iQ Works and MXChange

A configurator project can be connected to iQ Works or MXChange, but not both at the same time. The warning message below will be shown before switching between MXChange and iQ Works.



**Switch between MXChange and iQ Works**

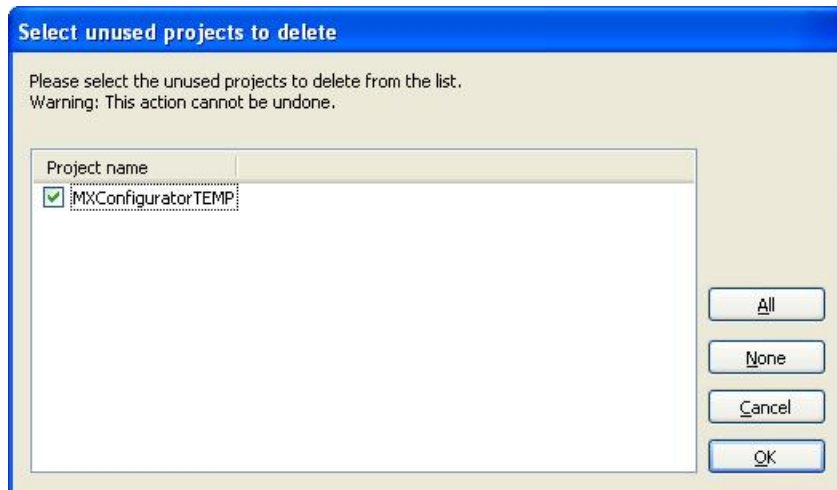
Select either:

- **Yes** - to switch to the other server type. This will reset all information about linked labels in the project, and cannot be undone.
- **No** - to cancel the connection, and leave the project connected to the same server.

## 10.13 Deleting unused projects

MX OPC Configurator projects do not appear in the project lists in Navigator. To delete unused MX OPC Configurator projects from the connected iQ Works workspace:

1. From the **File** menu, select **iQ Works workspace actions -> Delete unused projects**.
2. You will be shown a list of MX OPC Configurator projects in the connected iQ Works workspace (other than the current project), as in the figure below.



**Select unused projects to delete**

To select a project to delete, click on the box next to the device name so that it appears checked.

*Note:* Deleting a project is permanent and cannot be undone. Only projects that are no longer needed should be deleted.

The controls on this dialog are:

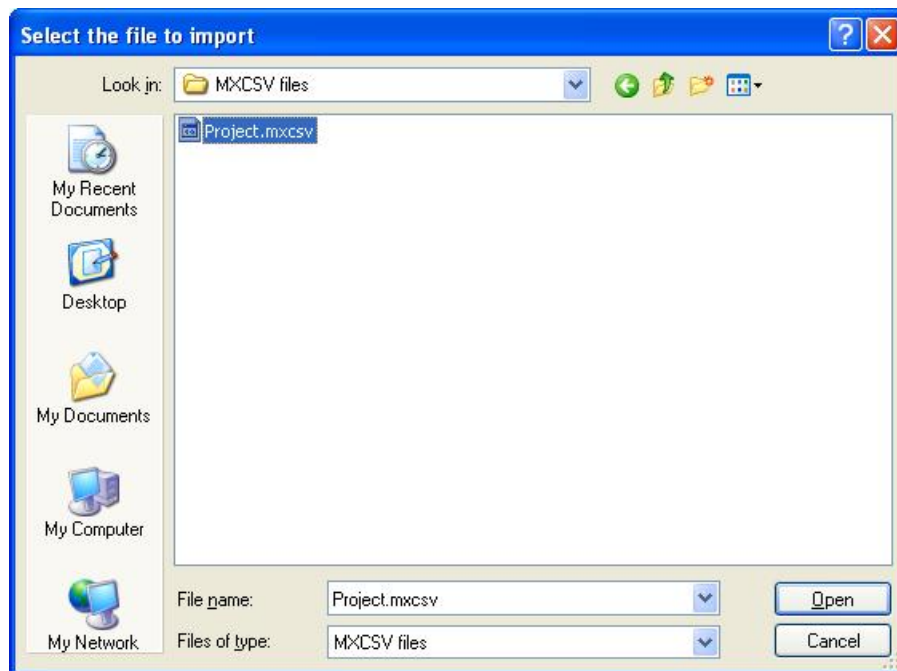
- **All** - select all available projects
  - **None** - deselect all projects
  - **Cancel** - close the dialog without deleting unused projects
  - **OK** - permanently delete the selected projects from the iQ Works workspace
3. When you click OK, the selected projects will be deleted from the iQ Works workspace.

## 10.14 Using MXCSV files

MX OPC Configurator supports connecting to iQ Works using settings provided by another application. These are supplied as a file with the extension '.MXCSV'.

The procedure to use these files is as follows:

1. From the **File** menu, select **iQ Works project Actions > Import CSV**.
2. In the dialog that appears, select the MXCSV file to import.



**Select MXCSV file to import**

3. If the MXCSV file name does not match the current MX OPC Configurator project, a new configuration database will be created.

You will first need to select the folder in which the new project should be stored.



Select folder for new configuration database

4. The project will be linked to the iQ Works workspace specified in the MXCSV file. If there any problems connecting to the specified workspace, the import will stop at this point.
5. In the dialog that appears, select the MXCSV file to import. You will be prompted to edit the [device properties](#) of each new device that will be linked to a GX Works2 project. Use the **Configuration** button to edit the connection settings before selecting **Save**.
6. The devices will be linked to iQ Works, and labels will be imported from each GX Works2 project.
7. To make sure that all details are up to date, you can select **iQ Works project Actions > Synchronise** from the **File** menu.

## 11 Transfer Setup

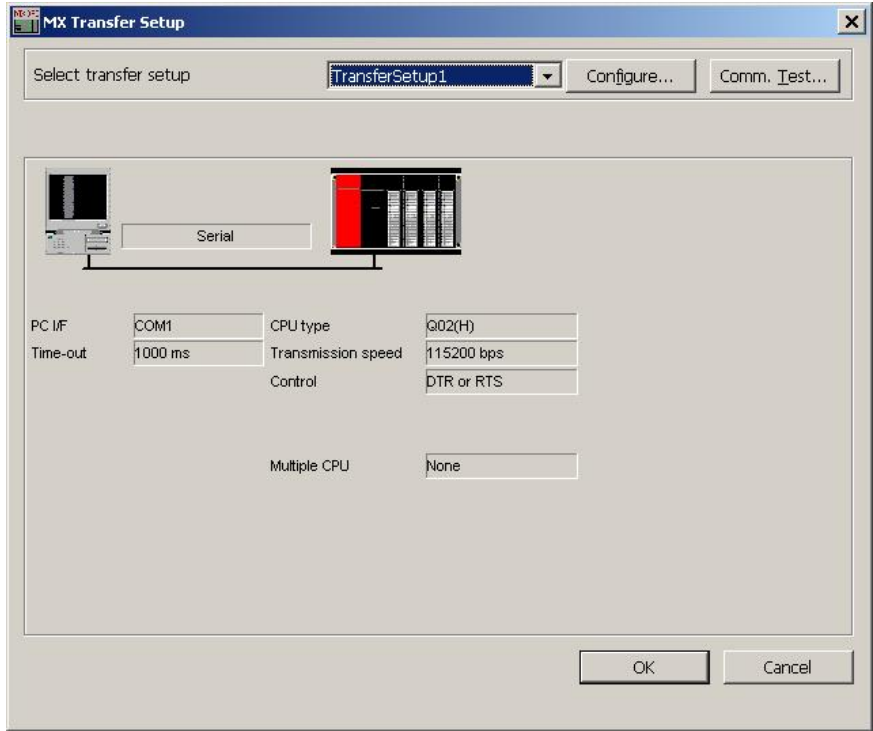
### 11.1 Configuring Transfer Setups

#### Transfer Setup Selection

The main dialog of the transfer setup provides access to the configured transfer setups. The user can


- add new setups
- remove existing setups
- modify a setup
- select a setup to be used
- test the communication with a selected setup

The network architecture and settings are displayed with bitmaps representing the network nodes. The parameters associated with a node are listed underneath the respective bitmap.



Name	Description	Choices / Setting range	Default
Select transfer setup	lists the names of existing transfer setups and the functions 'New <sup>[228]</sup> ' and 'Rename/Remove <sup>[228]</sup> '	1-32 characters	name of selected transfer setup
<a href="#">Configure<sup>[228]</sup></a>	edit the selected transfer setup	-	-
<a href="#">Comm. Test<sup>[229]</sup></a>	test the connection to the PLC	-	-

Pressing the **OK** button closes the dialog and uses the settings of the currently selected transfer setup to the device, for which the transfer setup has been opened. If the dialog is closed by pressing the 'Cancel' button, the transfer settings of the device remain unchanged.

	<p><b>Note:</b> please notice that devices using the same transfer setup name are not automatically updated. You need to select each device separately, open the transfer setup dialog and reassign the settings to the device.</p>
---	---

### New Transfer Setup

For defining a new transfer setup, the user selects the entry '<New...>' from the selection list.



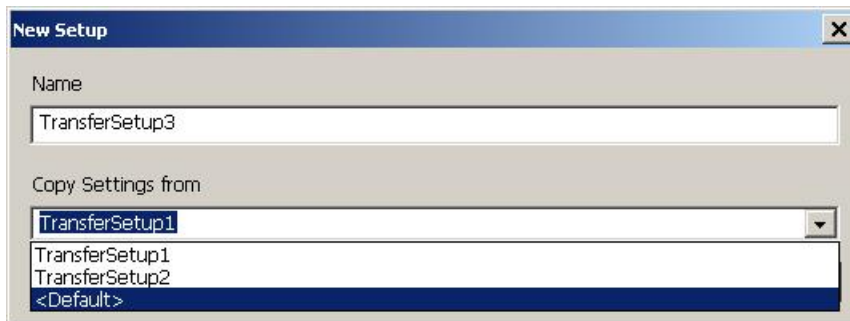
If the maximum number of transfer setups (default is max. 32 setups) has been reached, a message is displayed.



In this case the user must either delete a setup or modify an existing one.

If there is sufficient space for a new transfer setup, a dialog prompts the user to enter the name of the new transfer setup.

The user can select an existing transfer setup, from which the settings are copied, or select <Default>.



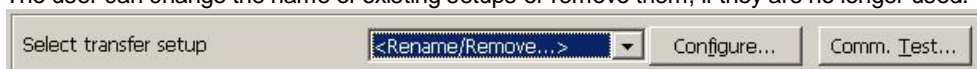
Name	Description	Choices / Setting range	Default
Name	name of new transfer setup	1-32 characters	TransferSetup<n>
Copy Settings from	uses the current settings of the selected setup for the new setup	- existing transfer setup names - <Default>	selected setup

The name assigned to the transfer setup must meet the following restrictions

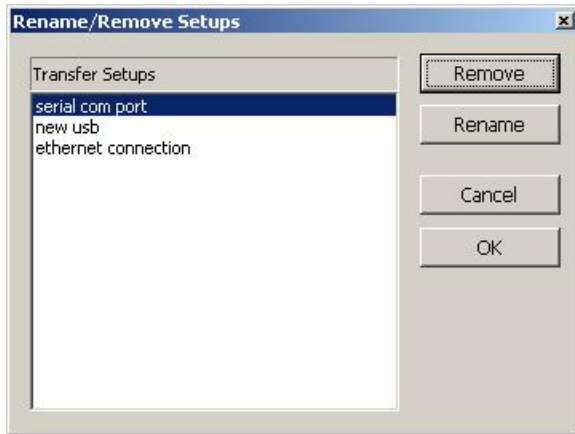
- must be unique
- must not be empty
- must not exceed 32 characters in length
- must not start or end with blanks (automatically removed)

### Rename / Remove Transfer Setup

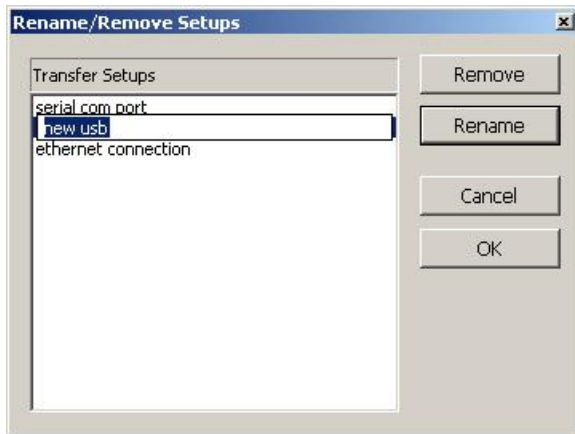
The user can change the name of existing setups or remove them, if they are no longer used.



When the entry '<Rename/Remove...>' is selected, a dialog is displayed, which lists the existing transfer setups.



Pressing the **'Remove'** button deletes the currently selected setup. Pressing the **'Rename'** button or double-clicking a setup name in the list opens an editor for the name within the list.



A setup name is rejected, if it does not meet the [requirements](#)<sup>[227]</sup> for setup names.

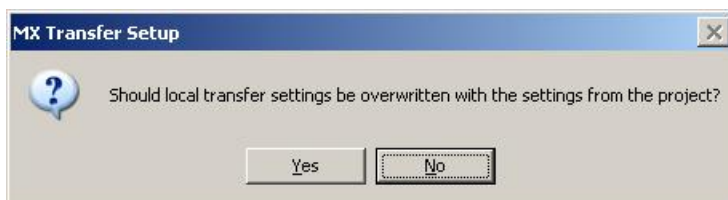
## Editing an Existing Transfer Setup

Pressing the **'Configure'** button opens the [transfer setup wizard](#)<sup>[230]</sup> for the selected transfer setup.

## Replace Transfer Setup Dialog

When the transfer setup dialog is opened, it imports the transfer setup of the currently selected device into the local transfer setup database. This database is kept in the Windows™ registry.

If the registry already contains a setup with the same name as the one of the selected device, the settings in the registry are compared with the settings used by the device. In case the settings differ, the user is asked, whether the local settings should be overwritten with the settings from the device.



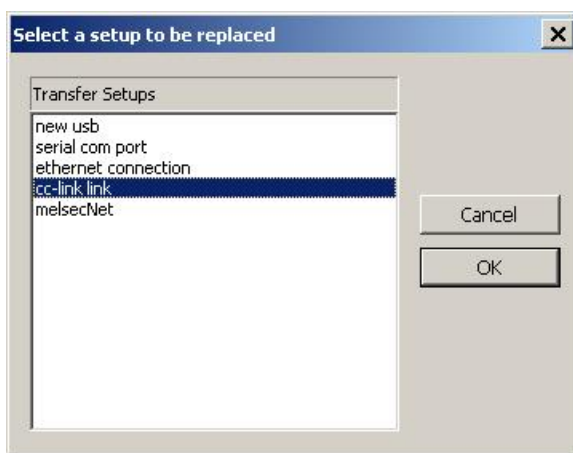
Selecting **'Yes'** causes the transfer settings from the device to overwrite the settings currently kept in the registry under the given transfer setup name. Otherwise the settings of the transfer setup in the registry remain unchanged.



If no setup with the name exists, a new setup is added to the database. In case there is insufficient space for a new setup, the following message is displayed.

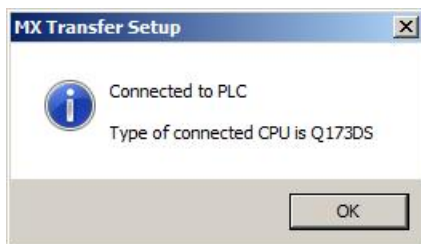


If the user selects 'Yes', a list of existing setups is displayed. The user can select a setup in the list, which is overwritten with the new settings.

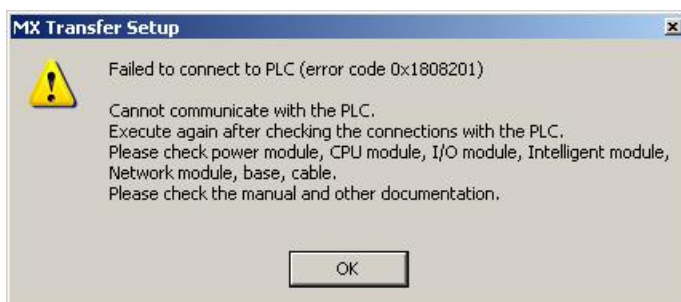


### Connection Test

The '**Comm. Test**' button uses the settings of the selected transfer setup to connect to the PLC. If the connection can be established, a success message is displayed.



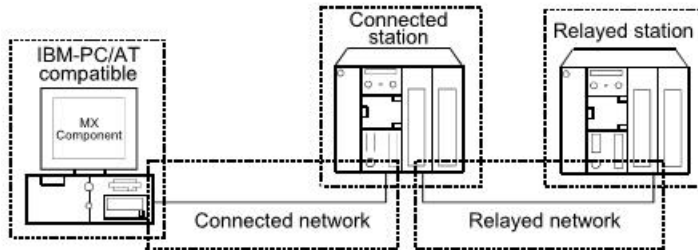
*If the connection to the PLC fails, the Easysocket error code is displayed, together with an explanatory message.*



## 11.2 Transfer Setup Wizard

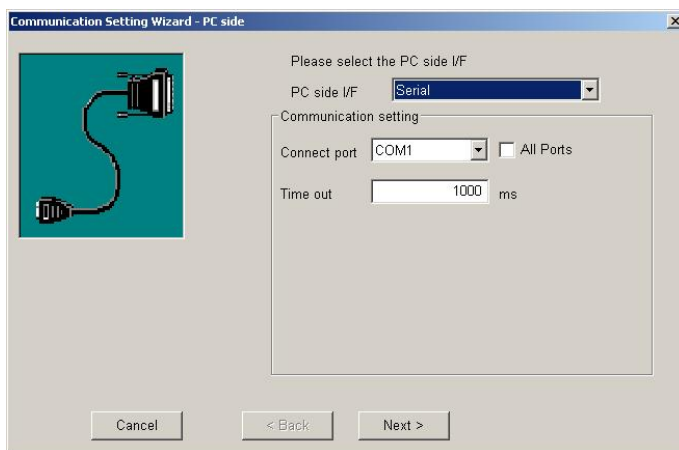
The transfer setup is used for the configuration of communication links from the PC to a PLC system.

The settings for a communication link between PC and PLC are entered in a set of dialogs. These are organized as a series of sequential pages. Moving between adjoining pages is done via the **'Back'** and **'Next'** buttons.



Screen Name	Description
Wizard 1)	set the connected network between the IBM-PC/AT compatible and connected station (PLC CPU and module).
Wizard 2)	set the connected station (PLC CPU and module).
Wizard 3)	set the relayed network between the connected station (PLC CPU and module) and relayed station (PLC CPU and module).
Wizard 4)	set the relayed station PLC CPU.
Wizard 5)	add a comment

Start the communication setting wizard.



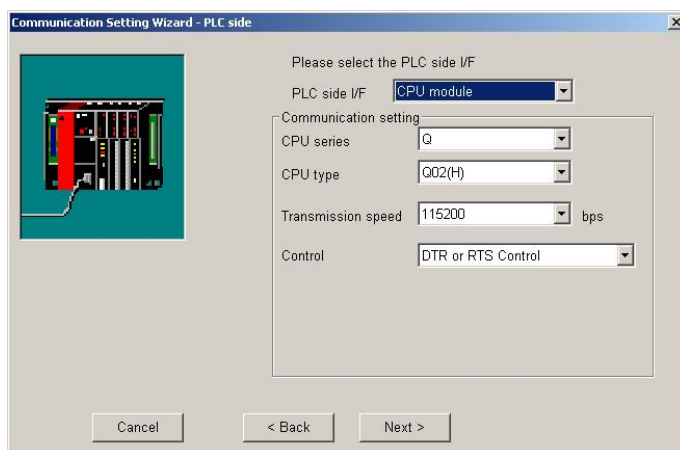
The items shown in the 'Communication setting' box depend on the selected PC side interface.

Choose the 'PC side I/F' to communicate with. The available interface types are listed below:

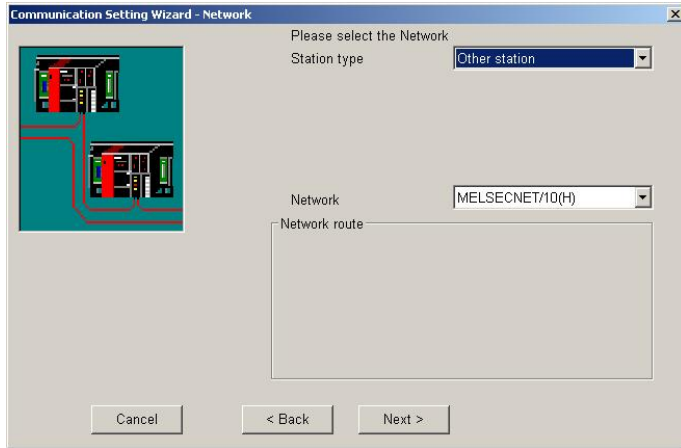
'PC side I/F' Selection	Communication Type
<a href="#">USB</a> <sup>[233]</sup>	USB communication
<a href="#">USB (via GOT)</a> <sup>[233]</sup>	USB communication via GOT

'PC side I/F' Selection	Communication Type
<a href="#">Serial</a> <sup>[234]</sup>	Computer link communication, CPU COM communication, CC-Link G4 communication
<a href="#">Serial (via GOT)</a> <sup>[235]</sup>	Serial communication via GOT
<a href="#">Ethernet board</a> <sup>[236]</sup>	Ethernet communication
<a href="#">Ethernet board (via GOT)</a> <sup>[248]</sup>	Ethernet communication via GOT
<a href="#">Modem</a> <sup>[248]</sup>	Modem communication
<a href="#">CC-Link IE Control board</a> <sup>[258]</sup>	CC-Link IE communication
<a href="#">MELSECNET/H board</a> <sup>[261]</sup>	MELSECNET/H communication
<a href="#">MELSECNET/10 board</a> <sup>[264]</sup>	MELSECNET/10 communication
<a href="#">CC-Link IE Field board</a> <sup>[267]</sup>	CC-Link IE Field communication
<a href="#">CC-Link board</a> <sup>[269]</sup>	CC-Link communication
<a href="#">Q Series Bus</a> <sup>[271]</sup>	Q Series bus communication
<a href="#">GX Simulator2</a> <sup>[274]</sup>	GX Simulator2 communication
<a href="#">GX Simulator</a> <sup>[273]</sup>	GX Simulator communication
<a href="#">MT Simulator2</a> <sup>[276]</sup>	MT Simulator2 communication
<a href="#">CPU board</a> <sup>[278]</sup>	CPU board communication

After setting the parameters continue by pressing the **'Next'** button.

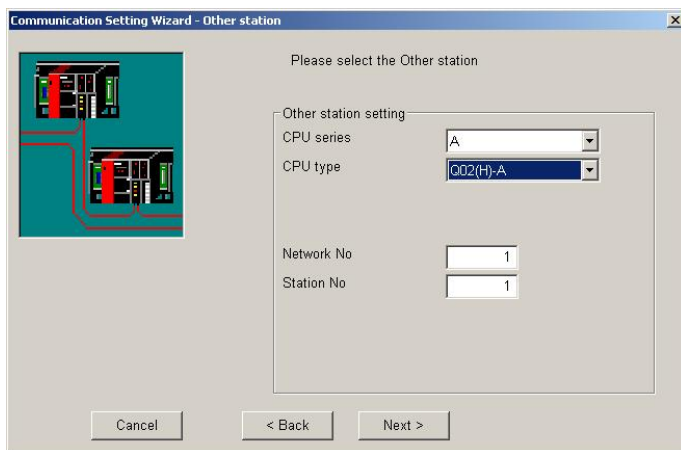


The contents of the 'PLC side I/F' page depend on the selected PC side interface. Set all available parameters and click the **Next** button.



The contents of the 'Network' page depend on the selections in the previous pages.  
Set all available parameters and click the **Next** button.


**Note:** when 'Modem' has been selected as 'PC side I/F', the line setting screen appears next. For details of the line setting screen refer to ['Modem Line Settings'](#)<sup>[249]</sup>.



The contents of the 'Other station' page depend on the selections in the previous pages.  
Set all available setting items and click the **Next** button.



You can enter an optional comment of up to 32 characters for the connection.  
Pressing the **Finish** button stores the current settings and closes the wizard.

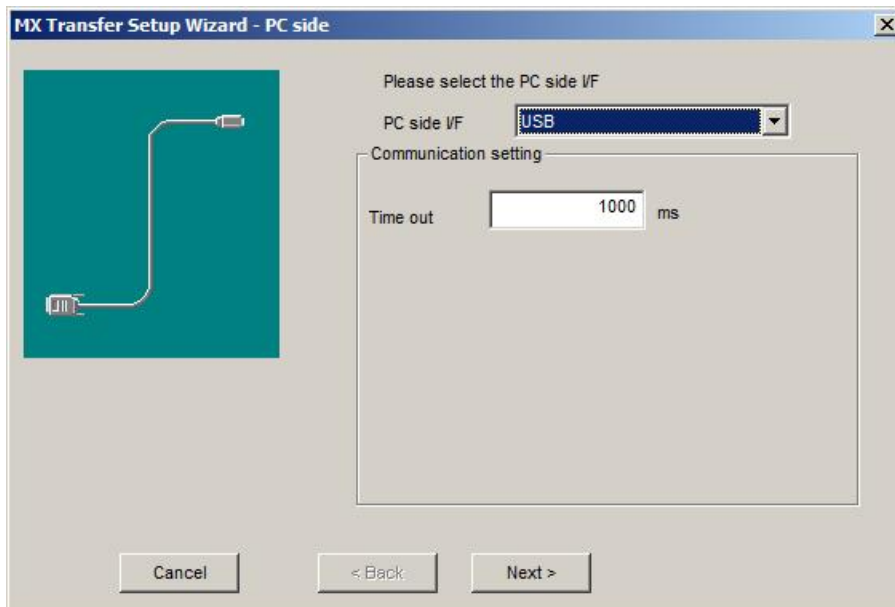
	<p><b>Note:</b></p> <p>The enclosed sample programs should serve as examples when writing the user program. They must be modified to match the specific application requirements. The use of code taken from the sample programs is done at the customer's own risk.</p>
---	--

### 11.2.1 PC Universal Serial Bus (USB) Interface

Configure a connection from an USB port of the PC to a PLC.

#### *PC side USB parameters*

Parameter	Values	Description
Time out	1 - 2147483647	communication timeout in milliseconds



For a description of the editable parameters on the page see [PC side USB parameters](#)<sup>[233]</sup>.

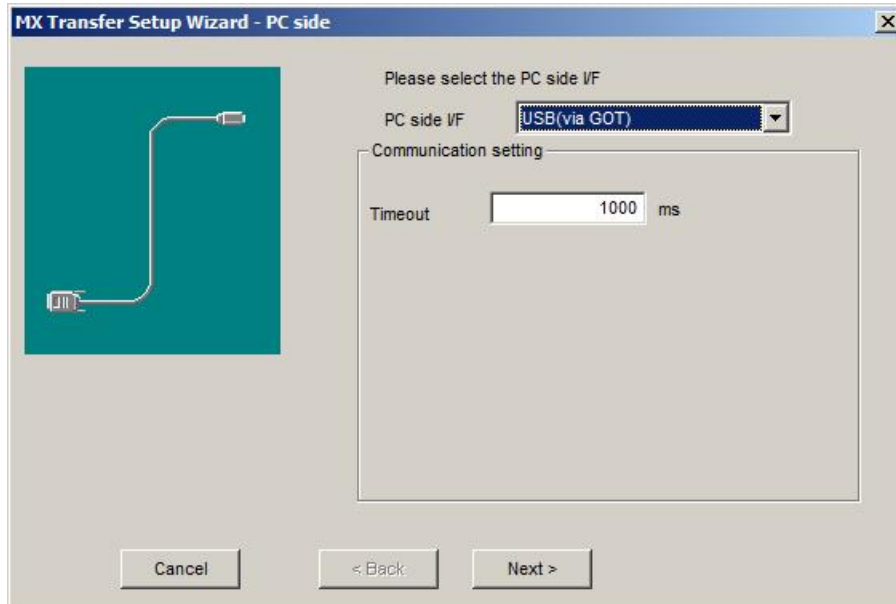
The connected interface on the PLC side is a [CPU module](#)<sup>[278]</sup>.

### 11.2.2 PC Universal Serial Bus (USB) Interface via GOT

Configure a connection from an USB port of the PC to a GOT and from there to a PLC.

### PC side USB parameters

Parameter	Values	Description
Time out	1 - 2147483647	communication timeout in milliseconds



For a description of the editable parameters on the page see [PC side USB parameters](#)<sup>[234]</sup>.

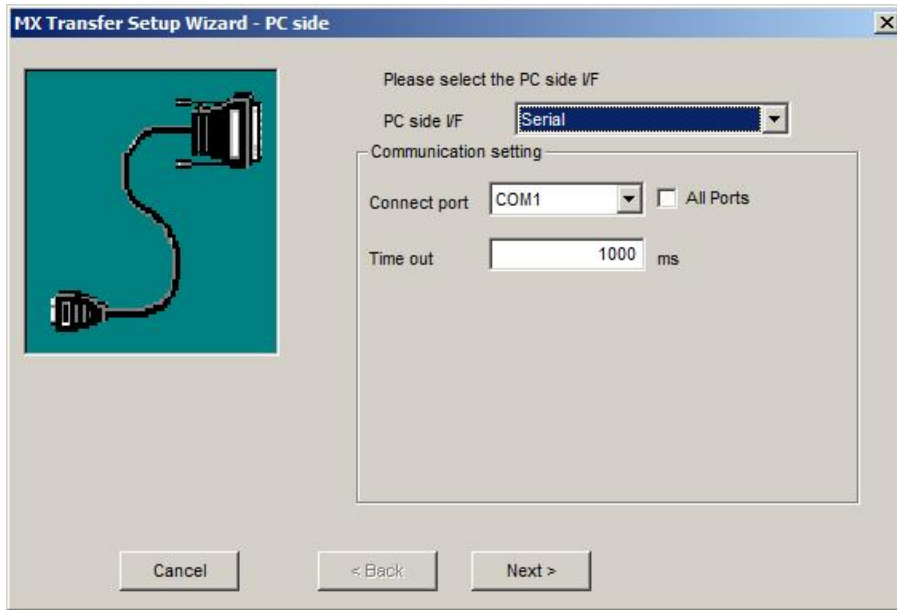
Pressing the <Next> button opens a [page](#)<sup>[346]</sup> for selecting the connected CPU type.

### 11.2.3 PC RS232 Serial Interface

Configure a serial connection to a PLC.

#### PC side serial parameters

Parameter	Values	Description
All Ports		if cleared, only one of the serial interfaces actually installed in the PC can be selected. if checked, any COM port can be selected. This may be useful when defining a configuration for a different PC.
Connect port	'All cleared Ports'	COM1 - COMn
	'All checked Ports'	COM1 COM256
Time out	1 2147483647	- communication timeout in milliseconds



For a description of the editable parameters on the page see [PC side serial parameters](#)<sup>[234]</sup>.  
 The connected interface on the PLC side can be

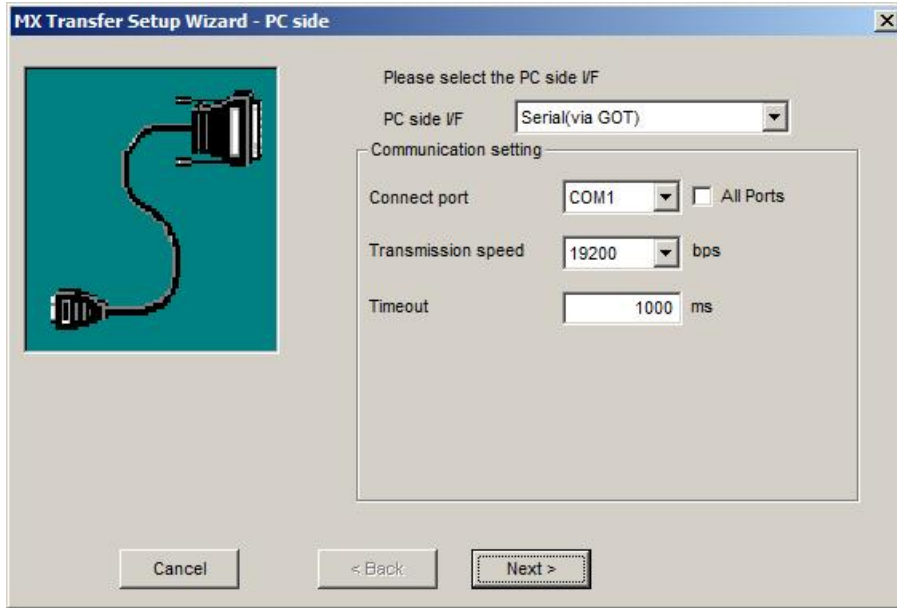
- a [CPU module](#)<sup>[278]</sup> (either [RS422](#)<sup>[279]</sup> or [USB](#)<sup>[283]</sup> interface)
- a [C24 module](#)<sup>[285]</sup>
- a [G4 module](#)<sup>[321]</sup>
- an [FX extended port](#)<sup>[296]</sup>

### 11.2.4 PC RS232 Serial Interface via GOT

Configure a serial connection to a GOT and from there to a PLC.

#### *PC side serial parameters*

Parameter		Values	Description
All Ports			if cleared, only one of the serial interfaces actually installed in the PC can be selected. if checked, any COM port can be selected. This may be useful when defining a configuration for a different PC.
Connect port	'All cleared Ports'	COM1 - COMn	selects the serial interface in the PC from a list of serial interfaces found on the PC
	'All checked Ports'	COM1 - COM256	selects the serial interface in the PC from a list of possible serial interfaces
Transmission speed		9600, 19200, 38400, 57600, 115200	baud rate of the serial connection
Timeout		1 - 2147483647	communication timeout in milliseconds



For a description of the editable parameters on the page see [PC side serial parameters](#)<sup>[235]</sup>. Pressing the <Next> button opens a [page](#)<sup>[346]</sup> for selecting the connected CPU type.

### 11.2.5 PC Ethernet Board


Configure the PC side ethernet interface

#### *PC side Ethernet parameters*

Parameter	Values	Description
Connect module	<a href="#">AJ71E71</a> <sup>[238]</sup> , <a href="#">AJ71QE71</a> <sup>[240]</sup> , <a href="#">QJ71E71</a> <sup>[243]</sup> , <a href="#">GOT</a> <sup>[245]</sup> , <a href="#">FX-ENET(-ADP)</a> <sup>[246]</sup> , <a href="#">FX3U-ENET-ADP</a> <sup>[246]</sup> , <a href="#">CPU module</a> <sup>[247]</sup> , <a href="#">CC IE Field Ethernet adapter</a> <sup>[247]</sup>	type of PLC ethernet module
Protocol	UDP, TCP	connection-less (UDP) or connection-oriented (TCP) protocol <b>not for FX-ENET(-ADP) modules</b>  Precautions for use of TCP protocol (a) Provide an interval longer than the sequence scan time of the Ethernet module loaded station from when the Open method is executed until the Close method is executed. (b) Provide an interval of at least 500ms from when the Close method is executed until the Open method is executed again.
Packet type	ASCII, Binary	selects ASCII or binary packet type for communication <b>only for AJ71E71 and AJ71QE71 modules</b>



Parameter	Values	Description
Network No	1 - 239	network number of the E71 ethernet module, the PC is connected to. The E71 network number is configured in the 'Network Parameters' of the PLC settings. <b>only for AJ71QE71(UDP) and QJ71E71 modules</b>
Station No.	1 - 64	station number of the PC side interface. This number can be freely chosen, but must be unique within the network specified by the network number. <b>only for AJ71QE71(UDP) and QJ71E71 modules</b>
Port No.	0 - 65535	free UDP port number on the PC side for data sent by the PLC. Port numbers below 1025 should not be used. <b>only for UDP protocol and for GOT module</b>
Time out	1 - 2147483647	communication timeout in milliseconds




**Note:**

When connecting from several MELSOFT applications to the same E71 module, make the following settings:

- (a) set the protocol of the communication setting wizard screen to 'UDP'.
- (b) set "SW2" of the communications setting switches of the E71 module to OFF (binary).

The following restrictions apply to simultaneous connections from multiple PCs to the same E71 module using the TCP/IP protocol:


- for Q series-compatible E71 modules (except QJ71E71-100) the first five digits of the serial number must be '02122' or later and the function version must at least be 'B'
- select 'MELSOFT connection' in the Ethernet parameter 'open system' (via the network parameters in GID/GD)



**Note:**

Connecting to the PLC CPU through Ethernet depends on the correct PLC CPU status:

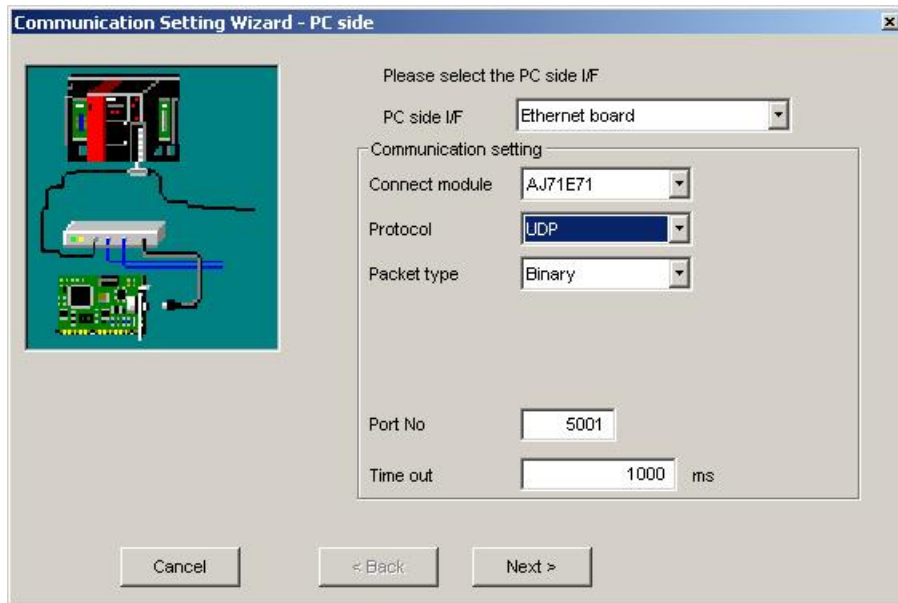
- 1) **TCP** selected: the target PLC CPU must be in RUN mode
- 2) **UDP** selected: the target PLC CPU must have been in RUN mode once



**Note:**

When a QnA-, AnU-, Q- (A mode) or motion controller CPU (via E71) are accessed, the device range is equivalent to that of an AnA CPU.

### AJ71E71



For a description of the editable parameters on the page see [PC side Ethernet parameters](#)<sup>[236]</sup>.

Pressing the <Next> button opens a [page](#)<sup>[308]</sup> for entering the address of the PLC ethernet module.

### TCP connections

Connected Station CPU			Relayed Network	Relayed Station CPU						
QCPU (A mode)	QnA CPU	A CPU *1		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FX CPU	Q Motion	LCPU
	*2		MELSECNET/H	x	x	x	x	x	x	x
			MELSECNET/10	x	o	*2	o	x	x	x
			MELSECNET(II)	x	o	*2	o	x	x	x
			Ethernet	x	x	x	x	x	x	x
			Computer link	x	x	x	x	x	x	x
			CC-Link	x	x	x	x	x	x	x
			CC-Link IE	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

\*2 : Operates as the one equivalent to AnACPU

### UDP connections

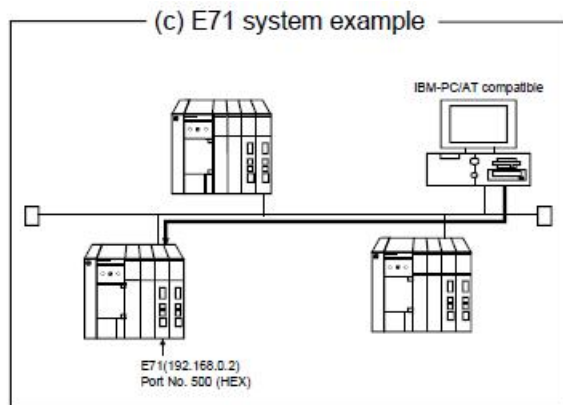
Connected Station CPU			Relayed Network	Relayed Station CPU						
QCPU (A mode)	QnA CPU	ACPU *1		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FX CPU	Q Motion	LCPU
	*2		MELSECNET/H	x	x	x	x	x	x	x
			MELSECNET/10	x	o	*2	o	x	x	x
			MELSECNET(II)	x	o	*2	o	x	x	x
			Ethernet	x	x	x	x	x	x	x
			Computer link	x	x	x	x	x	x	x
			CC-Link	x	x	x	x	x	x	x
			CC-Link IE	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

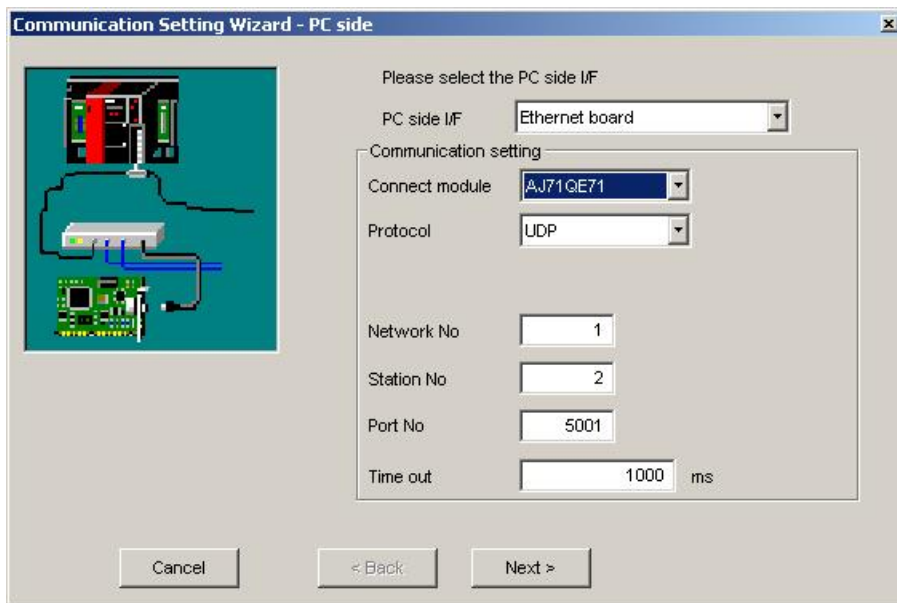
\*2 : Operates as the one equivalent to AnACPU



Switch *1		Setting			
		TCP/IP		UDP/IP	
		When ASCII packet is used	When binary packet is used	When ASCII packet is used	When binary packet is used
Operation mode setting switch		0 (online)			
Communications condition setting switches	Line processing selection for TCP time-out error	OFF			
	Data code setting	ON (ASCII code)	OFF (binary code)	ON (ASCII code)	OFF (binary code)
	CPU communications timing setting)	ON			
	Initial timing setting	OFF			

\*1: For switch numbers, refer to the E71 module manual.

### AJ71QE71



For a description of the editable parameters on the page see [PC side Ethernet parameters](#)<sup>[236]</sup>.

Pressing the <Next> button opens a [page](#)<sup>[309]</sup> for entering the address of the PLC ethernet module.

## TCP connections

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FX CPU	Q Motion	LCPU
QnACPU	MELSECNET/H	x	x	x	x	x	x	x
	MELSECNET/10	x	x	o	x	x	x	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	x	x	o	x	x	x	x
	C24	x	x	o	x	x	x	x
	CC-Link	x	x	x	x	x	x	x
	CC-Link IE	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

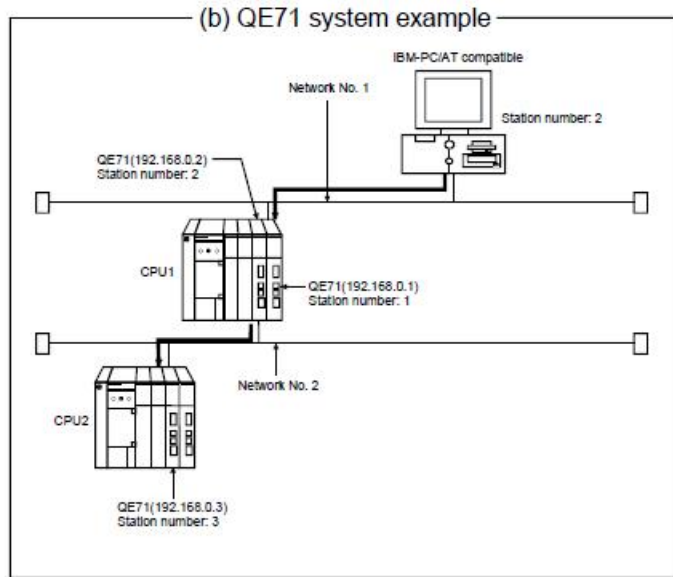
## UDP connections

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FX CPU	Q Motion	LCPU
QnACPU	MELSECNET/H	x	x	x	x	x	x	x
	MELSECNET/10	x	x	o	x	x	x	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	x	x	o	x	x	x	x
	C24	x	x	o	x	x	x	x
	CC-Link	x	x	x	x	x	x	x
	CC-Link IE	x	x	x	x	x	x	x

o: Accessible

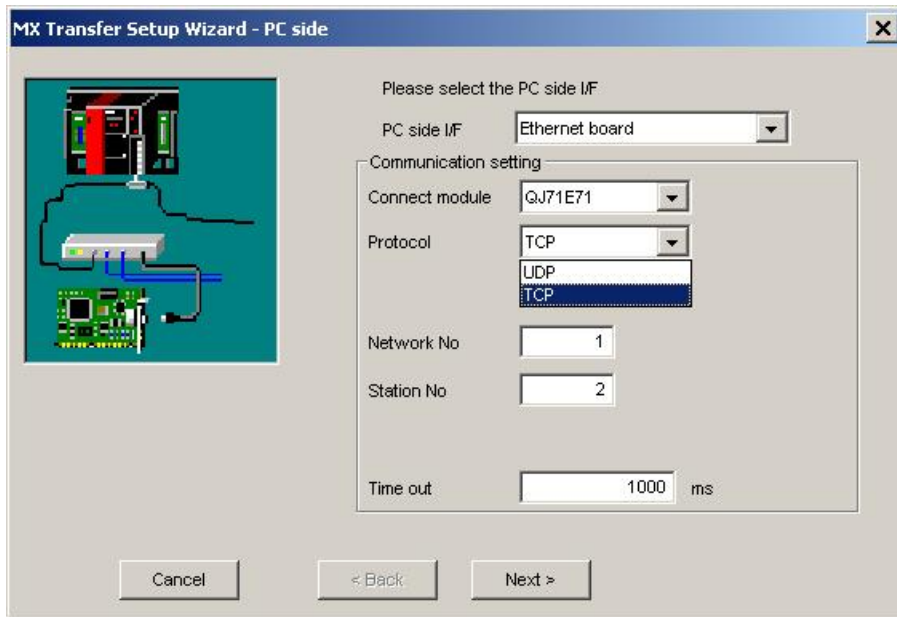
x: Inaccessible

\*1 : Including motion controller CPU



Switch (Switch Number)		Setting		
		TCP / IP		UDP / IP
		When ASCII packet is used	When binary packet is used	When binary packet is used
Operation mode setting switch		0 (online)		
Communications condition setting switches	Line processing selection for TCP time-out error (SW1)	OFF		
	Data code setting (SW2)	ON (ASCII code)	OFF (binary code)	As set by user
	Automatic start mode setting (SW3)	OFF		ON
	— (SW4 to SW6)	All OFF		
	CPU communications timing setting (SW7)	ON		
	Initial timing setting (SW8)	OFF		

### QJ71E71



For a description of the editable parameters on the page see [PC side Ethernet parameters](#)<sup>[236]</sup>.

Pressing the <Next> button opens a [page](#)<sup>[310]</sup> for entering the address of the PLC ethernet module.

### TCP connections

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FX CPU	Q Motion	LCPU
	MELSECNET/H	o	x	x	x	o	o	x
	MELSECNET/10	o	o	o	o	x	o	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	o	x	o	x	x	o	x
	C24	*2	x	x	x	x	o	o
	CC-Link	o	o	o	o	x	o	o
	CC-Link IE Field	o	x	x	x	x	x	o
	CC-Link IE Control	o	x	x	x	x	o	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

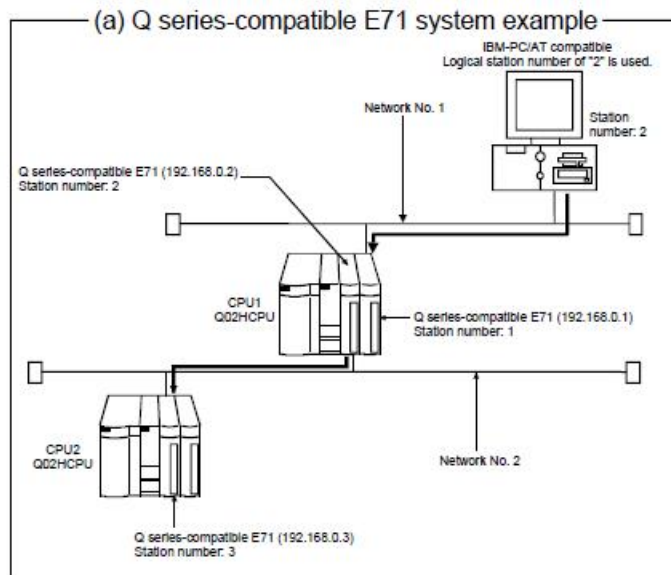
\*2 : For the Redundant PLC, access is disabled

\*3 : partially supported, depends on CPU type

### UDP connections

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FX CPU	Q Motion	LCPU
	MELSECNET/H	o	x	x	x	o	o	x
	MELSECNET/10	o	o	o	o	x	o	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	o	x	o	x	x	o	x
	C24	*2	o	x	x	x	o	o
	CC-Link	o	o	o	o	x	o	o
	CC-Link IE Field	o	x	x	x	x	x	o
	CC-Link IE Control	o	x	x	x	x	o	x

o: Accessible  
 x: Inaccessible  
 \*1 : Including motion controller CPU  
 \*2 : For the Redundant PLC, access is disabled  
 \*3 : partially supported, depends on CPU type

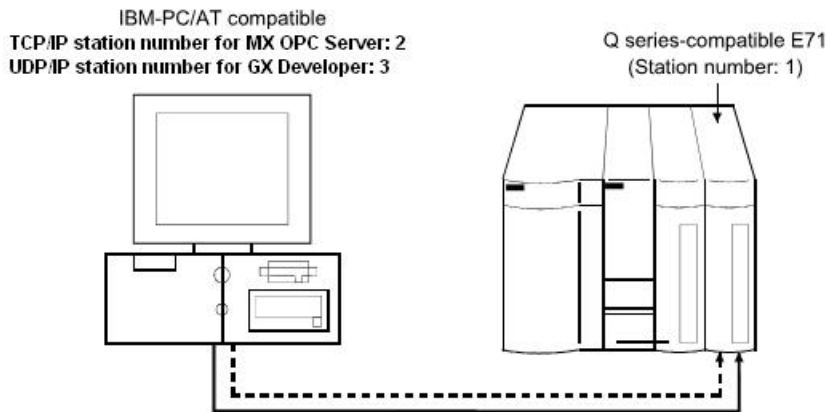


### Simultaneous use of different protocol types

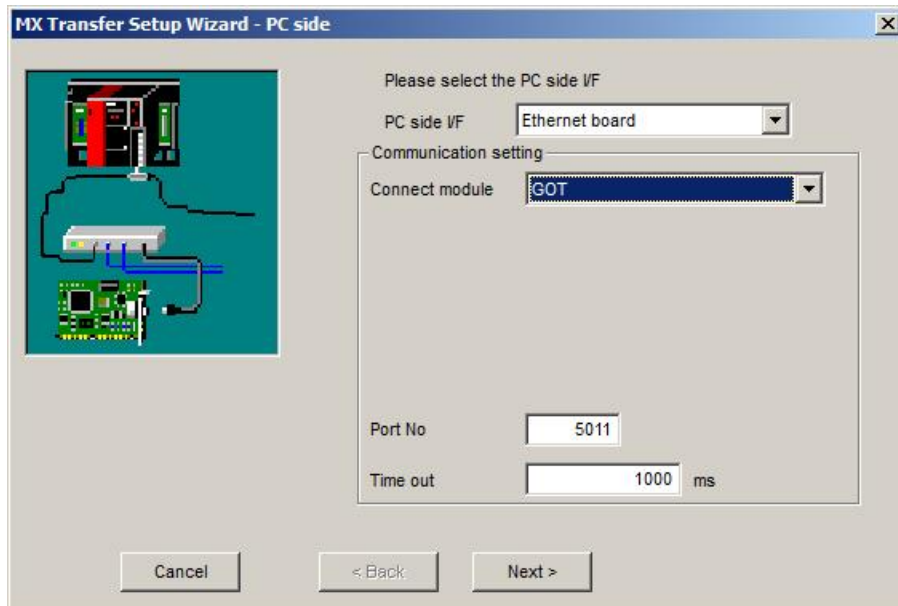
When two different communication systems (protocols) are used to access the same Q series-compatible E71 with a serial number before 05051 from one PC, different station numbers must be set for TCP/IP and for UDP/IP.

(Example) When MX OPC Server uses TCP and GX Developer uses UDP



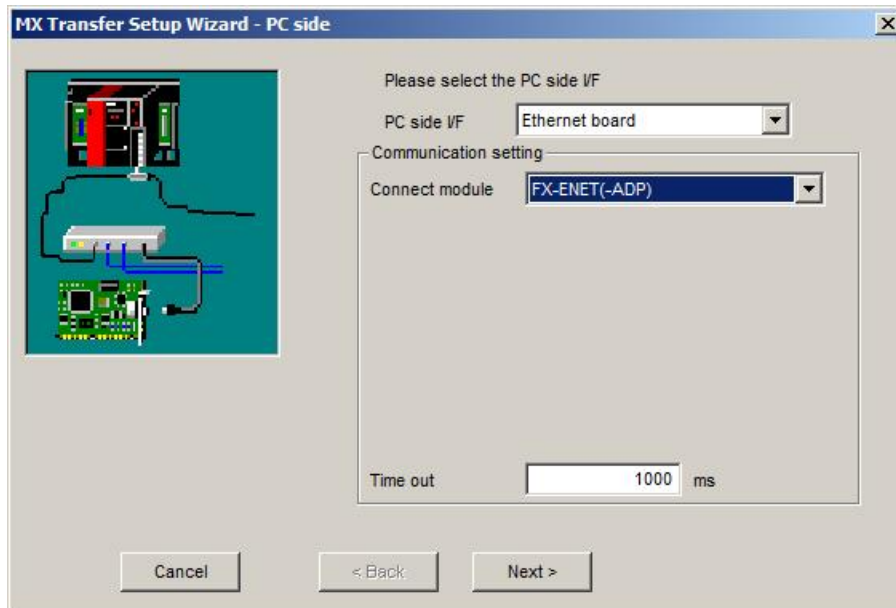


**GOT**



For a description of the editable parameters on the page see [PC side Ethernet parameters](#)<sup>[236]</sup>. Pressing the **<Next>** button opens a [page](#)<sup>[312]</sup> for entering the address of the GOT ethernet module.

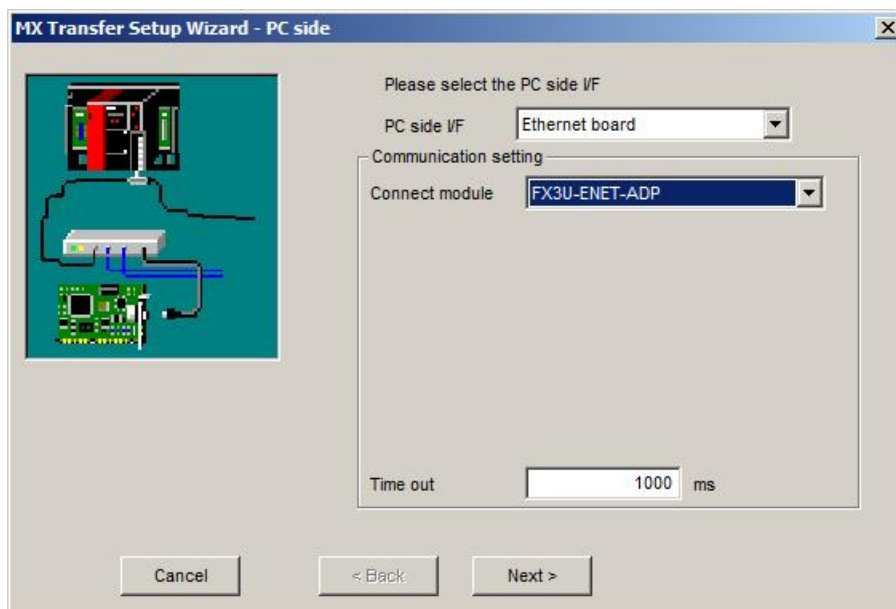
## FX Ethernet Module



For a description of the editable parameters on the page see [PC side Ethernet parameters](#)<sup>[236]</sup>.

Pressing the <Next> button opens a [page](#)<sup>[312]</sup> for entering the address of the PLC ethernet module.

## FX3U Ethernet Module

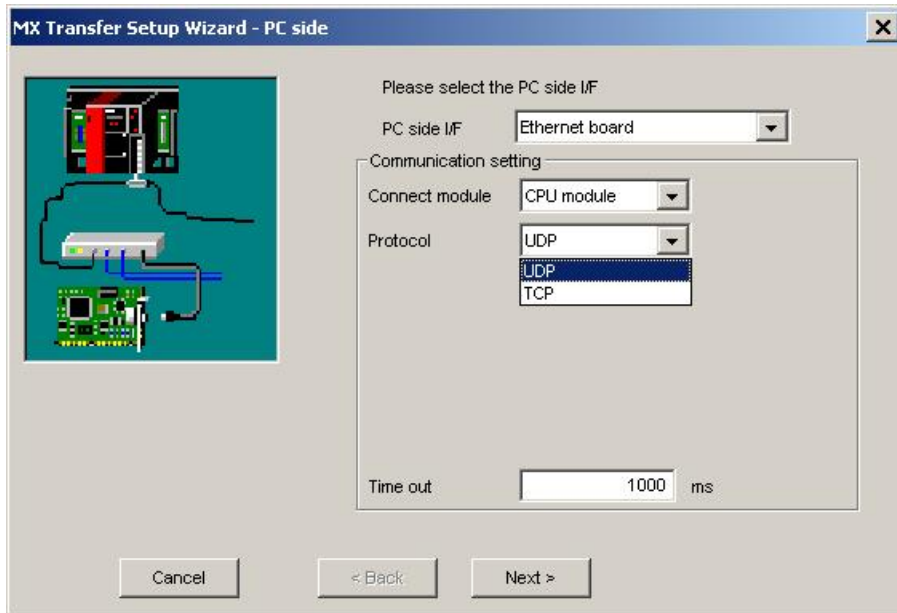


For a description of the editable parameters on the page see [PC side Ethernet parameters](#)<sup>[236]</sup>.

Pressing the <Next> button opens a [page](#)<sup>[316]</sup> for entering the address of the PLC ethernet module.

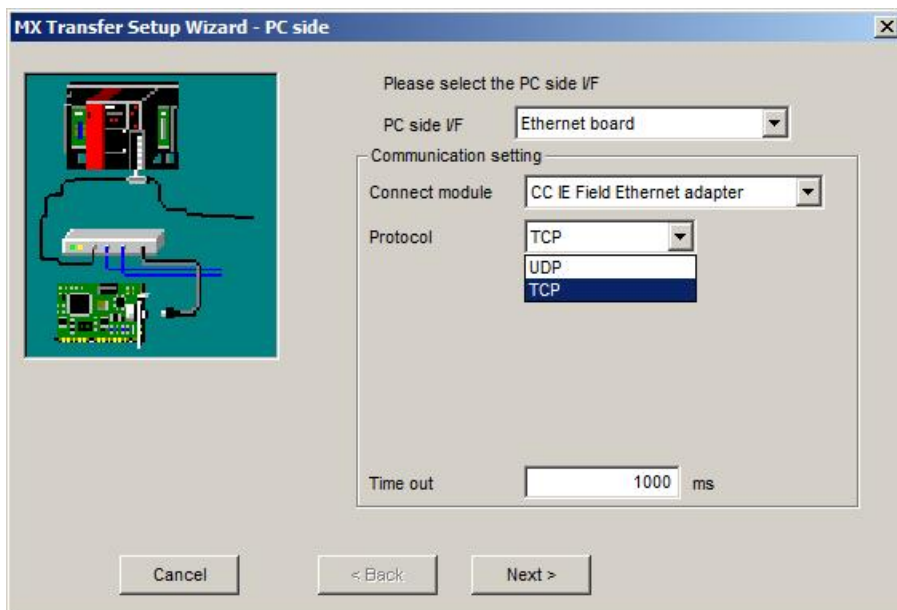
### CPU Module

'CPU module' is selected when connecting to a Q-series, L-series or Q Motion series CPU with Ethernet interface.



Pressing the **<Next>** button opens a [page<sup>317</sup>](#) to search for the PLC CPU or to manually enter its IP address.

### CC IE Field Ethernet Adapter



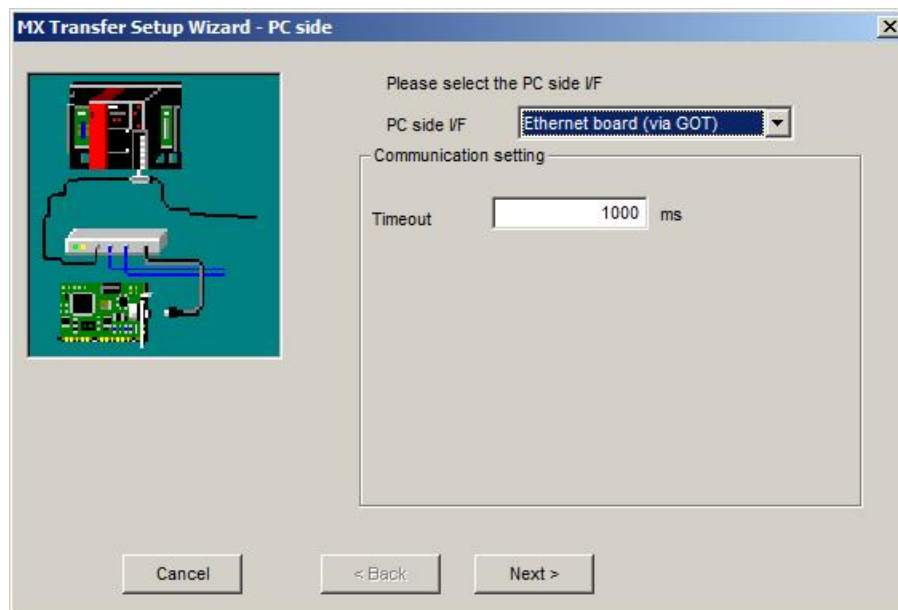
Pressing the **<Next>** button opens a [page<sup>321</sup>](#) for entering the address of the CC-Link IE Field ethernet adapter.

### 11.2.6 PC Ethernet Board via GOT

Configure an Ethernet connection to a GOT and from there to a PLC.

#### *PC side Ethernet parameters*

Parameter	Values	Description
Time out	1 - 2147483647	communication timeout in milliseconds




For a description of the editable parameters on the page see [PC side Ethernet parameters](#)<sup>[248]</sup>.

Pressing the <Next> button opens a [page](#)<sup>[348]</sup> for selecting the connected CPU type.

### 11.2.7 PC Modem


Configure a dial-up connection via modem.

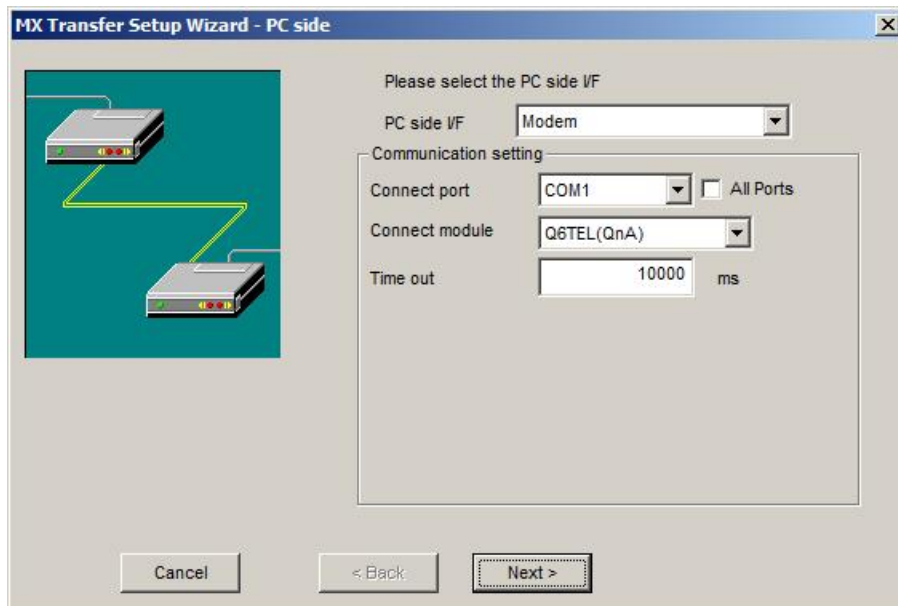
	<p><b>Note:</b></p> <p>It is recommended to configure and test the connection in GX Developer or GX IEC Developer before using this wizard.</p>
---	---

#### *PC side modem parameters*

Parameter	Values	Description
All Ports		<p>if cleared, only one of the serial interfaces actually installed in the PC can be selected</p> <p>if checked, any COM port can be selected. This may be useful when defining a configuration to a different PC</p>

Parameter	Values	Description
Connect port	'All Ports' flag cleared	selects the serial interface in the PC from a list of serial interfaces found on the PC
	'All Ports' flag checked	selects the serial interface in the PC from a list of possible serial interfaces
Connect module	Q6TEL(QnA), FXCPU, AJ71QC24N, QJ71C24, QJ71CMO, LJ71C24	type of remote PLC module
Time out	1 - 2147483647	communication timeout in milliseconds

 **Note:** Simultaneous modem communication by several MELSOFT applications is not supported. Communication errors, a disconnection of the telephone line or similar problems may occur.




For a description of the editable parameters on the page see [PC side modem parameters](#)<sup>[248]</sup>.

Pressing the <Next> button opens a [page](#)<sup>[249]</sup> for configuring the modem connection (e.g. telephone number etc.).

### 11.2.7.1 Modem Line Settings

Enter the parameters for the modem connection to a remote PLC including the phone number and any AT commands to configure the local modem.

 **Note:**

- (a) Do not use a call-waiting phone line.
- (b) Do not connect the line to master/slave phones.

On a call-waiting phone line, data corruption, telephone line disconnection or similar may occur due to interrupt reading sounds.

If the slave phone picks up while the telephone line is connecting to the master/slave phones, the telephone line may be disconnected.

(c) Use an analog 2 wire type telephone line.

When using a digital line, use a terminal adaptor.

When the telephone line is of 4 wire type, the line may not be connected depending on the wiring type of the modular jack.


For the 4 wire type conduct connection tests in advance to check for connection.

**Note:**

(a) Modem for radio communication using a cellular phone

Although the modem name is different depending on the maker, the modem is generically referred to as the cellular phone communication unit in this manual. Select the model of the cellular phone communication unit according to the cellular phone used.

For details, contact the company of your cellular phone.

 (b) Cellular phone without auto answer function


For a cellular phone without auto answer function, use a cellular phone communication unit that has the ANS/ORG/TEL select switch. If the cellular phone communication unit does not have the ANS/ORG/TEL select switch, it is impossible to connect the line. The line connection procedure is different depending on the cellular phone company and cellular phone model.

For details, contact the maker of your cellular phone.

### Connect line dialog

Set the line connection system, telephone line, AT command, etc.

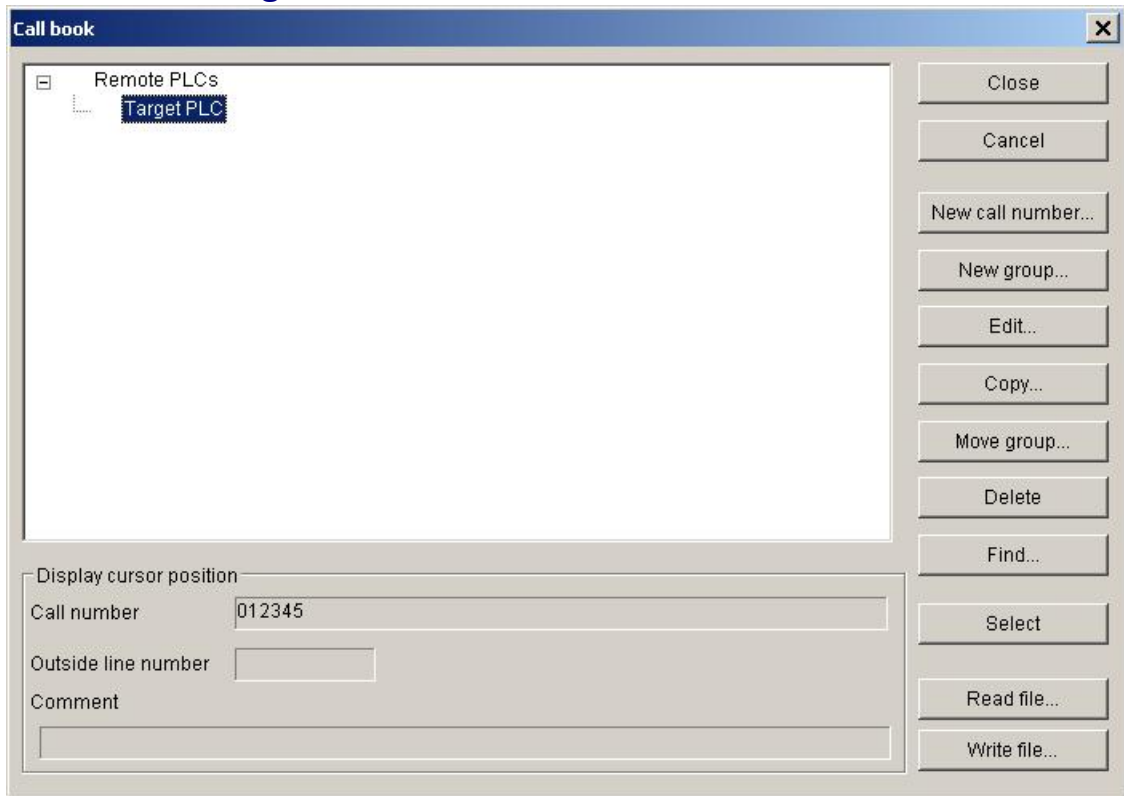


 **Note:**

The 'Next' button is disabled, if no number has been entered in the 'Call number' field.

Item		Description
Connect way		fixed to 'Auto line connect'
Callback number		not available
Line	Line type	set the line type (Default: Tone)
	Outside line number	set the outside line access number. The number must not exceed a length of 10 characters and can consist of 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, -, #
Connection target	Call number	enter the telephone number of the connection target.  When the connection target has been selected on the phone book screen, the telephone number of the connection target appears.  The phone number must not exceed a length of 50 characters and can consist of 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, -, #
	Name	an optional name, which has been assigned to the phone number in the phone book
	Browse	displays the phone book screen. For details see <a href="#">'Call book dialog'</a> <sup>[252]</sup>
AT command	Modem standard	when selected the standard AT commands are used
	AT command setting	AT command sequence to initialize the local modem  The command must not exceed a length of 70 ASCII characters
	Title	the optional title of the AT command selected in the AT command registration screen
	Browse	displays the AT command registration screen. For details see <a href="#">'AT command registration dialog'</a> <sup>[253]</sup>
	Help of AT command	displays the AT command help
Details setting		displays the <a href="#">'Details setting dialog'</a> <sup>[257]</sup>

### Call book dialog



Item	Description
Close	store the edited data and close the phone book dialog
Cancel	discard the edited data and close the phone book dialog
New call number	open the ' <a href="#">Call number setting dialog</a> ' <sup>[253]</sup> to enter a new phone number
New group	create a new group; opens the ' <a href="#">Group setting</a> ' <sup>[253]</sup> dialog for entering the group name
Edit	if a group is selected, the ' <a href="#">Group setting</a> ' <sup>[253]</sup> dialog is opened. Otherwise it opens the ' <a href="#">Call number setting dialog</a> ' <sup>[253]</sup> to edit a phone number
Copy	copies the selected phone number to a different group (see ' <a href="#">Select phone group name</a> ' <sup>[254]</sup> )
Move group	moves the selected phone number to a different group (see ' <a href="#">Select phone group name</a> ' <sup>[254]</sup> )
Delete	deletes the selected phone number or group. Only empty groups can be deleted. If a group is not empty, the phone numbers must be deleted first.
Find	used to search phone numbers and associated names (see ' <a href="#">Find a phone number</a> ' <sup>[254]</sup> )
Select	closes the dialog and copies the selected phone number to the line dialog
Read file	reads phone numbers from a file selected by the user
Write file	stores the contents of the phone book in a file selected by the user



### Call number setting dialog

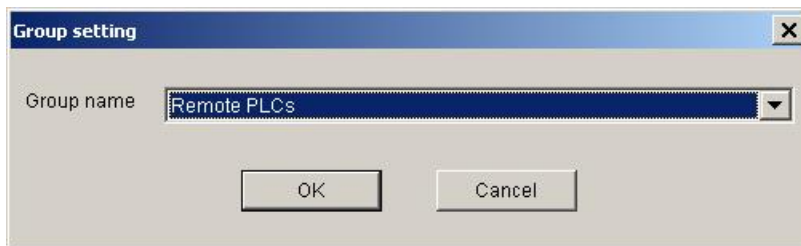
Set the telephone number to be registered to the phone book.

Item	Description
Group name	the group name of the registration destination.
Destination name	name associated with the phone number
Call number	enter the telephone number of the connection target. The phone number must not exceed a length of 50 characters and can consist of 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, -, , #
Outside line number	set the outside line access number. The number must not exceed a length of 10 characters and can consist of 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, -, , #
For only line connection	if a password has been set in A6TEL, Q6TEL or Q series C24 for remote access, the password must be entered here in order to access the target PLC
Comment	an optional comment describing the phone number

### Edit phone group name

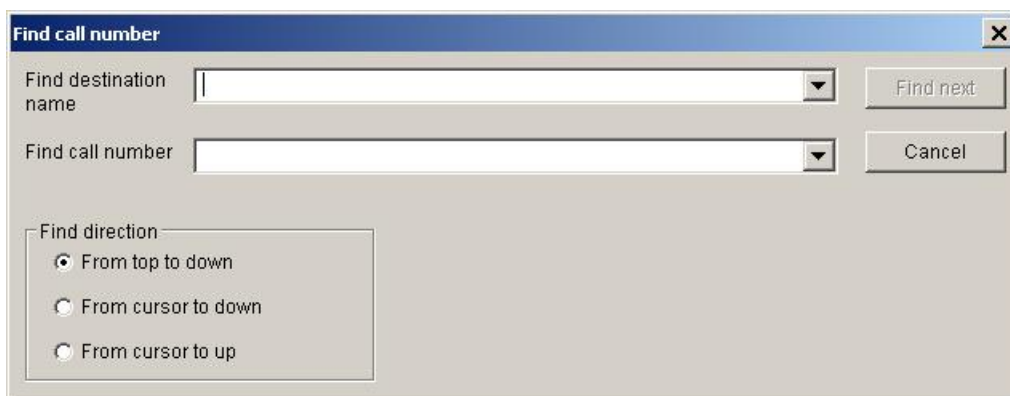
Enter the name for a group of telephone numbers.

## Select phone group name



Select the name of the group, where the selected phone number is copied to.

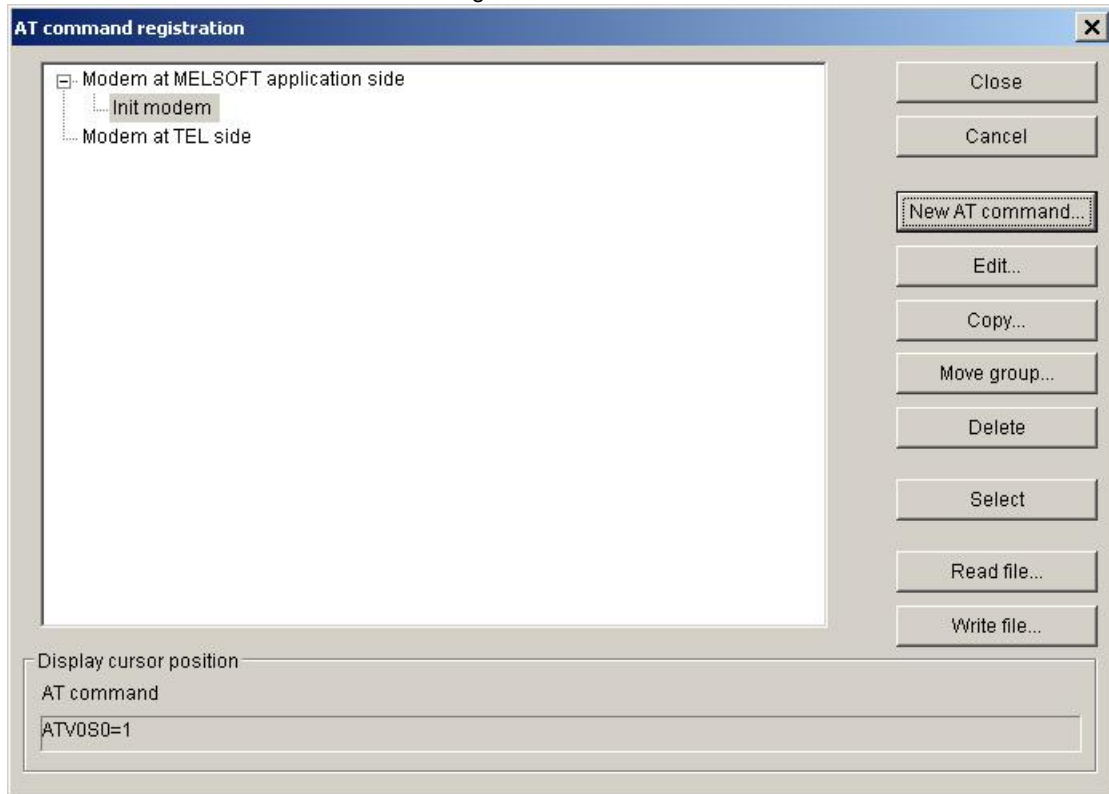
## Find a phone number



Search the phone book for a given name or number.

### AT command registration dialog

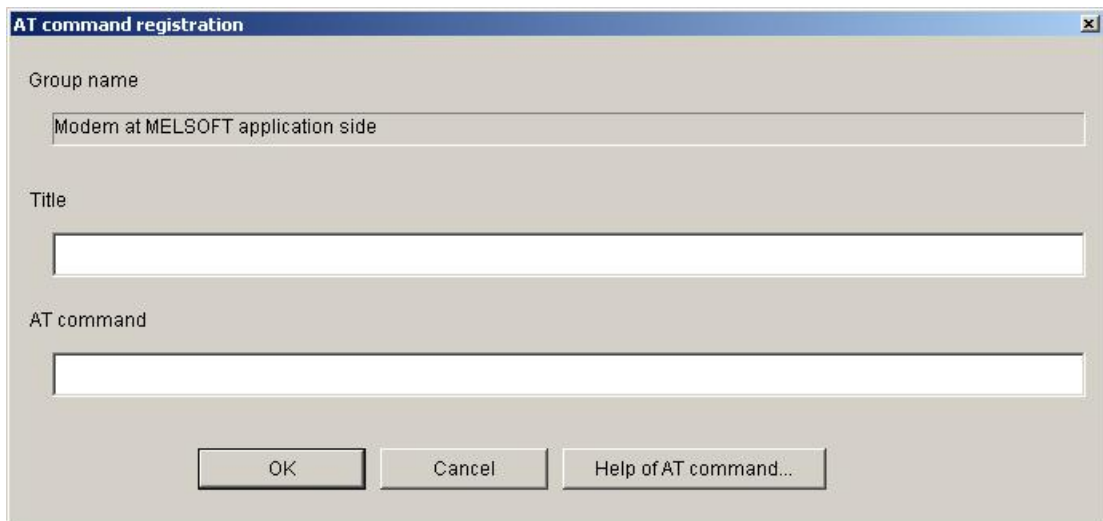
Set the AT commands used on the line setting screen.



Item	Description
Close	store the edited data and close the AT command registration dialog
Cancel	discard the edited data and close the AT command registration dialog
New AT command	define a new AT command. For details see ' <a href="#">AT command edit dialog</a> ' <sup>[256]</sup>
Edit	edit an existing AT command. For details see ' <a href="#">AT command edit dialog</a> ' <sup>[256]</sup>
Copy	copy the selected command to a different group (see ' <a href="#">Select AT command group name</a> ' <sup>[256]</sup> )
Move group	move the selected command to a different group (see ' <a href="#">Select AT command group name</a> ' <sup>[256]</sup> )
Delete	delete the selected AT command
Select	Used to display the AT command selected in the AT command display list on the line setting screen.
Read file	reads AT commands from a file selected by the user
Write file	stores the AT commands in a file selected by the user

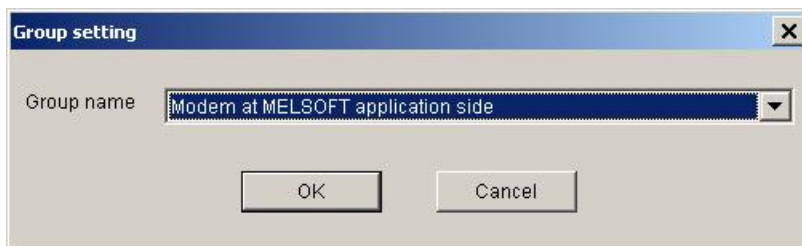
### AT command edit dialog

Register a new AT command and edit the AT command.



Item	Description
Group name	displays the name of the group to which the AT command is assigned
Title	a name for the AT command (max 60 ASCII characters)
AT command	enter the AT command for modem initialization (max. 70 ASCII characters)
Help of AT command	displays the AT command help

### Select AT command group name



Select the name of the group, where the selected AT command is copied to.

### Details setting dialog

Set details for telephone line connection.

Item	Value	Unit
Line connection CD signal wait time	90	seconds.
Line connection modem report wait time	5	seconds.
Line disconnection CD signal wait time	5	seconds.
Line disconnection delay time	3	seconds.
Data transmission delay time	0	seconds.
AT command response wait time	1	seconds.
Password cancellation response wait time	5	seconds.
AT command/password cancellation retry times	3	times.
Line callback cancel wait time	90	seconds.
Call back delay time	20	seconds.
Call back reception waiting time-out	120	seconds.

Item	Description
Line connection CD signal wait time	Set the maximum time to wait for the carrier-detect signal to be set (Default: 90) Increase the time, if the CD signal does not turn ON within the set time depending on the line-connected region (example: overseas). Setting range: 1 to 999
Line connection modem report waiting time	Set the line connection modem report wait time. (Default: 5) Increase the set time if the response speed of the modem is low. Setting range: 1 to 999
Line disconnection CD signal wait time	Set the maximum time to wait for the carrier-detect signal to be dropped (Default: 5) Increase the set time if the CD signal does not turn OFF within the preset time depending on the line-connected region (example: overseas). Setting range: 1 to 999
Line disconnection delay time	Set the line disconnection delay time. (Default: 3) Increase the set time if the response speed of the modem is low. Setting range: 1 to 999
Data transmission delay time	Set the data transmission delay time. (Default: 0) Increase the set time if the response speed of the modem is low. Setting range: 1 to 999

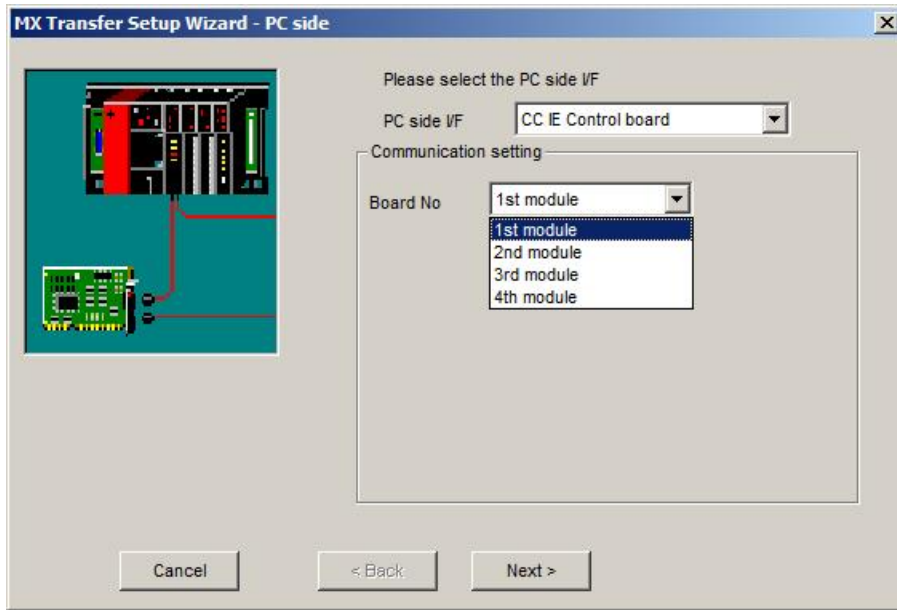
Item	Description
AT command response wait time	Set the AT command response wait time. (Default: 1) Increase the set time if the response speed of the modem is low. Setting range: 1 to 999
Password cancellation response wait time	Set the password cancellation response wait time. (Default: 5) Increase the set time if the quality of the line with the other end is low. Setting range: 1 to 999
AT command/password cancellation retry times	Set the AT command/password cancellation retry count. (Default: 3) Increase the set count if the AT command cannot be sent or the password cannot be canceled. Setting range: 1 to 999
Line callback cancel wait time	Set the Line callback cancel wait time. (Default: 90) Increase the set time if the line at the other end (Q series corresponding C24 side) is not disconnected within the set time depending on the line-connected region (example: overseas). Setting range: 1 to 180
Call back delay time	Set the callback delay time. (Default: 20) Increase the set time if the device for relaying connection to the line (example: modem, etc.) requires the set time for reconnection after line disconnection. Setting range: 1 to 999
Call back reception waiting time-out	Set the callback reception waiting time-out. (Default: 120) Increase the set time if a time-out occurs in a callback receive waiting status. Setting range: 1 to 3600

### 11.2.8 PC CC-Link IE Control Board

A connection between PC and PLC via CC-Link IE requires the CC-Link IE interface board and the corresponding driver to be installed in the PC.

#### *PC side CC-Link IE Control parameters*

Parameter	Values	Description
Board No.	1st, 2nd, 3rd, 4th module	selects the interface board in the PC



For a description of the editable parameters on the page see [PC side CC-Link IE parameters](#)<sup>[258]</sup>.  
 The PLC station, to which a connection is established, can be either

- the [host station](#)<sup>[330]</sup>
- a [different station in an attached network](#)<sup>[334]</sup>
- a [different station in a separate network](#)<sup>[336]</sup>

### Q CPU (Q mode)

Own Board	Connected Station CPU	Relayed Network	Relayed Station CPU						
	QCPU (Q mode)		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion	LCPU
o	o	MELSECNET/H	o	x	x	x	x	o	o
		MELSECNET/10	o	x	x	x	x	o	o
		MELSECNET(II)	x	x	x	x	x	x	x
		Ethernet	o	x	x	x	x	o	x
		C24	o	o	o	x	x	o	x
		CC-Link	o	x	x	x	x	o	o
		CC-Link IE Field	*2	x	x	x	x	x	o
		CC Link IE Control	o	x	x	x	x	o	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

\*2 : partially supported, no redundant CPUs

### Q CPU (A mode)

Own Board	Connected Station CPU	Relayed Network	Relayed Station CPU							
	QCPU (A mode)		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion	LCPU	
o	o	MELSECNET/H	x	x	x	x	x	x	x	x
		MELSECNET/10	x	x	x	x	x	x	x	x
		MELSECNET(II)	x	x	x	x	x	x	x	x
		Ethernet	x	x	x	x	x	x	x	x
		C24	x	x	x	x	x	x	x	x
		CC-Link	x	x	x	x	x	x	x	x
		CC-Link IE Field	x	x	x	x	x	x	x	x
		CC-Link IE Control	x	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

### QnA CPU

Own Board	Connected Station CPU	Relayed Network	Relayed Station CPU							
	QnACPU		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion	LCPU	
o	o	MELSECNET/H	x	x	x	x	x	x	x	x
		MELSECNET/10	x	x	o	x	x	x	x	x
		MELSECNET(II)	x	x	x	x	x	x	x	x
		Ethernet	x	x	o	x	x	x	x	x
		C24	x	x	x	x	x	x	x	x
		CC-Link	x	x	x	x	x	x	x	x
		CC-Link IE	x	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU



### A CPU

Own Board	Connected Station CPU	Relayed Network	Relayed Station CPU						
	ACPU *1		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion	LCPU
o	o	MELSECNET/H	x	x	x	x	x	x	x
		MELSECNET/10	x	x	x	x	x	x	x
		MELSECNET(II)	x	x	x	x	x	x	x
		Ethernet	x	x	x	x	x	x	x
		C24	x	x	x	x	x	x	x
		CC-Link	x	x	x	x	x	x	x
		CC-Link IE	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

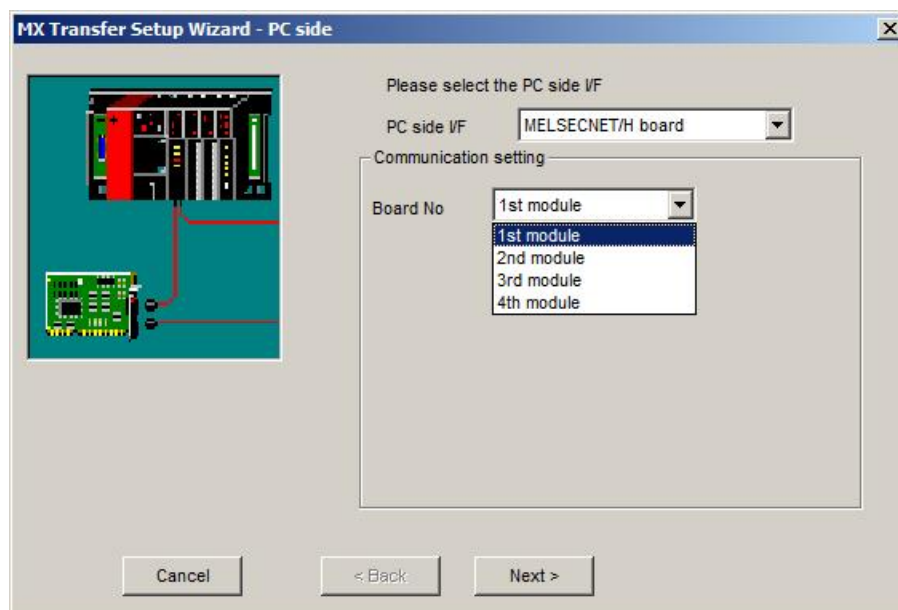
\*1 : Including motion controller CPU

### 11.2.9 PC MELSECNET/H Board

A connection between PC and PLC via MELSECNET/H requires the MELSECNET/H interface board and the corresponding driver to be installed in the PC.

#### PC side MELSECNET/H parameters

Parameter	Values	Description
Board No.	1st, 2nd, 3rd, 4th module	selects the interface board in the PC



For a description of the editable parameters on the page see [PC side MELSECNET/H parameters](#)<sup>261</sup>.

The PLC station, to which a connection is established, can be either

- the [host station](#)<sup>[330]</sup>
- a [different station in an attached network](#)<sup>[334]</sup>
- a [different station in a separate network](#)<sup>[334]</sup>

### MELSECNET/H mode to Q CPU (Q mode)

Own Board	Connected Station CPU	Relayed Network	Relayed Station CPU							
	QCPU (Q mode)		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion	LCPU	
o	o	MELSECNET/H	o	x	x	x	x	x	o	x
		MELSECNET/10	o	o	o	o	o	x	o	x
		MELSECNET(II)	x	x	x	x	x	x	x	x
		Ethernet	o	x	x	o	x	x	o	x
		C24	o	x	o	x	x	x	o	o
		CC-Link	o	x	x	x	x	o	o	o
		CC-Link IE Field	*3	x	x	x	x	x	o	x
		CC-Link IE Control	o	x	x	x	x	x	o	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

\*2 : Operates only when QCPU (Q mode) is used.

\*3 : For the Redundant PLC, access is disabled.

\*4 : partially supported, depends on CPU type

### MELSECNET/10 mode to Q CPU (Q mode)

Own Board	Connected Station CPU		Relayed Network	Relayed Station CPU						
	QCPU (Q mode)			QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion	LCPU
o	o		MELSECNET/H	x	x	x	x	x	o	x
			MELSECNET/10	o	o	o	o	x	o	x
			MELSECNET(II)	x	x	x	x	x	x	x
			Ethernet	o	x	x	x	x	o	x
			C24	o	x	o	x	x	o	o
			CC-Link	o	x	x	x	x	o	o
			CC-Link IE Field	*2	x	x	x	x	o	x
			cc-Link Control IE	o	x	x	x	x	o	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

\*2 : For the Redundant PLC, access is disabled

\*3 : partially supported, depends on CPU type

### MELSECNET/10 mode to Q CPU (A mode) and A CPU

Own Board	Connected Station CPU		Relayed Network	Relayed Station CPU						
	QCPU (A mode)	ACPU *1		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion	LCPU
o	o	o	MELSECNET/H	x	x	x	x	x	x	x
			MELSECNET/10	o	o	o	o	x	x	x
			MELSECNET(II)	x	x	x	x	x	x	x
			Ethernet	x	x	x	x	x	x	x
			C24	x	x	x	x	x	x	x
			CC-Link	x	x	x	x	x	x	x
			CC-Link IE	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

### MELSECNET/10 mode to QnA CPU

Own Board	Connected Station CPU	Relayed Network	Relayed Station CPU						
	QnACPU		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion	LCPU
o	o	MELSECNET/H	x	x	x	x	x	x	x
		MELSECNET/10	o	o	o	o	x	x	x
		MELSECNET(II)	x	x	x	x	x	x	x
		Ethernet	x	x	o	x	x	x	x
		C24	x	x	o	x	x	x	x
		CC-Link	x	x	x	x	x	x	x
		CC-Link IE	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

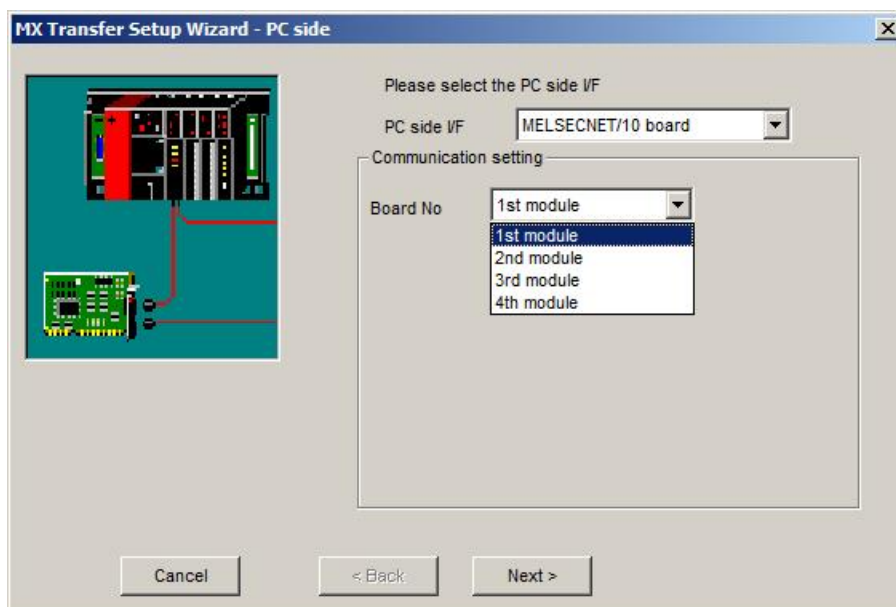
\*1 : Including motion controller CPU

#### 11.2.10 PC MELSECNET/10 Board

A connection between PC and PLC via MELSECNET/10 requires the MELSECNET/10 interface board and the corresponding driver to be installed in the PC.

#### PC side MELSECNET/10 parameters

Parameter	Values	Description
Board No.	1st, 2nd, 3rd, 4th module	selects the interface board in the PC



For a description of the editable parameters on the page see [PC side MELSECNET/10 parameters](#)<sup>[264]</sup>.

The PLC station, to which a connection is established, can be either

- the [host station](#)<sup>[330]</sup>
- a [different station in an attached network](#)<sup>[334]</sup>
- a [different station in a separate network](#)<sup>[334]</sup>

### Q CPU (Q mode)

Own Board	Connected Station CPU	Relayed Network	Relayed Station CPU						
	QCPU (Q mode)		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion	LCPU
0	0	MELSECNET/H	o	x	x	x	x	o	x
		MELSECNET/10	o	o	o	o	x	o	x
		MELSECNET(II)	x	x	x	x	x	x	x
		Ethernet	o	x	x	x	x	x	x
		C24	o	x	o	x	x	o	o
		CC-Link	o	x	x	x	x	o	o
		CC-Link IE Field	*3	x	x	x	x	x	o
		CC-Link IE Control	o	x	x	x	x	o	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

\*2 : For the Redundant PLC, access is disabled

\*3 : partially supported, depends on CPU type

### Q CPU (A mode) and A CPU

Own Board	Connected Station CPU		Relayed Network	Relayed Station CPU							
	QCPU (A mode)	ACPU *1		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion	LCPU	
o	o	o	MELSECNET/H	x	x	x	x	x	x	x	x
			MELSECNET/10	o	o	o	o	x	x	x	
			MELSECNET(II)	x	x	x	x	x	x	x	x
			Ethernet	x	x	x	x	x	x	x	x
			Computer link	x	x	x	x	x	x	x	x
			CC-Link	x	o	x	x	x	x	x	x
			CC-Link IE	x	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

### QnA CPU

Own Board	Connected Station CPU		Relayed Network	Relayed Station CPU							
	QnACPU			QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion	LCPU	
o	o		MELSECNET/H	x	x	x	x	x	x	x	x
			MELSECNET/10	o	o	o	o	x	x	x	
			MELSECNET(II)	x	x	x	x	x	x	x	x
			Ethernet	x	x	o	x	x	x	x	x
			C24	x	x	o	x	x	x	x	x
			CC-Link	x	x	x	x	x	x	x	x
			CC-Link IE	x	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

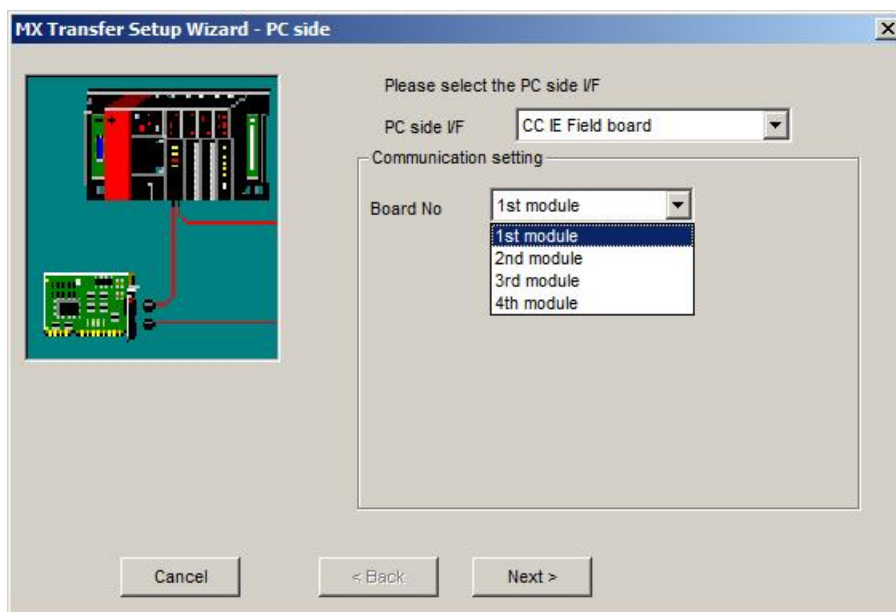
\*1 : Including motion controller CPU

### 11.2.11 PC CC-Link IE Field Board

A connection between PC and PLC via CC-Link IE Field requires the CC-Link IE Field interface board and the corresponding driver to be installed in the PC.

#### *PC side CC-Link IE Field parameters*

Parameter	Values	Description
Board No.	1st, 2nd, 3rd, 4th module	selects the interface board in the PC



For a description of the editable parameters on the page see [PC side CC-Link IE parameters](#)<sup>[267]</sup>.

The PLC station, to which a connection is established, can be either

- the [host station](#)<sup>[330]</sup>
- a [different station in an attached network](#)<sup>[334]</sup>
- a [different station in a separate network](#)<sup>[336]</sup>

### Q CPU (Q mode)

Own Board	Connected Station CPU	Relayed Network	Relayed Station CPU							
	QCPU (Q mode)		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion	LCPU	
o	o	MELSECNET/H	o	x	x	x	x	x	x	x
		MELSECNET/10	o	o	o	o	o	x	x	x
		MELSECNET(II)	x	x	x	x	x	x	x	x
		Ethernet	o	x	x	x	x	x	x	x
		C24	o	x	x	x	x	x	x	o
		CC-Link	o	x	x	x	x	x	x	o
		CC-Link IE Field	*2	x	x	x	x	x	x	o
		CC-Link IE Control	o	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

\*2 : For the Redundant PLC, access is disabled

### L CPU

Own Board	Connected Station CPU	Relayed Network	Relayed Station CPU							
	LCPU		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion	LCPU	
o	o	MELSECNET/H	x	x	x	x	x	x	x	x
		MELSECNET/10	x	x	x	x	x	x	x	x
		MELSECNET(II)	x	x	x	x	x	x	x	x
		Ethernet	x	x	x	x	x	x	x	x
		C24	o	x	x	x	x	x	x	o
		CC-Link	o	x	x	x	x	x	x	o
		CC-Link IE Field	*2	x	x	x	x	x	x	x
		CC-Link IE Control	x	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

\*2 : For the Redundant PLC, access is disabled




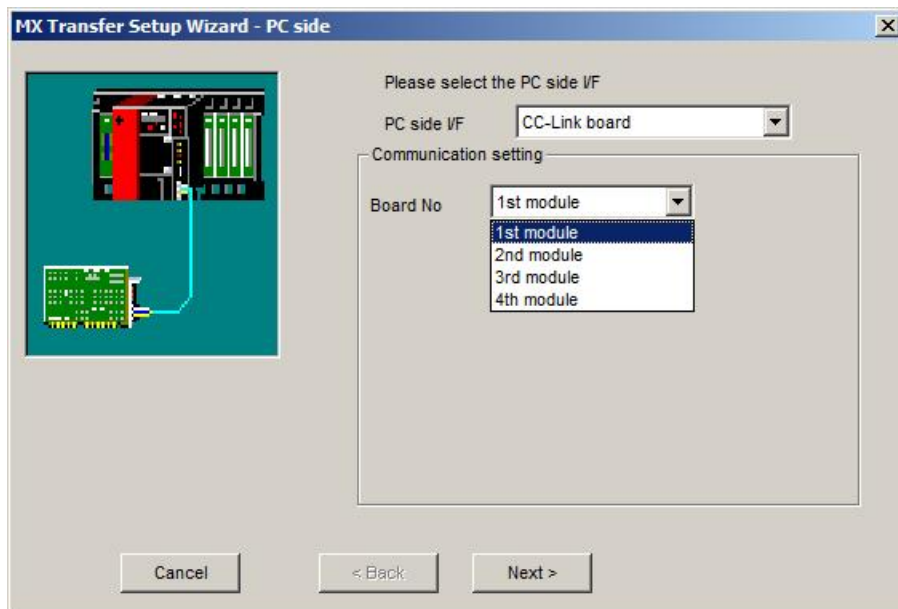
### 11.2.12 PC CC-Link Board

A connection between PC and PLC via CC-Link requires the CC-Link interface board and the corresponding driver to be installed in the PC.

#### *PC side CC-Link parameters*

Parameter	Values	Description
Board No.	1st, 2nd, 3rd, 4th module	selects the interface board in the PC

	<p><b>Note:</b></p> <p>The CC-Link master/local module used in CC-Link communication or CC-Link G4 communication (only when the AJ65BT-G4 is used), must have software version 'N' or later. Modules with software version 'M' or earlier will not operate properly.</p>
---	--



For a description of the editable parameters on the page see [PC side CC-Link parameters](#)<sup>[269]</sup>.

The PLC station, to which a connection is established, can be either

- the [host station](#)<sup>[330]</sup>
- a [different station in an attached network](#)<sup>[334]</sup>
- a [different station in a separate network](#)<sup>[335]</sup>

### Q CPU (Q mode)

Own Board	Connected Station CPU	Relayed Network	Relayed Station CPU						
	QCPU (Q mode)		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion	LCPU
o	o	MELSECNET/H	o	x	x	x	x	o	x
		MELSECNET/10	o	x	x	x	x	o	x
		MELSECNET(II)	x	x	x	x	x	x	x
		Ethernet	o	x	x	x	x	o	x
		C24	x	x	x	x	x	x	x
		CC-Link	x	x	x	x	x	x	x
		CC-Link IE Control	o	x	x	x	x	o	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

\*2 : partially supported, depends on CPU type

### Q CPU (A mode)

Own Board	Connected Station CPU	Relayed Network	Relayed Station CPU						
	QCPU (A mode)		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion	LCPU
o	o	MELSECNET/H	x	x	x	x	x	x	x
		MELSECNET/10	x	x	x	x	x	x	x
		MELSECNET(II)	x	x	x	x	x	x	x
		Ethernet	x	x	x	x	x	x	x
		C24	x	x	x	x	x	x	x
		CC-Link	x	x	x	x	x	x	x
		CC-Link IE	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

### QnA CPU

Own Board	Connected Station CPU	Relayed Network	Relayed Station CPU						
	QnACPU		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion	LCPU
o	o	MELSECNET/H	x	x	x	x	x	x	x
		MELSECNET/10	x	x	o	x	x	x	x
		MELSECNET(II)	x	x	x	x	x	x	x
		Ethernet	x	x	o	x	x	x	x
		C24	x	x	x	x	x	x	x
		CC-Link	x	x	x	x	x	x	x
		CC-Link IE	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

### A CPU

Own Board	Connected Station CPU	Relayed Network	Relayed Station CPU						
	ACPU *1		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion	LCPU
o	o	MELSECNET/H	x	x	x	x	x	x	x
		MELSECNET/10	x	x	x	x	x	x	x
		MELSECNET(II)	x	x	x	x	x	x	x
		Ethernet	x	x	x	x	x	x	x
		C24	x	x	x	x	x	x	x
		CC-Link	x	x	x	x	x	x	x
		CC-Link IE	x	x	x	x	x	x	x

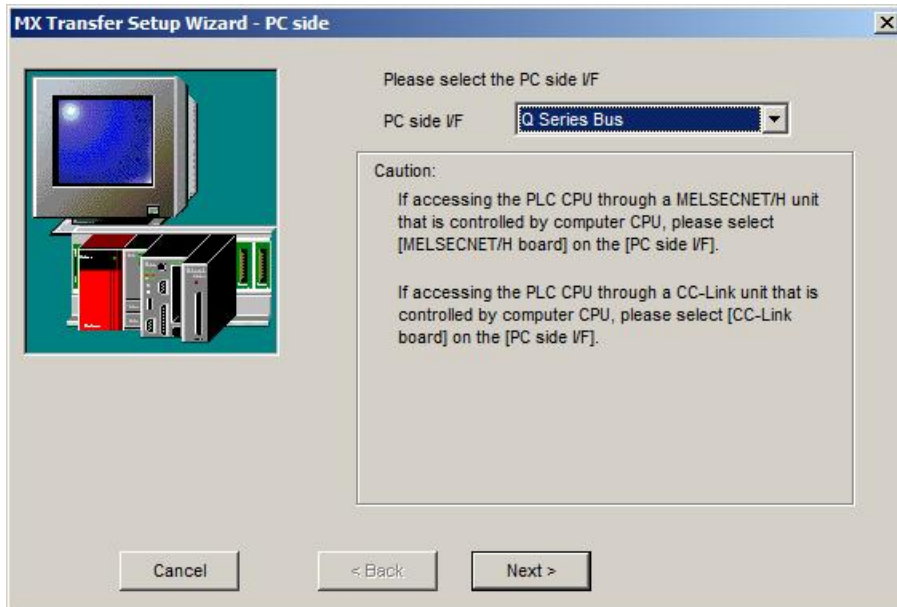
o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

### 11.2.13 PC Q Series Bus

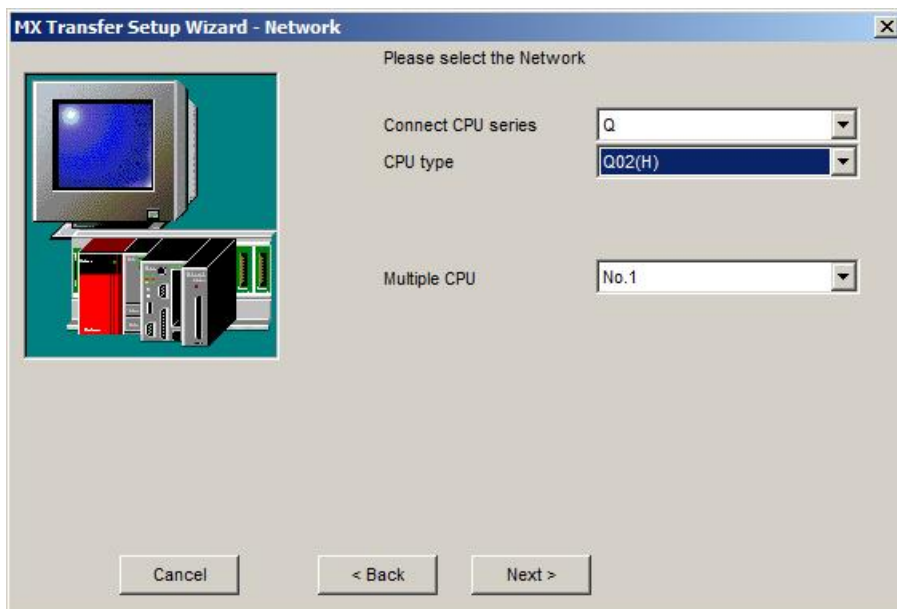
Configure a connection from the PC CPU module installed in a Q series rack to a PLC CPU within the same rack, using the back panel bus. Alternatively the communication between PC CPU and PLC CPU module can use a [MELSECNET/H](#)<sup>[261]</sup> or a [CC-Link](#)<sup>[269]</sup> interface.



Pressing the <Next> button opens a [page<sup>272</sup>](#) for selecting the connected CPU type.

### Q Series Bus - Network

Parameter	Values	Description
CPU type	see list of supported CPU types	type of PLC CPU module
Multiple CPU	No. 1, No. 2, No. 3	selects the CPU module in the PLC




Pressing the <Next> button opens a [page<sup>350</sup>](#) for entering a comment for the connection.

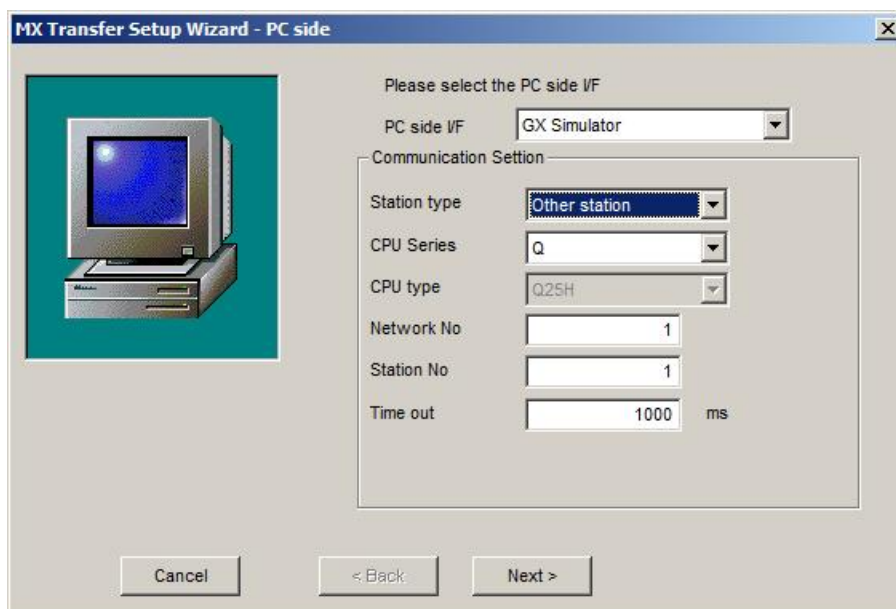
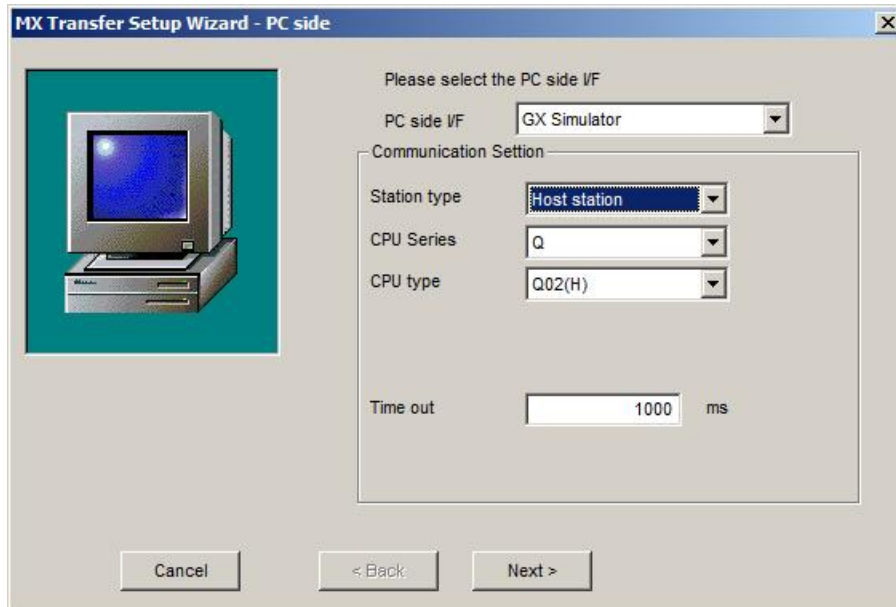
### 11.2.14 PC GX Simulator

Define a connection to an instance of GX Simulator running on the same computer.

#### *PC side GX Simulator parameters*

Parameter	Values	Description
Station type	'Host station' or 'Other station'	direct connection or MELSECNET network
CPU series	Q, QnA, A, FX	series of the CPU type selected in GX Simulator  (FX is only available for station type 'Host station')
CPU type	see list of supported CPU types	type of CPU selected in GX Simulator  <b>not selectable for 'Other station' as 'Station type'</b>
Network No	1 - 239	network number of the E71 ethernet module, the PC is connected to. The E71 network number is configured in the 'Network Parameters' of the PLC settings.  <b>only for 'Other station' as 'Station type'</b>
Station No.	1 - 64	station number of the PC side interface. This number can be freely chosen, but must be unique within the network specified by the network number.  <b>only for 'Other station' as 'Station type'</b>
Time out	1 - 2147483647	communication timeout in milliseconds

	<p><b>Note:</b></p> <p>Before configuring the connection please make sure that GX Simulator and GX Developer are operating.</p> <p>In addition, do not terminate GX Simulator and GX Developer while the user program is running. If you do so, you will not be able to terminate the user program normally.</p>
---	--



For a description of the editable parameters on the page see [PC side GX Simulator parameters](#)<sup>[273]</sup>.

Pressing the <Next> button opens a [page](#)<sup>[350]</sup> for entering a comment for the connection.


### 11.2.15 PC GX Simulator2

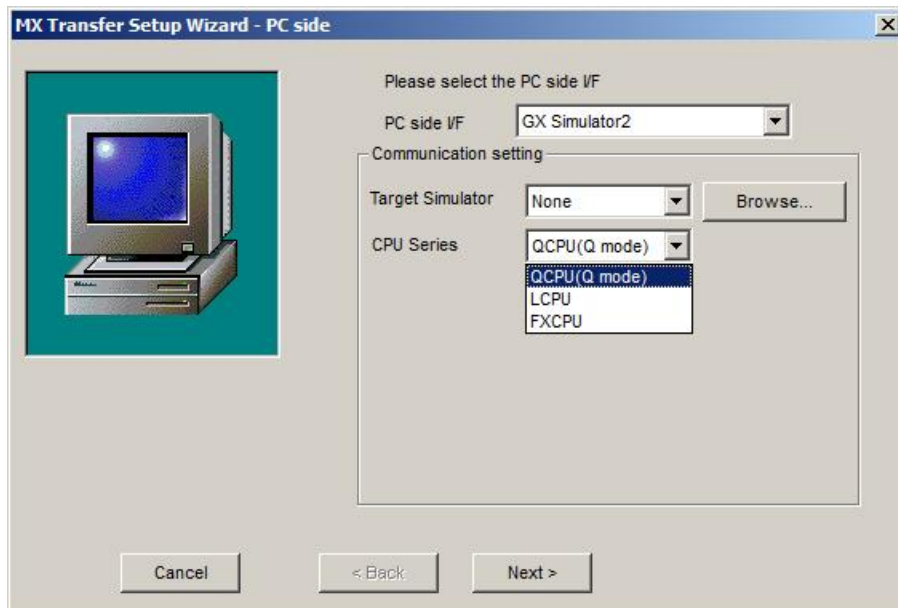
Define a connection to an instance of GX Simulator2 running on the same computer.

**PC side GX Simulator2 parameters**

Parameter	Values	Description
Target simulator	<ul style="list-style-type: none"> <li>• None</li> <li>• Simulator A</li> <li>• Simulator B</li> <li>• Simulator C</li> <li>• Simulator D</li> </ul>	simulator instance to connect to (for FX series projects at present only one simulation can be executed at a time)  if the simulation is currently running, pressing the 'Browse' button opens the ' <a href="#">Find GX Simulator2 Runtime dialog</a> <sup>[275]</sup> to view additional information of the available targets, e.g. the GXW2 project name
CPU series	Q, L, FX	series of the CPU type selected in GX Simulator2

**Note:**

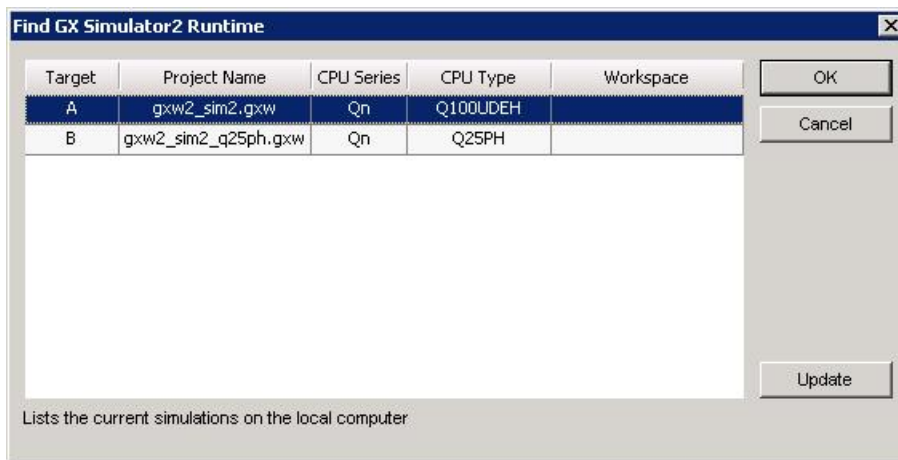
 Before configuring the connection please make sure that GX Simulator2 is operating. In addition, do not terminate GX Simulator2 while the user program is running. If you do so, you will not be able to terminate the user program normally.



For a description of the editable parameters on the page see [PC side GX Simulator2 parameters](#)<sup>[275]</sup>. Pressing the **<Next>** button opens a [page](#)<sup>[350]</sup> for entering a comment for the connection.

### 'Find GX Simulator2 Runtime' Dialog

This dialog shows the list of locally active GX Simulator2 runtimes.



For each runtime the following items are displayed:

Column	Description
Target	marks the instance of the runtime, if multiple runtimes are active
Project Name	file name of the GXW2 project
CPU Series	family of the CPU type set in the GXW2 project
CPU Type	CPU type set in the GXW2 project
Workspace	the GXW2 workspace of the project, if it is part of a workspace

The table contents are not automatically refreshed, when a runtime is started or stopped. To update the list of runtimes press the <Update> button.

Pressing <OK> sets the 'Target Simulator' selection in the transfer setup wizard to the simulation, which has been selected in the table.

#### 11.2.16 PC MT Simulator2

Define a connection to an instance of MT Simulator2 running on the same computer. MT Simulator2 simulates only motion CPUs which are supported by MT Developer2. The simulator runtime can be controlled with MT Developer2.

#### PC side MT Simulator2 parameters

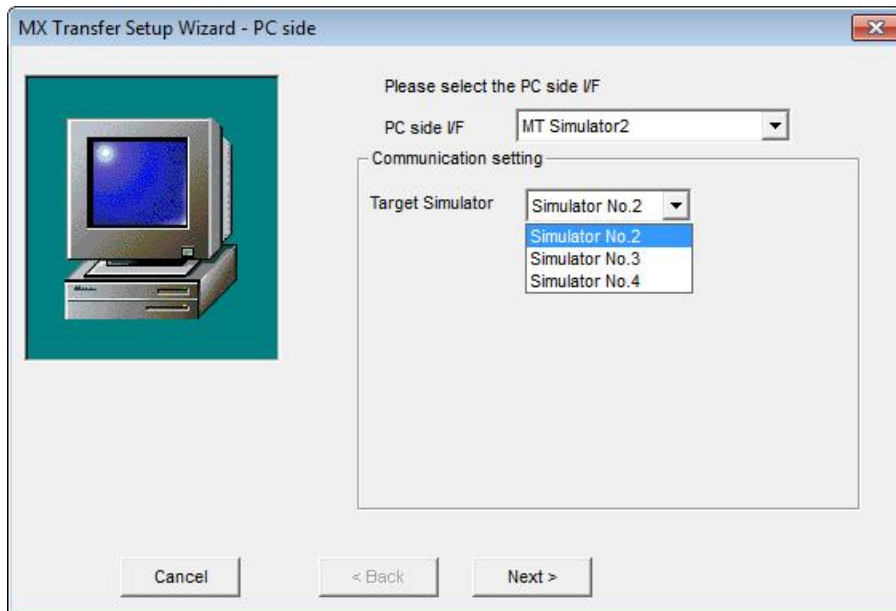
Parameter	Values	Description
Target simulator	<ul style="list-style-type: none"> <li>• Simulator No. 2</li> <li>• Simulator No. 3</li> <li>• Simulator No. 4</li> </ul>	simulator instance to connect to



**Note:**

Before configuring the connection please make sure that MT Simulator2 is operating.

In addition, do not terminate MT Simulator2 while the user program is running. If you do so, you will not be able to terminate the user program normally.

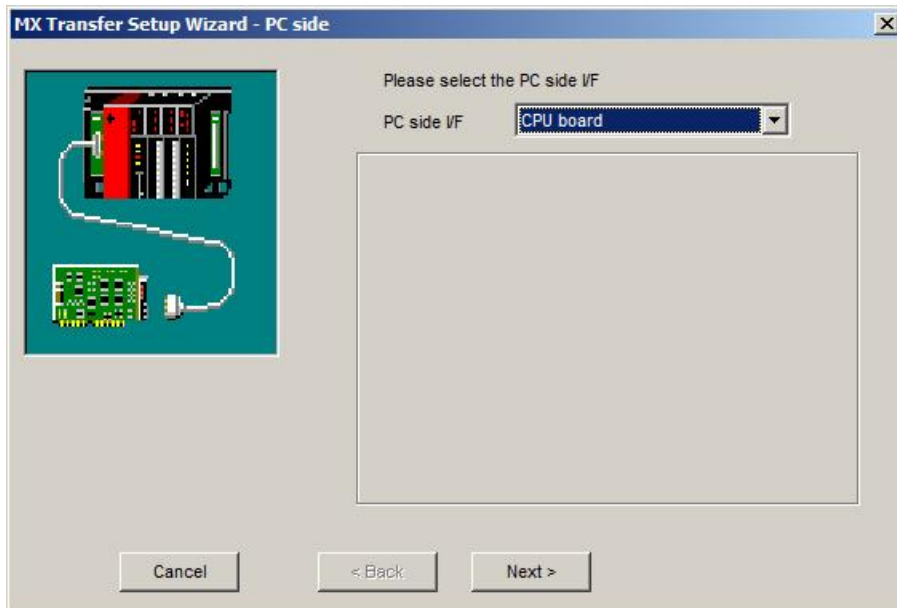


For a description of the editable parameters on the page see [PC side MT Simulator2 parameters](#)<sup>[276]</sup>.

Pressing the <Next> button opens a [page](#)<sup>[350]</sup> for entering a comment for the connection.

### 11.2.17 PC CPU Board

A connection between PC and PLC using a CPU board requires the CPU board and the corresponding driver to be installed in the PC.



The PLC station, to which a connection is established, can be either


- the [host station](#)<sup>[333]</sup>
- a [different station in an attached network](#)<sup>[336]</sup>


### 11.2.18 PLC CPU Interface

Enter the settings for the serial interface of a PLC CPU module.

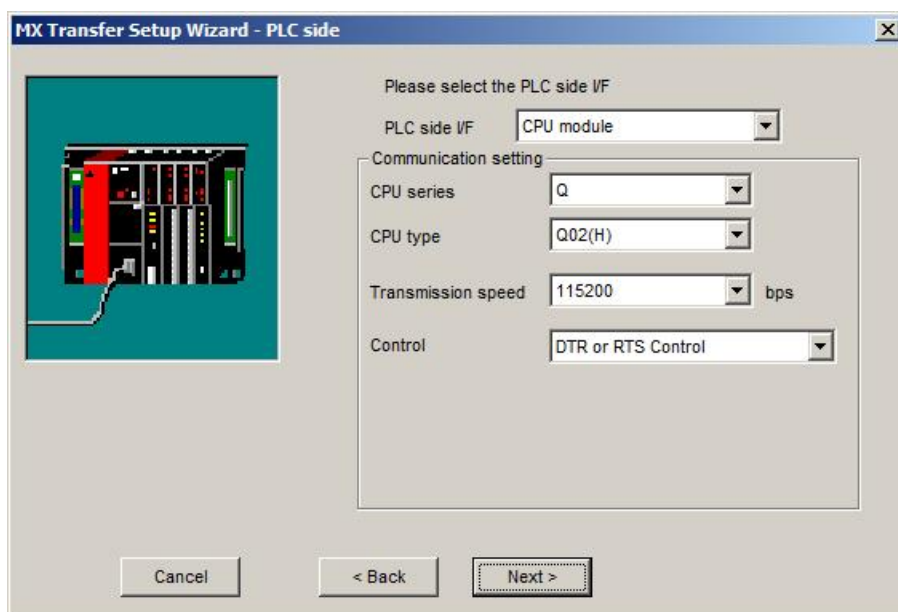
#### *PLC side CPU interface parameters*

Parameter	Values	Description
CPU series	Q, QnA, A, FX, L, Q Motion	series of the CPU type selected in serial PC side interface
CPU type	see list of supported CPU types	type of PLC CPU module
Transmission speed	300 - 115200	baudrate on the serial link
Control	DTR only, RTS only, DTR and RTS, DTR or RTS	handshake signals

	<p><b>Note for Q00J/Q00/Q01 CPUs:</b></p> <p>If the baudrate set in the PC differs from the baudrate in the CPU module, the actual baudrate is fixed at 9600 bps. To increase the communication speed, match the personal computer side transmission speed with the Q00J/Q00/Q01 CPU side transmission speed.</p>
---	---

	<p><b>Supported transmission speeds:</b></p> <p>As the transmission speed of the QCPU(Q mode) and QCPU(A mode), you can set 9600bps, 19200bps, 38400bps, 57600bps or 115200bps.</p> <p>For the QnACPU of version 9707B or later, you can set the transmission speed of 9600bps, 19200bps or 38400bps.</p> <p>For the QnACPU of other versions, you can set 9600bps or 19200bps.</p> <p>The transmission speeds of the ACPU (except A2USHCPU-S1), FXCPU and motion controller CPU are fixed to 9600bps. (The A2USHCPU-S1 may be set to 19200bps.)</p>
---	--

### PLC CPU Standard Serial Interface (RS 422)



For a description of the editable parameters on the page see [PLC side CPU interface parameters](#)<sup>[278]</sup>.  
 Pressing the <Next> button opens a [page](#)<sup>[350]</sup> for entering a comment for the connection.

### Q CPU (Q mode)

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
	MELSECNET/H	o	x	x	x	x	o	x
	MELSECNET/10	o	o	o	o	x	o	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	o	x	o	x	x	o	x
	C24	o	x	o	x	x	o	o
	CC-Link	o	*2	*2	*2	*4	o	o
	CC-Link IE Field	*3	x	x	x	x	x	o
	CC-Link IE Control	o	x	x	x	x	o	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

\*2 : Use the QnA or ACPUs side CC-Link module whose ROM version is "S" or later

\*3 : For the Redundant PLC, access is disabled

\*4 : partially supported, depends on CPU type

### Q CPU (A mode)

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
	MELSECNET/H	x	x	x	x	x	x	x
	MELSECNET/10	x	o	x	o	x	x	x
	MELSECNET(II)	x	o	x	o	x	x	x
	Ethernet	x	x	x	x	x	x	x
	C24	x	x	x	x	x	x	x
	CC-Link	x	x	x	x	x	x	x
	CC-Link IE	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

## QnA CPU

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
QnACPU	MELSECNET/H	x	x	x	x	x	x	x
	MELSECNET/10	x	x	o	x	x	x	x
	MELSECNET(II)	x	x	o	x	x	x	x
	Ethernet	x	x	o	x	x	x	x
	C24	x	x	o	x	x	x	x
	CC-Link	x	x	x	x	x	x	x
	CC-Link IE	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

## A CPU

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
ACPU *1	MELSECNET/H	x	x	x	x	x	x	x
	MELSECNET/10	x	o	x	o	x	x	x
	MELSECNET(II)	x	o	x	o	x	x	x
	Ethernet	x	x	x	x	x	x	x
	C24	x	x	x	x	x	x	x
	CC-Link	x	x	x	x	x	x	x
	CC-Link IE	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

### FX CPU

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
FXCPU	MELSECNET/H	x	x	x	x	x	x	x
	MELSECNET/10	x	x	x	x	x	x	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	x	x	x	x	x	x	x
	C24	x	x	x	x	x	x	x
	CC-Link	x	x	x	x	*3	x	x
	CC-Link IE	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

\*3 : partially supported, depends on CPU type

### LCPU

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
LCPU	MELSECNET/H	x	x	x	x	x	x	x
	MELSECNET/10	x	x	x	x	x	x	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	x	x	x	x	x	x	x
	C24	o	x	o	x	x	x	o
	CC-Link	o	o	o	o	x	x	o
	CC-Link IE Field	*2	x	x	x	x	x	o

o: Accessible

x: Inaccessible

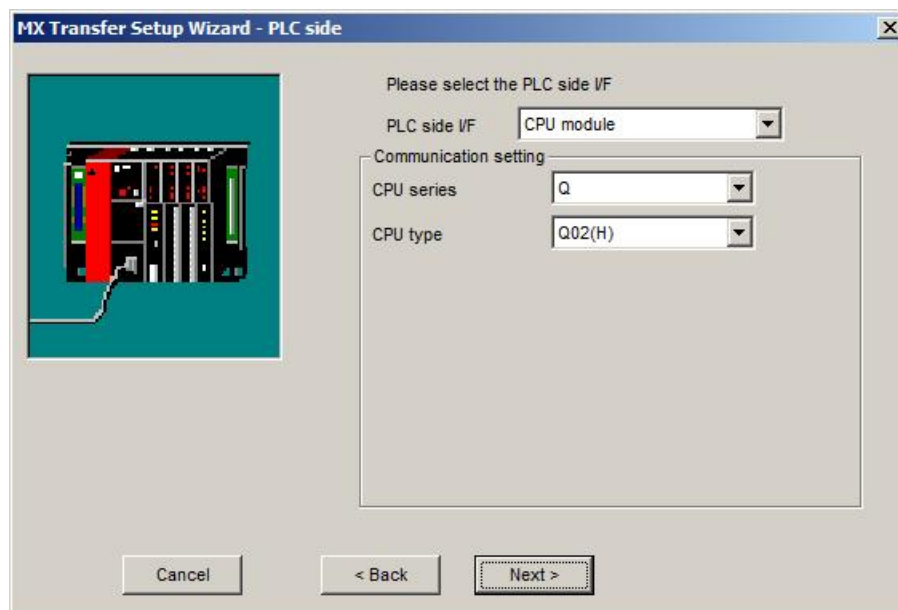
\*1 : Including motion controller CPU

\*2 : for the redundant CPU types , access is disabled

## PLC CPU Universal Serial Bus Interface (USB)

### PLC side CPU USB interface parameters

Parameter	Values	Description
CPU series	Q, FX, L, Q Motion	series of the CPU type
CPU type	see list of supported CPU types	type of PLC CPU module



For a description of the editable parameters on the page see [PLC side CPU USB interface parameters](#)<sup>[283]</sup>.

Pressing the <Next> button opens a [page](#)<sup>[350]</sup> for entering a comment for the connection.

### Q CPU (Q mode)

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
	MELSECNET/H	o	x	x	x	x	o	x
	MELSECNET/10	o	o	o	o	x	o	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	o	x	o	x	x	o	x
	C24	o	x	o	x	x	o	o
	CC-Link	o	*2	*2	*2	*4	o	o
	CC-Link IE Field	*3	x	x	x	x	x	o
	CC-Link IE Control	o	x	x	x	x	o	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

\*2 : Use the QnA or ACPUs side CC-Link module whose ROM version is "S" or later

\*3 : For the Redundant PLC, access is disabled

\*4 : partially supported, depends on CPU type

### FX CPU

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU *2	Q Motion CPU	LCPU
	MELSECNET/H	x	x	x	x	x	x	x
	MELSECNET/10	x	x	x	x	x	x	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	x	x	x	x	x	x	x
	C24	x	x	x	x	x	x	x
	CC-Link	x	x	x	x	*2	x	x
	CC-Link IE Field	x	x	x	x	x	x	x
	CC-Link IE Control	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

\*2 : partially supported, depends on CPU type



### L CPU

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
	MELSECNET/H	o	x	x	x	x	x	x
	MELSECNET/10	o	o	o	o	x	x	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	o	x	o	x	x	x	x
	C24	o	x	o	x	x	x	o
	CC-Link	o	*2	*2	*2	x	x	o
	CC-Link IE Field	*3	x	x	x	x	x	o
	CC-Link IE Control	x	x	x	x	x	x	x


o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

\*2 : Use the QnA or ACPUs side CC-Link module whose ROM version is "S" or later

\*3 : For the Redundant PLC, access is disabled

	<p><b>Warning:</b></p> <p>Frequently disconnecting/reconnecting the USB cable or resetting or powering ON/OFF the PLC CPU during communications with the PLC CPU may cause a communication error which cannot be recovered.</p> <p>If it is not recovered, completely disconnect the USB cable once and then reconnect it after 5 or more seconds have elapsed.</p>
---	---

### PLC Ethernet CPU

Please see [PLC Ethernet Interface](#)<sup>[317]</sup>.


### 11.2.19 PLC C24 Interface

Enter the settings for a serial interface module in the PLC.

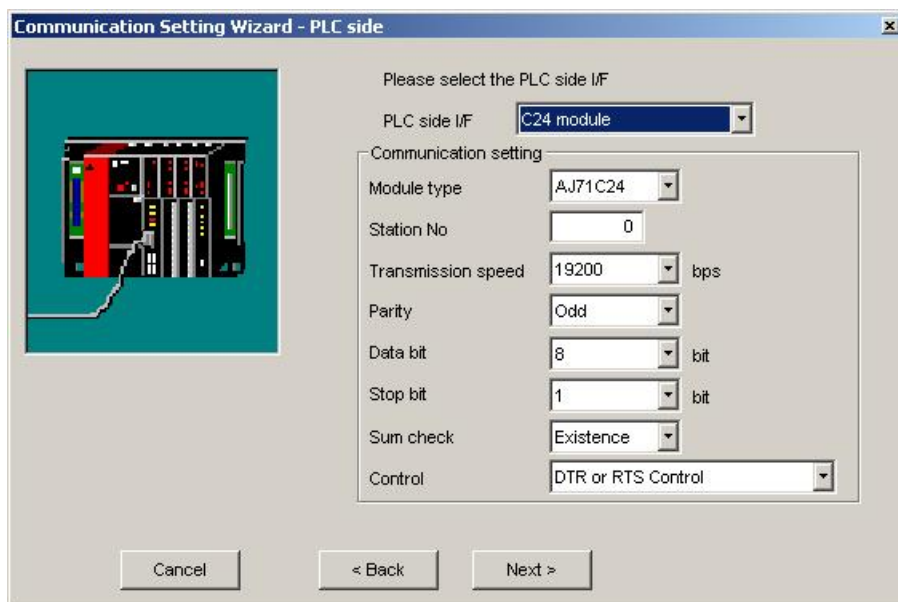
#### PLC side C24 module parameters

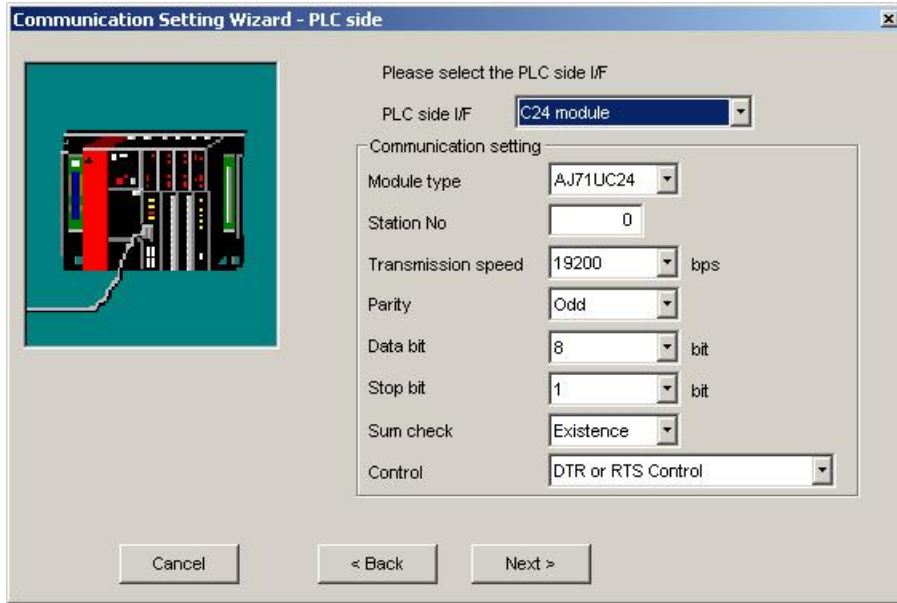
Parameter	Values	Description
Module type	<a href="#">AJ71C24</a> <sup>[286]</sup> , <a href="#">AJ71QC24</a> <sup>[288]</sup> , <a href="#">LJ71C24</a> <sup>[296]</sup> , <a href="#">AJ71UC24</a> <sup>[286]</sup> , <a href="#">QJ71C24</a> <sup>[290]</sup>	type of PLC C24 module

Parameter	Values	Description
Station No.	0 - 31	station number of the PC side interface. This number can be freely chosen, but must be unique within the network specified by the network number.
Transmission speed	300 - 115200	baudrate on the serial link
Parity	Odd, None, Even	parity
Data bit	7, 8	number of data bits in a byte <b>only for AJ71C24 and AJ71UC24</b>
Stop bit	1, 2	number of stop bits <b>only for AJ71C24 and AJ71UC24</b>
Sum check	Existence, None	enable/disable sumcheck <b>only for AJ71C24 and AJ71UC24</b>
Control	DTR only, RTS only, DTR and RTS, DTR or RTS	handshake signals


 **Note:**  
In any connection form (direct coupling, relaying), where the target station of the UC24 or C24 is the QnACPU, an error is returned, if clock data read/write is executed.

### AJ71C24 / AJ71UC24





For a description of the editable parameters on the page see [PLC side C24 module parameters](#)<sup>[285]</sup>. Pressing the <Next> button opens a [page](#)<sup>[331]</sup> for specifying an optional additional network layer.

	<p><b>Note:</b></p> <p>If the connected station CPU is the AnUCPU and the computer link module is the UC24 for computer link connection, remote operation will result in an error when access is made to the AnNCPU, AnACPU or QnACPU via the MELSECNET/10.</p>
---	---

Connected Station CPU			Relayed Network	Relayed Station CPU						
QCPU (A mode)	QnACPU	ACPU *1		QCPU (Q mode)	QCPU (A mode)	QnACPU	ACPU *1	FX CPU	Q Motion CPU	LCPU
	*2		MELSECNET/H	x	x	x	x	x	x	x
			MELSECNET/10	x	o	*2	o	x	x	x
			MELSECNET(II)	x	o	*2	o	x	x	x
			Ethernet	x	x	x	x	x	x	x
			C24	x	o	*2	o	x	x	x
			CC-Link	x	x	x	x	x	x	x
			CC-Link IE Field	x	x	x	x	x	x	x
			CC-Link IE Control	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

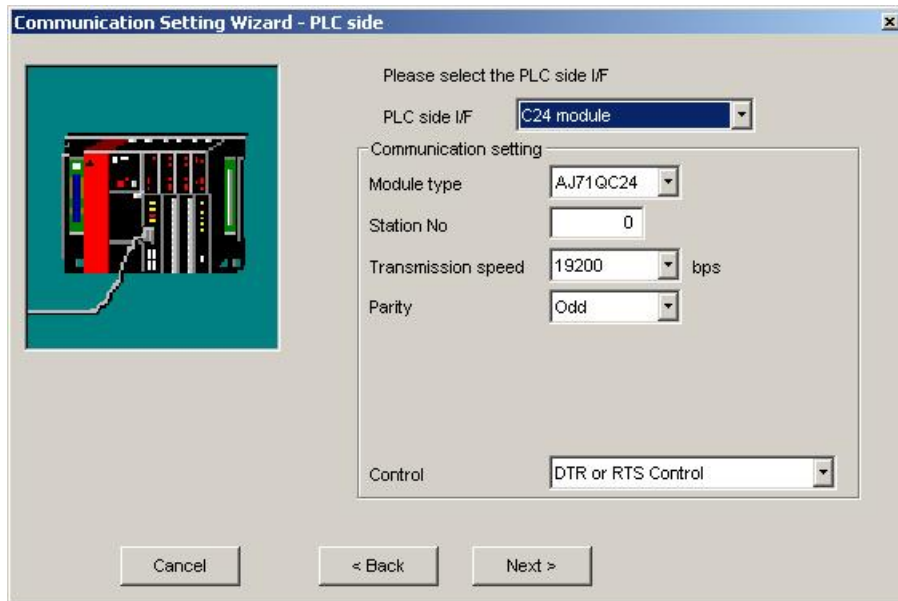
\*2 : Operates as the one equivalent to AnACPU

Switch *1		Settings		
		For 1:1 communication	For 1:n communication	
			Module 1)	Module 2)
Mode setting switch		1 (format 1)	A (format 1)	5 (format 1)
Station number setting switches		0	As set by user	
Transmission specifications setting switches	Main channel setting	OFF (RS-232)	OFF (RS-232)	ON (RS-422)
	Data bit setting	As set by user	As set by user *2	
	Transmission speed setting	As set by user	As set by user *2	
	Parity bit yes/no setting	As set by user	As set by user *2	
	Stop bit setting	As set by user	As set by user *2	
	Sum check yes/no setting	As set by user	As set by user *2	
	Online change enable/disable setting	As set by user		
	Computer link/multidrop setting	ON (computer link)	ON (computer link)	ON (computer link)

\*1: For switch numbers, refer to the computer link module manual.


\*2: Make the same settings to Module 1 and Module 2.

### AJ71QC24



For a description of the editable parameters on the page see [PLC side C24 module parameters](#)<sup>[285]</sup>.

Pressing the <Next> button opens a [page](#)<sup>[331]</sup> for specifying an optional additional network layer.



**Note:**

For the QC24, note that the illegal case of specifying the first I/O number of a nonexistent module and reading/writing U \G will not return an error if the software version of the module is "k" or earlier.

With AJ71QC24-R2, A1SJ71QC4-R2, AJ71QC24N-R2 or A1SJ71QC24N-R2 modules only CH1 can be used.

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
	MELSECNET/H	x	x	x	x	x	x	x
	MELSECNET/10	x	x	o	x	x	x	x
	MELSECNET(II)	x	x	o	x	x	x	x
	Ethernet	x	x	o	x	x	x	x
	C24	x	x	o	x	x	x	x
	CC-Link	x	x	o	x	x	x	x
	CC-Link IE	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

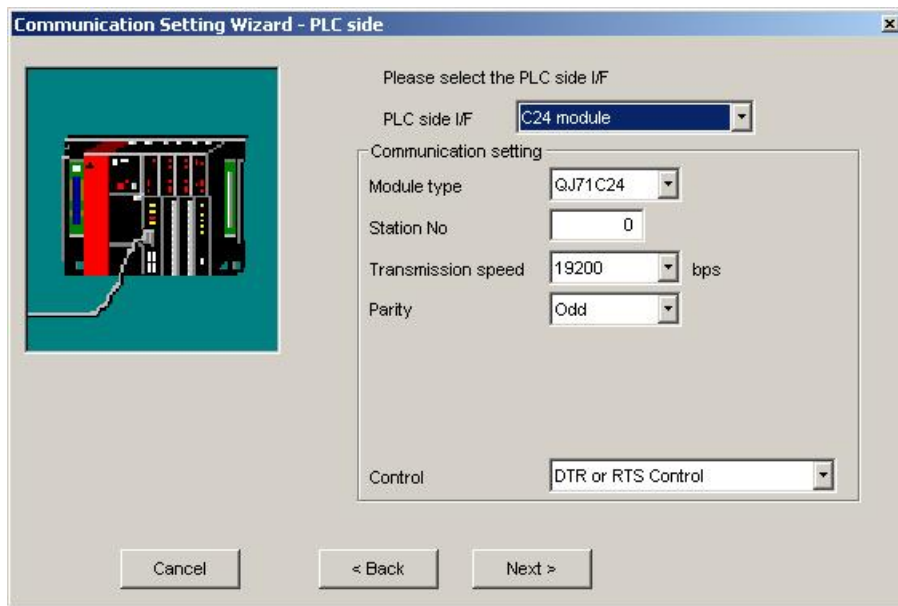
Switch (Switch Number)		Settings					
		For 1:1 communication		For 1:n communication			
		CH1 side	CH2 side	Module 1)		Module 2)	
		CH1 side	CH2 side	CH1 side	CH2 side	CH1 side	CH2 side
Mode setting switch		5 (format 5)		0 or 5 (format 5)	5 (format 5)	5 (format 5)	
Station number setting switch		0		As set by user			
Transmission specification setting switches	Operation setting switch (SW01)	OFF (independent operation)		OFF (independent operation)	ON or OFF *1	OFF (independent operation)	
	Data bit setting (SW02)	ON (8 bit)					
	Parity bit yes/no setting (SW03)	As set by user		As set by user *2			
	Even parity/odd parity setting (SW04)	As set by user		As set by user *2			
	Stop bit setting (SW05)	OFF (1 bit)					
	Sum check yes/no setting (SW06)	ON (yes)					
Online change enable/disable setting		As set by user					

(SW07)		
Setting change enable/disable setting (SW08)	As set by user	As set by user *2
Transmission speed setting (SW09 to SW12)	As set by user	As set by user *2
— (SW13 to SW15)	All OFF	


\*1: Set to ON if the CH1 side mode setting switch setting is 0 or to OFF if the setting is 5 (format 5).

\*2: Make the same settings to Module 1 and Module 2.

### QJ71C24



For a description of the editable parameters on the page see [PLC side C24 module parameters](#)<sup>[285]</sup>. Pressing the <Next> button opens a [page](#)<sup>[331]</sup> for specifying an optional additional network layer.

	<p><b>Note:</b></p> <p>When QJ71C24-R2 of function version A is used, only either of CH1 and CH2 can be used.</p> <p>When the MELSOFT product, such as GX Developer or GOT, is using one channel, the application cannot use the other channel.</p> <p>When the QJ71C24-R2 of function version B is used, the application can use both channels.</p>
---	--

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
	MELSECNET/H	o	x	x	x	x	o	x
	MELSECNET/10	o	o	o	o	x	o	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	o	x	o	x	x	o	x
	C24	o	x	o	x	x	x	o
	CC-Link	o	o	o	o	*3	o	o
	CC-Link IE Field	*2	x	x	x	x	x	o
	CC-Link IE Control	o	x	x	x	x	o	x

o: Accessible

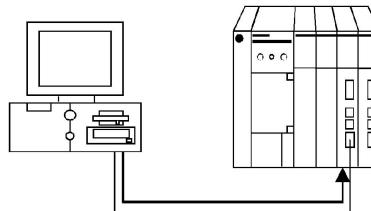
x: Inaccessible

\*1 : Including motion controller CPU

\*2 : For the Redundant PLC, access is disabled

\*3 : partially supported, depends on CPU type

(a) For 1:1 communication



Item	Settings		Set Value
	b15 to b8	b7 to b0	
Switch 1	CH1 communication speed	CH1 transmission setting *1	0000H
Switch 2	—	CH1 communications protocol	0000H
Switch 3	CH2 communication speed	CH2 transmission setting *1	0000H
Switch 4	—	CH2 communications protocol	0000H
Switch 5	Module station number		0000H

\*1: Settings of CH1 and CH2 are indicated below.

Bit	Description	CH1 transmission setting	CH2 transmission setting
b0	Operation setting	0 (independent)	0 (independent)
b1	Data bit	0 (7)	0 (7)

Bit	Description	CH1 transmission setting	CH2 transmission setting
b2	Parity bit	0 (no) *3	0 (no) *3
b3	Odd/even parity	0 (odd) *3	0 (odd) *3
b4	Stop bit	0 (1)	0 (1)
b5	Sum check code	0 (no)	0 (no)
b6	Online change *2	0 (disable)	0 (disable)
b7	Setting change	0 (disable)	0 (disable)

\*2: When the communication protocol is set to GX Developer connection

(0H), the online change bit (b6) setting is made invalid to enable online change regardless of the online change setting.

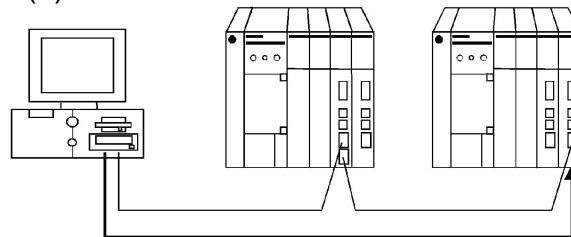
\*3: Set the followings to odd:

"Parity bit" of the communication setting utility/"ActParity" of the ActQJ71C24 control property.

Setting CH1/CH2 communication protocol to GX Developer connection (0H) makes the communication speed/transmission settings to 0H (all OFF).

Refer to the Q series-compatible C24 manual for details.

(b) For 1:n communication



Module 1)

Item	Settings		Set Value
	b15 to b8	b7 to b0	Synchronous operation
Switch 1	CH1 communication speed	CH1 transmission setting *1	0726H
Switch 2	—	CH1 communications protocol	0008H
Switch 3	CH2 communication speed	CH2 transmission setting *1	0727H
Switch 4	—	CH2 communications protocol	0000H
Switch 5	Module station number		As set by user

\*1: Settings of CH1 and CH2 are indicated below.



Bit	Description	Settings Synchronous operation	
		CH1 transmission setting	CH2 transmission setting
b0	Operation setting	Match to module 2) setting.	
b1	Data bit	Match to module 2) setting.	
b2	Parity bit	Match to module 2) setting.	
b3	Odd/even parity	Match to module 2) setting.	
b4	Stop bit	Match to module 2) setting.	
b5	Sum check code	Match to module 2) setting.	
b6	Online change	Match to module 2) setting.	
b7	Setting change	Match to module 2) setting.	

Module 2)

Item	Settings		Set Value
	b15 to b8	b7 to b0	Synchronous operation
Switch 1	CH1 communication speed	CH1 transmission setting *1	0726H
Switch 2	—	CH1 communications protocol	0008H
Switch 3	CH2 communication speed	CH2 transmission setting *1	0727H
Switch 4	—	CH2 communications protocol	0000H
Switch 5	Module station number		As set by user

\*1: Settings of CH1 and CH2 are indicated below.

Bit	Description	CH1 transmission setting	CH2 transmission setting
b0	Operation setting	0 (independent)	1 (synchronous)
b1	Data bit	1 (8)	1 (8)
b2	Parity bit	1 (yes)	1 (yes)

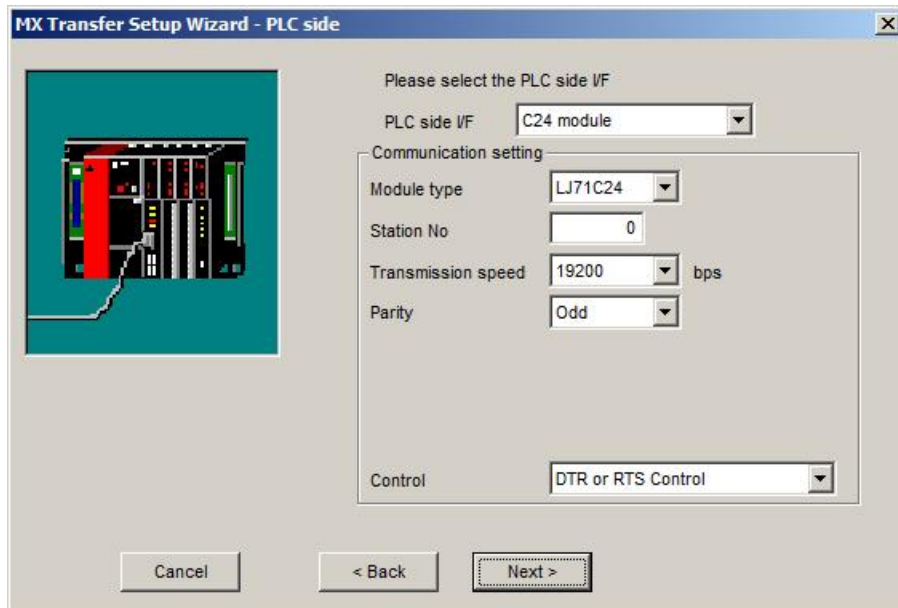
Bit	Description	CH1 transmission setting	CH2 transmission setting
b3	Odd/even parity	0 (odd)	0 (odd)
b4	Stop bit	0 (1)	0 (1)
b5	Sum check code	1 (yes)	1 (yes)
b6	Online change *2	0 (disable)	0 (disable)
b7	Setting change	0 (disable)	0 (disable)

\*2: When the communication protocol is set to GX Developer connection (0H), the online change bit (b6) setting is made invalid to enable online change regardless of the online change setting.

Refer to the Q series-compatible C24 manual for details.

Switch (Switch Number)		Settings																							
		Module 1)		Module 2)																					
		CH1 side	CH2 side	CH1 side	CH2 side																				
Mode setting switch		0	5 (format 5)	5 (format 5)																					
Station number setting switches		1		3																					
Transmission specifications setting switches	Operation setting switch (SW01)	OFF (independent operation)	ON (synchronous operation)	OFF (independent operation)																					
	Data bit setting (SW02)	ON (8 bit)		ON (8 bit)																					
	Parity bit yes/no setting (SW03)	ON (yes)		ON (yes)																					
	Even parity/odd parity setting (SW04)	OFF (odd)		OFF (odd)																					
	Stop bit setting (SW05)	OFF (1 bit)		OFF (1 bit)																					
	Sum check yes/no setting (SW06)	ON (yes)		ON (yes)																					
	Online change enable/disable setting (SW07)	ON (enable)		ON (enable)																					
	Setting change enable/disable setting (SW08)	OFF (disable)		OFF (disable)																					
	Transmission speed setting (SW09 to SW12)	19200bps <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>SW</th> <th>Setting</th> </tr> </thead> <tbody> <tr> <td>SW09</td> <td>OFF</td> </tr> <tr> <td>SW10</td> <td>ON</td> </tr> <tr> <td>SW11</td> <td>ON</td> </tr> <tr> <td>SW12</td> <td>OFF</td> </tr> </tbody> </table>		SW	Setting	SW09	OFF	SW10	ON	SW11	ON	SW12	OFF	19200bps <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>SW</th> <th>Setting</th> </tr> </thead> <tbody> <tr> <td>SW09</td> <td>OFF</td> </tr> <tr> <td>SW10</td> <td>ON</td> </tr> <tr> <td>SW11</td> <td>ON</td> </tr> <tr> <td>SW12</td> <td>OFF</td> </tr> </tbody> </table>		SW	Setting	SW09	OFF	SW10	ON	SW11	ON	SW12	OFF
	SW	Setting																							
SW09	OFF																								
SW10	ON																								
SW11	ON																								
SW12	OFF																								
SW	Setting																								
SW09	OFF																								
SW10	ON																								
SW11	ON																								
SW12	OFF																								
— (SW13 to SW15)	All OFF		All OFF																						

### LJ71C24



For a description of the editable parameters on the page see [PLC side C24 module parameters](#)<sup>[285]</sup>. Pressing the <Next> button opens a [page](#)<sup>[331]</sup> for specifying an optional additional network layer.

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
	MELSECNET/H	x	x	x	x	x	x	x
	MELSECNET/10	x	x	x	x	x	x	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	x	x	x	x	x	x	x
	C24	o	x	o	x	x	x	o
	CC-Link	o	o	o	o	x	x	o
	CC-Link IE Field	*2	x	x	x	x	x	o
	CC-Link IE Control	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

\*2 : For the Redundant PLC, access is disabled


\*3 : partially supported, depends on CPU type

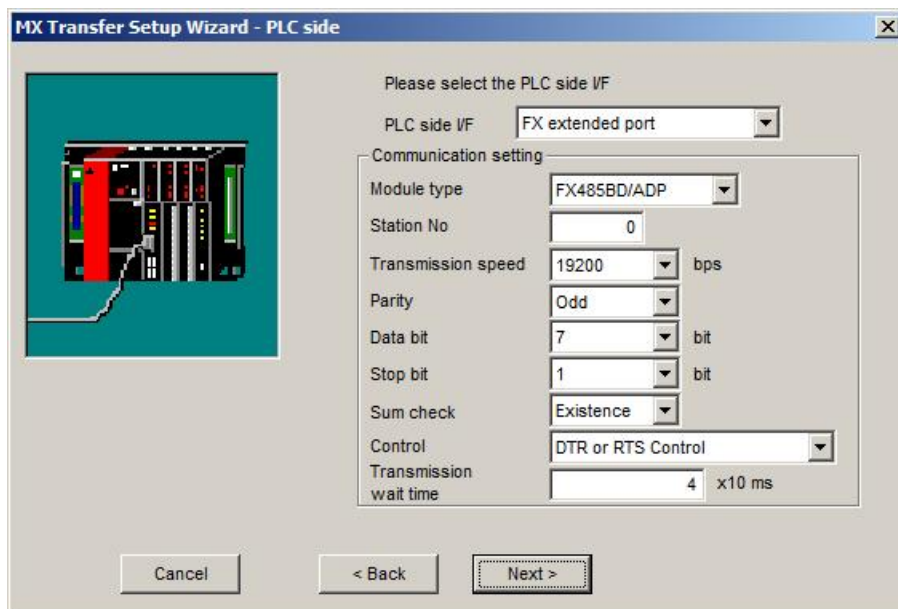
### 11.2.20 PLC FX Extended Port

Enter the settings for the serial port of an FX CPU.

**PLC side FX port parameters**

Parameter	Values	Description
Module type	FX485BD/ADP	type of PLC C24 module
Station No.	0 - 15	station number of the PLC side interface
Transmission speed	300 - 19200	baudrate on the serial link
Parity	Odd, None, Even	parity
Data bit	7, 8	number of data bits in a byte
Stop bit	1, 2	number of stop bits
Sum check	Existence, None	enable/disable sumcheck
Control	DTR only, RTS only, DTR and RTS, DTR or RTS	handshake signals
Transmission wait time	1 - 65535	transmission timeout as multiple of 10 milliseconds

**Note:**  
 The FX extended port is required when using C24 communication with FX0N, FX1S, FX1N(C), FX2N(C), FX3U(C), FX3G (C/E), FX3S CPU.



For a description of the editable parameters on the page see [PLC side FX port parameters](#)<sup>[297]</sup>. Pressing the <Next> button opens a [page](#)<sup>[331]</sup> for specifying an optional additional network layer.

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
	MELSECNET/H	x	x	x	x	x	x	x
	MELSECNET/10	x	x	x	x	x	x	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	x	x	x	x	x	x	x
	C24	x	x	x	x	o	x	x
	CC-Link	x	x	x	x	x	x	x
	CC-Link IE	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

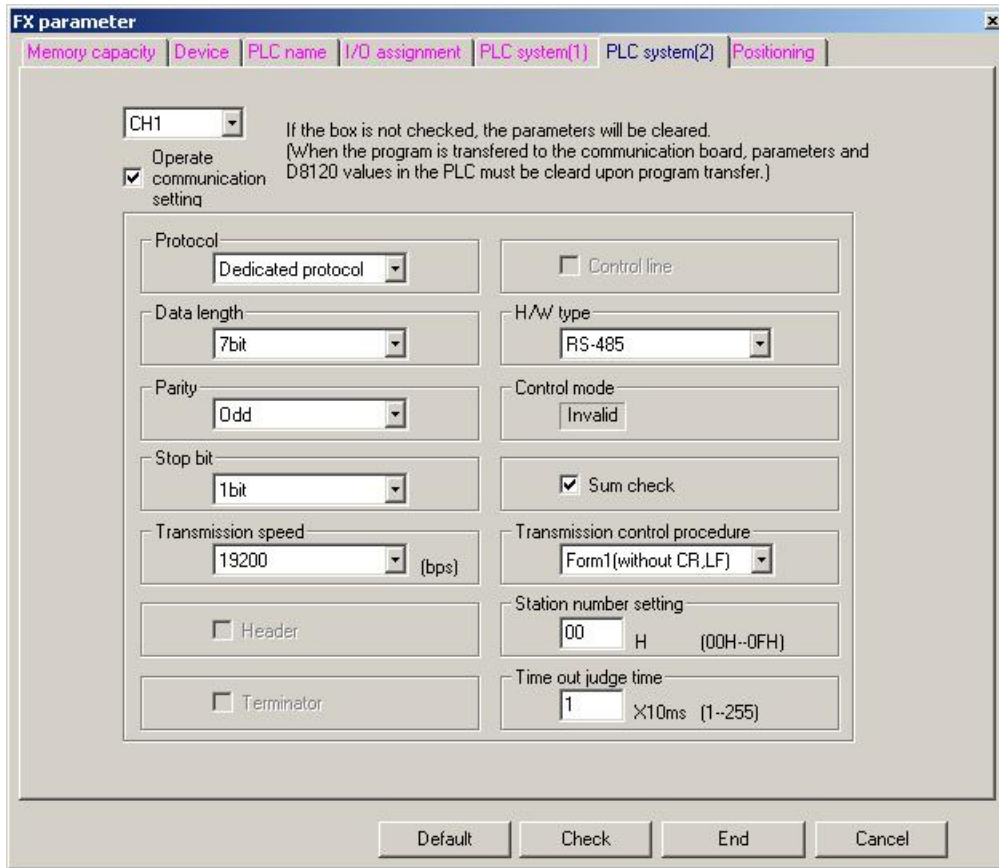
\*1 : Including motion controller CPU

Before use the FX extended port must be configured for communication. There are two ways of configuring the FX extended port module:


1. use PLC parameters
2. write values to special data registers (D8120, D8121, D8129) in a sequence program. This is the only approach for FX0N CPUs.

### Set PLC parameters

Start GX Developer or GX IEC Developer and select [Parameter]-[PLC parameter] in the project list. Then select the "PLC system (2)" tab.



Item	Setting value
Operate communication setting	Check the corresponding check box.
Protocol	Dedicated protocol
Data length	As set by user.
Parity	As set by user.
Stop bit	As set by user.
Transmission speed	As set by user.
H/W type	RS-485
Sum check	As set by user.
Transmission control procedure	Form 1
Station number setting	00H to 0FH
Time out judge time	1 to 255

	<p><b>Note:</b></p> <ul style="list-style-type: none"> <li>When communication setting is made, power the FX CPU again after writing to the PLC.</li> <li>When performing multi-drop connection, make the same communication settings for the devices. However, make sure that the station numbers do not overlap.</li> </ul>
---	--

### Special data registers

1) D8120 (Communication format)

Bit	Description	Setting details																																								
b0	Data length	0: 7 bit 1: 8 bit																																								
b1	Parity	<table border="1"> <tr> <td></td> <td>b2</td> <td>b1</td> </tr> <tr> <td>None</td> <td>0</td> <td>0</td> </tr> <tr> <td>Odd number</td> <td>0</td> <td>1</td> </tr> <tr> <td>Even number</td> <td>1</td> <td>1</td> </tr> </table>		b2	b1	None	0	0	Odd number	0	1	Even number	1	1																												
		b2	b1																																							
None		0	0																																							
Odd number	0	1																																								
Even number	1	1																																								
b2																																										
b3	Stop bit	0: 1 bit 1: 2 bit																																								
b4	Transmission speed	<table border="1"> <tr> <td></td> <td>b7</td> <td>b6</td> <td>b5</td> <td>b4</td> </tr> <tr> <td>300bps</td> <td>0</td> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>600bps</td> <td>0</td> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1200bps</td> <td>0</td> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>2400bps</td> <td>0</td> <td>1</td> <td>1</td> <td>0</td> </tr> <tr> <td>4800bps</td> <td>0</td> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>9600bps</td> <td>1</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>19200bps</td> <td>1</td> <td>0</td> <td>0</td> <td>1</td> </tr> </table>		b7	b6	b5	b4	300bps	0	0	1	1	600bps	0	1	0	0	1200bps	0	1	0	1	2400bps	0	1	1	0	4800bps	0	1	1	1	9600bps	1	0	0	0	19200bps	1	0	0	1
		b7	b6	b5	b4																																					
300bps		0	0	1	1																																					
600bps		0	1	0	0																																					
1200bps		0	1	0	1																																					
2400bps		0	1	1	0																																					
4800bps	0	1	1	1																																						
9600bps	1	0	0	0																																						
19200bps	1	0	0	1																																						
b5																																										
b6																																										
b7																																										
b8	---	0																																								
b9	---	0																																								
b10	H/W type	<table border="1"> <tr> <td></td> <td>b11</td> <td>b10</td> </tr> <tr> <td>RS-485</td> <td>0</td> <td>0</td> </tr> </table>		b11	b10	RS-485	0	0																																		
		b11	b10																																							
RS-485	0	0																																								
b11																																										
b12	---	0																																								
b13	Sum check	0: N/A 1: Available																																								
b14	Communication protocol	1: Computer link																																								
b15	Transmission control procedure	0: Form 1																																								

2) D8121 (Station No. setting)

Specify the station No. in the range of 00H to 0FH




3) D8129 (Time out judge time setting)

Specify the FX CPU timeout as a multiple of 10ms. The range depends on the CPU type:

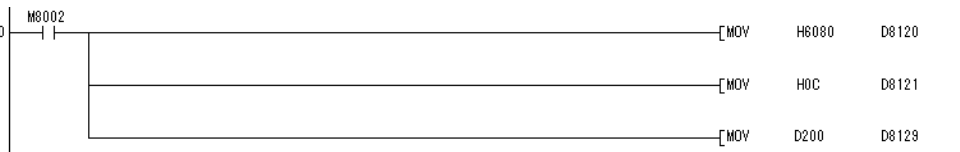
CPU Type	Range
FX0N, FX1S and FX1N(C)	1 to 255 (10 to 2550ms)
FX2N(C)	1 to 3276 (10 to 32760ms)

A value of 0 corresponds to 100ms.

	<p><b>Note:</b></p> <ul style="list-style-type: none"> <li>· when communication settings have been downloaded, reset the FX CPU</li> <li>· when setting up a multi-drop connection, apply the same communication settings for all devices. However make sure that the station numbers do not overlap.</li> </ul>
---	--

REMARK

The following shows an example of setting values to the special data registers by using GX Developer.



The following shows the setting details of each special data register in the above program.

(1) D8120

Transmission control procedure : Form 1

Communication protocol : Computer link

Sum check : Available

H/W type : RS-485

Transmission speed : 9600bps Stop bit : 1 bit Parity : N/A

Data length : 7 bit

(2) D8121

Station No. : 12

(3) D8129

Time out time : 200ms

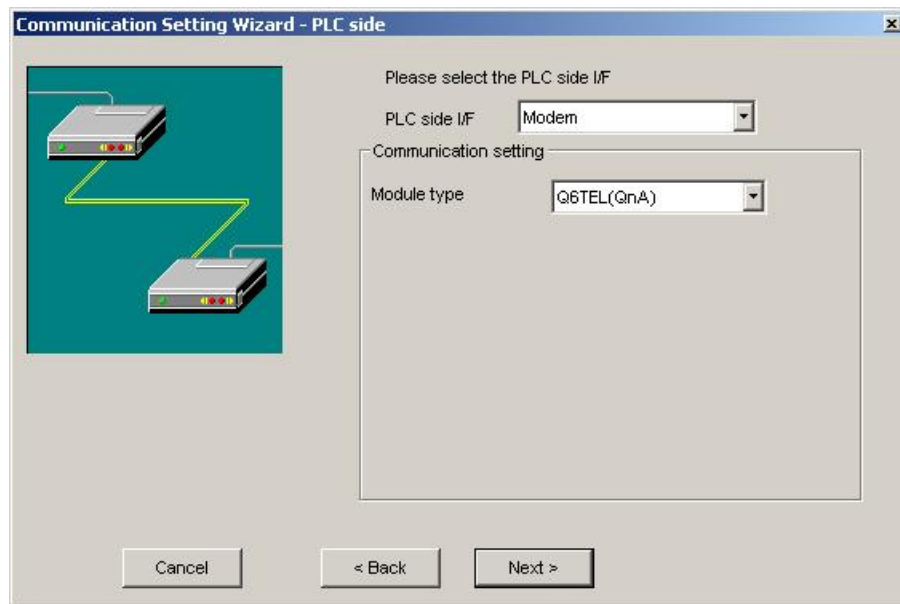
### 11.2.21 PLC Modem

Enter the settings for a modem interface module in the PLC.

#### *PLC side modem parameters*

Parameter	Values	Description
Module type	<a href="#">Q6TEL(QnA)</a> <sup>[302]</sup> , <a href="#">FXCPU</a> <sup>[305]</sup> , <a href="#">AJ71QC24N</a> <sup>[303]</sup> , <a href="#">QJ71C24</a> <sup>[303]</sup> , <a href="#">QJ71CMO</a> <sup>[303]</sup> , <a href="#">LJ71C24</a> <sup>[306]</sup>	type of PLC modem or serial interface module
Station No.	0 - 31	station number of the PLC side interface <b>only for AJ71QC24N, QJ71C24, QJ71CMO</b>
Transmission speed	300 - 115200	baudrate on the serial link <b>only for AJ71QC24N, QJ71C24, QJ71CMO</b>

### Q6TEL



For a description of the editable parameters on the page see [PLC side modem parameters](#)<sup>[302]</sup>. Pressing the <Next> button opens a [page](#)<sup>[333]</sup> for specifying an optional additional network layer.

### Q6TEL CPU

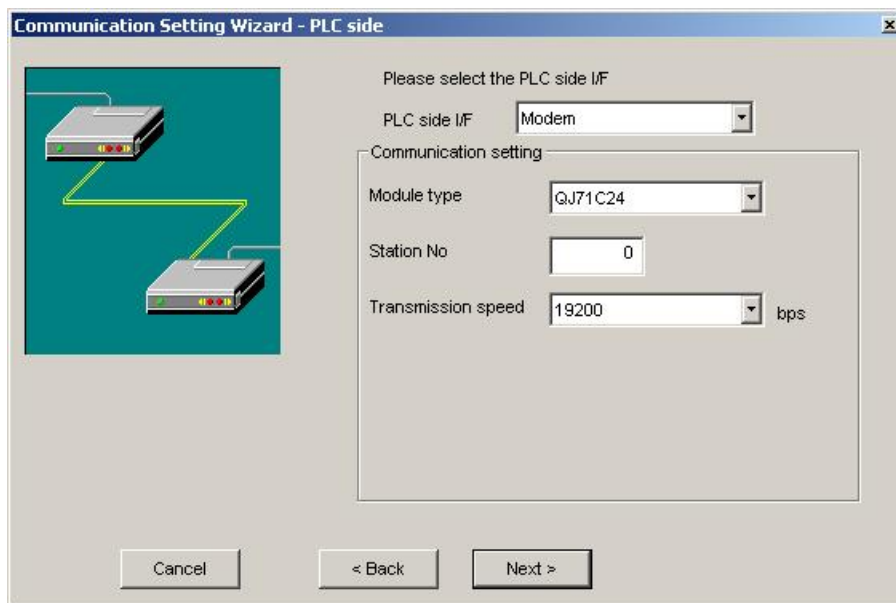
Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
0	MELSECNET/H	x	x	x	x	x	x	x
	MELSECNET/10	x	x	o	x	x	x	x
	MELSECNET(II)	x	x	o	x	x	x	x
	Ethernet	x	x	o	x	x	x	x
	C24	x	x	o	x	x	x	x
	CC-Link	x	x	x	x	x	x	x
	CC-Link IE	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

### AJ71QC24N / QJ71C24 / QJ71CMO



For a description of the editable parameters on the page see [PLC side modem parameters](#)<sup>[302]</sup>.

Pressing the <Next> button opens a [page](#)<sup>[333]</sup> for specifying an optional additional network layer.

### QJ71C24 CPU

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
o	MELSECNET/H	o	x	x	x	x	x	x
	MELSECNET/10	o	o	o	o	x	x	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	o	x	o	x	x	x	x
	C24	o	x	o	x	x	x	o
	CC-Link	o	o	o	o	x	x	o
	CC-Link IE Field	*2	x	x	x	x	x	o
	CC-Link IE Control	o	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

\*2 :For the Redundant CPU, access is disabled

### AJ71QC24N CPU

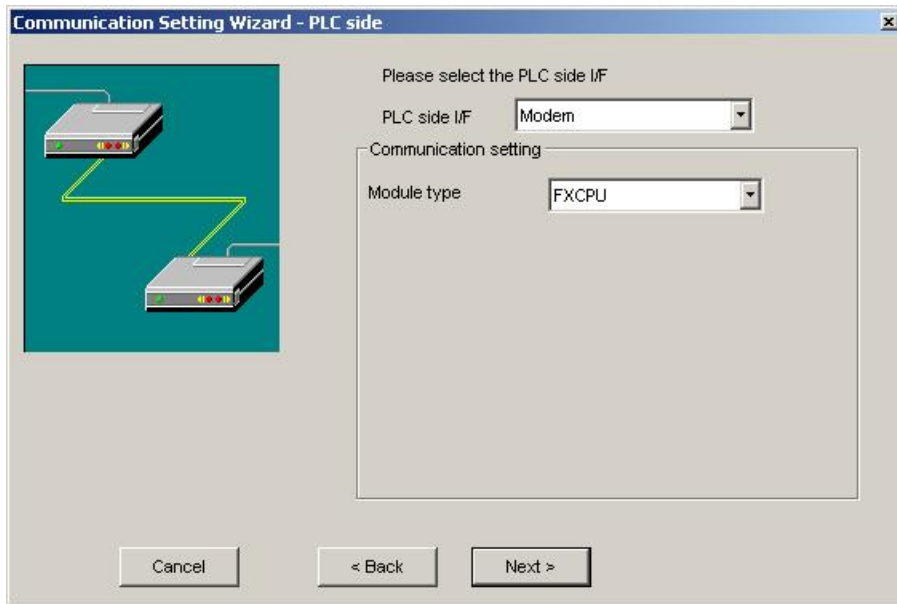
Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
o	MELSECNET/H	x	x	x	x	x	x	x
	MELSECNET/10	x	x	o	x	x	x	x
	MELSECNET(II)	x	x	o	x	x	x	x
	Ethernet	x	x	o	x	x	x	x
	C24	x	x	o	x	x	x	x
	CC-Link	x	x	o	x	x	x	x
	CC-Link IE Field	x	x	x	x	x	x	x
	CC-Link IE Control	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

### FX CPU



For a description of the editable parameters on the page see [PLC side modem parameters](#)<sup>[302]</sup>. Pressing the <Next> button opens a [page](#)<sup>[333]</sup> for specifying an optional additional network layer.

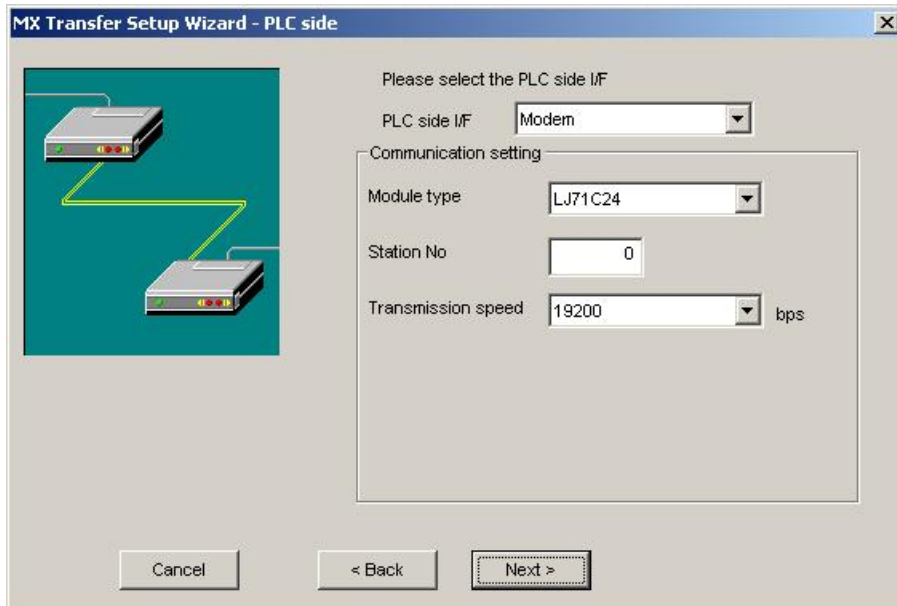
Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
o	MELSECNET/H	x	x	x	x	x	x	x
	MELSECNET/10	x	x	x	x	x	x	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	x	x	x	x	x	x	x
	Computer link	x	x	x	x	x	x	x
	CC-Link	x	x	x	x	x	x	x
	CC-Link IE	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

### LJ71C24



For a description of the editable parameters on the page see [PLC side modem parameters](#)<sup>[302]</sup>.

Pressing the <Next> button opens a [page](#)<sup>[333]</sup> for specifying an optional additional network layer.

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
0	MELSECNET/H	x	x	x	x	x	x	x
	MELSECNET/10	x	x	x	x	x	x	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	x	x	x	x	x	x	x
	C24	o	x	o	x	x	x	o
	CC-Link	o	o	o	o	x	x	o
	CC-Link IE Field	*2	x	x	x	x	x	o
CC-Link IE Control		x	x	x	x	x	x	

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU


\*2 :For the Redundant CPU, access is disabled


### 11.2.22 PLC Ethernet Interface

Enter the settings for an Ethernet module in the PLC.

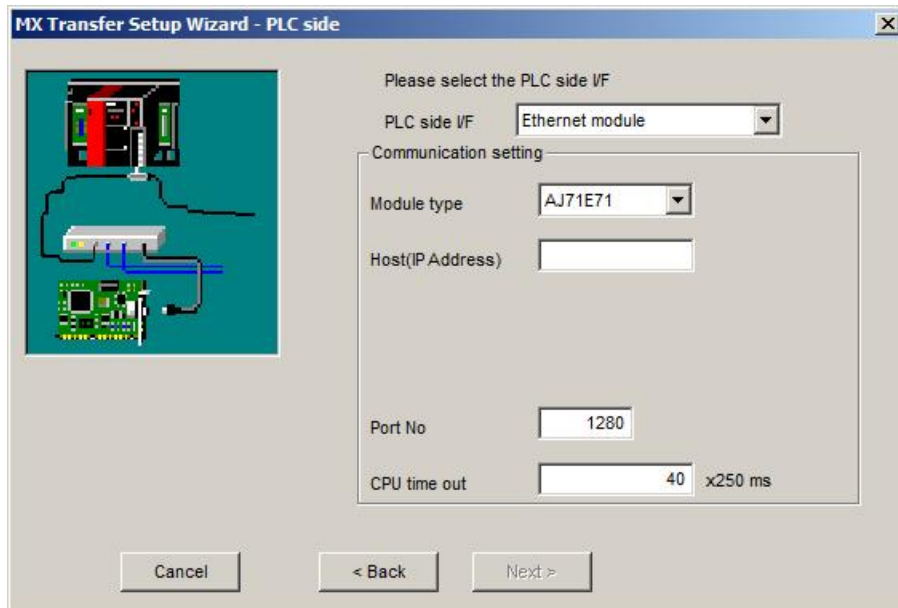
#### *PLC side Ethernet module parameters*

Parameter	Values	Description
Connect module	<a href="#">AJ71E71</a> <sup>[308]</sup> , <a href="#">AJ71QE71</a> <sup>[309]</sup> , <a href="#">QJ71E71</a> <sup>[310]</sup> , <a href="#">GOT</a> <sup>[312]</sup> , <a href="#">FX-ENET(-ADP)</a> <sup>[312]</sup> , <a href="#">FX3U-ENET-ADP</a> <sup>[316]</sup> , <a href="#">CPU module</a> <sup>[317]</sup> , <a href="#">CC</a> <a href="#">IE Field Ethernet</a> <a href="#">Adapter</a> <sup>[321]</sup>	type of PLC ethernet module
Host (IP address)	nnn.nnn.nnn.nnn	IP address of the PLC side interface. This value must equal the address, which has been configured in the 'Network Parameters' of the PLC settings.
Network No	1 - 239	network number of the E71 ethernet module. This value can only be edited in the ' <a href="#">PC Ethernet board</a> ' <sup>[236]</sup> page. <b>only for AJ71QE71, QJ71E71 modules</b>
Station No.	1 - 64	station number of the PLC side interface. This value must equal the station number, which has been configured in the 'Network Parameters' of the PLC settings. <b>only for AJ71QE71, QJ71E71 modules</b>
Port No.	1 - 65535	free UDP port number on the PC side for data sent by the PLC. Port numbers below 1025 should not be used. <b>only for AJ71E71 (UDP and TCP), AJ71QE71 (TCP), FX-ENET-ADP modules</b>
CPU time out	1 - 65535	CPU timeout as multiple of 250 milliseconds <b>only for AJ71E71 modules</b>

 **Note:**  
The 'Next >' button is disabled, if no IP address has been entered.

 **Note:**  
Replacement of Ethernet modules  
If the PLC Ethernet module has been replaced during Ethernet communication, the communication on the PC must be restarted, because Ethernet (MAC) address is different.

### AJ71E71



For a description of the editable parameters on the page see [PLC side Ethernet module parameters](#)<sup>[307]</sup>. Pressing the <Next> button opens a [page](#)<sup>[332]</sup> for specifying an optional additional network layer.

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
0	MELSECNET/H	x	x	x	x	x	x	x
	MELSECNET/10	x	o	o	o	x	x	x
	MELSECNET(II)	x	o	o	o	x	x	x
	Ethernet	x	x	x	x	x	x	x
	C24	x	x	x	x	x	x	x
	CC-Link	x	x	x	x	x	x	x
	CC-Link IE	x	x	x	x	x	x	x

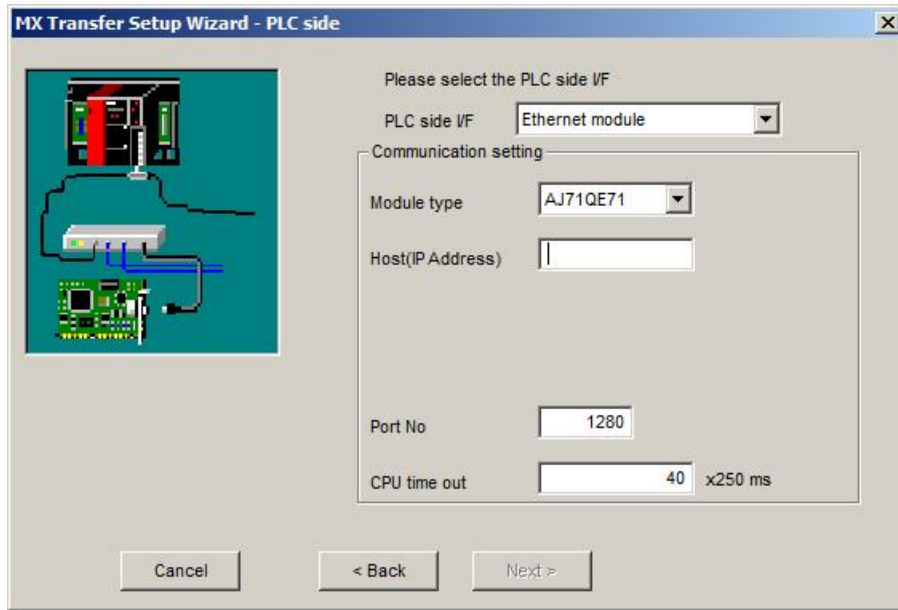
o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU



### AJ71QE71



For a description of the editable parameters on the page see [PLC side Ethernet module parameters](#)<sup>[307]</sup>. Pressing the <Next> button opens a [page](#)<sup>[332]</sup> for specifying an optional additional network layer.

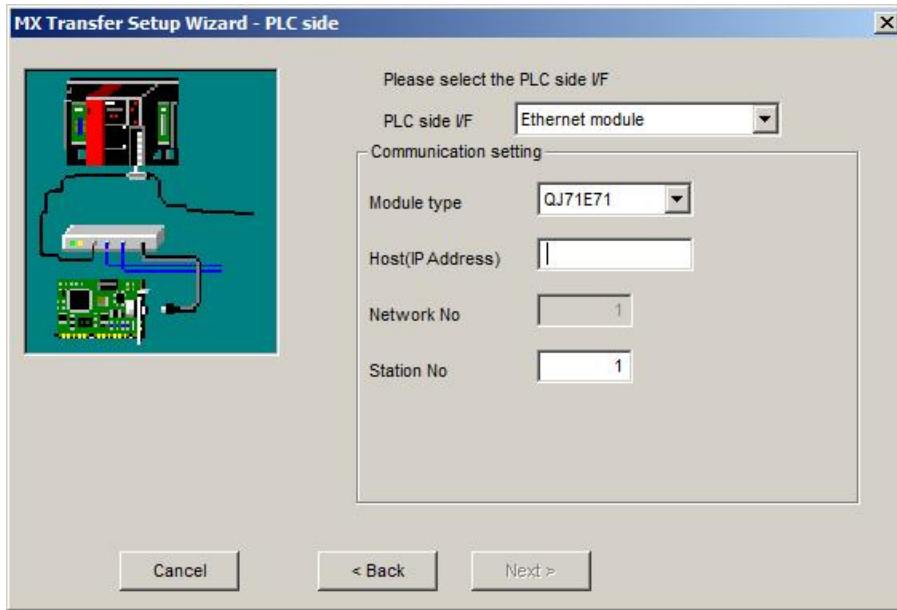
Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
o	MELSECNET/H	x	x	x	x	x	x	x
	MELSECNET/10	x	x	o	x	x	x	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	x	x	o	x	x	x	x
	C24	x	x	o	x	x	x	x
	CC-Link	x	x	x	x	x	x	x
	CC-Link IE	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

**QJ71E71**



For a description of the editable parameters on the page see [PLC side Ethernet module parameters](#)<sup>[307]</sup>. Pressing the <Next> button opens a [page](#)<sup>[332]</sup> for specifying an optional additional network layer.

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
0	MELSECNET/H	o	x	x	x	x	o	x
	MELSECNET/10	o	o	o	o	x	o	x
	MELSECNET(II)	x	o	o	o	x	x	x
	Ethernet	o	x	o	x	x	o	x
	C24	o	x	x	x	x	o	o
	CC-Link	o	o	o	o	x	o	o
	CC-Link IE Field Control	*2	x	x	x	x	x	o

o: Accessible

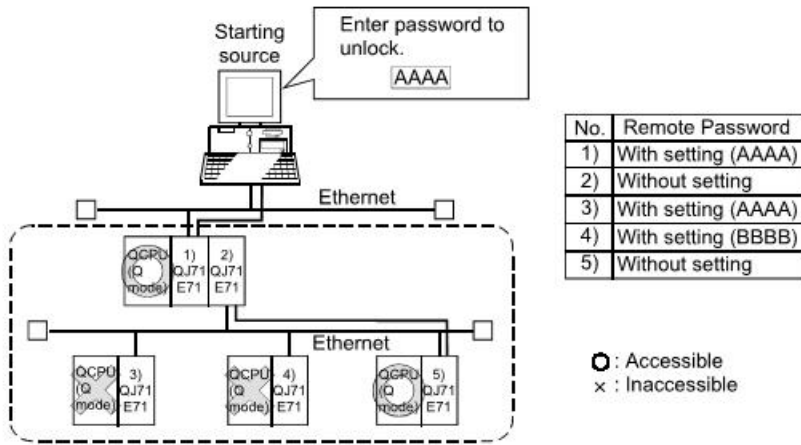
x: Inaccessible

\*1 : Including motion controller CPU

\*2 : For the Redundant CPU, access is disabled

### Unlocking password when using QJ71E71

If a password has been set in a QJ71E71 for remote access, the corresponding PLC cannot be accessed as target PLC. A password has however no effect, when the E71 module is used as a relay.

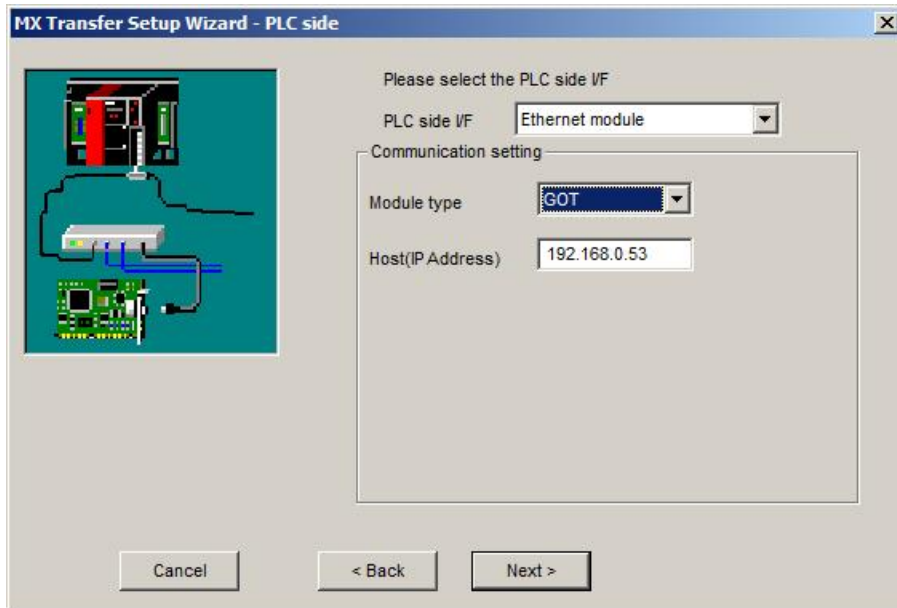


1) Unlocking QJ71E71 password enables access to PLC CPUs in this range.

### Use of Q4AR CPU

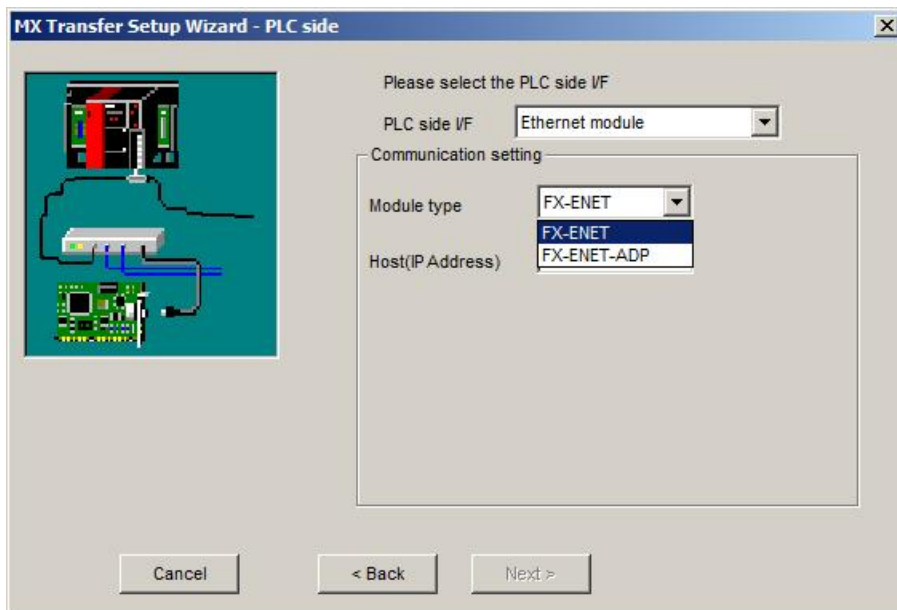
For using UDP/IP protocol the manufacture date (year and month) of the Q4AR CPU must be "0012" or later and the function version must be 'B' at least.

### GOT



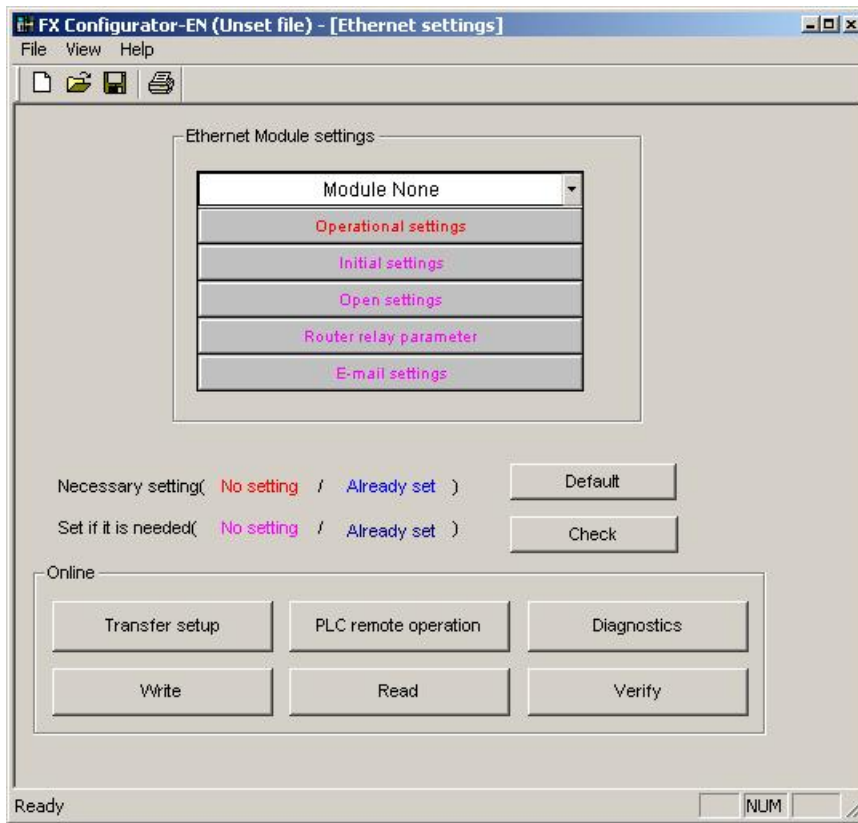
For a description of the editable parameters on the page see [PLC side Ethernet module parameters](#)<sup>[307]</sup>. Pressing the <Next> button opens a [page](#)<sup>[332]</sup> for specifying an optional additional network layer.

### FX Ethernet Module

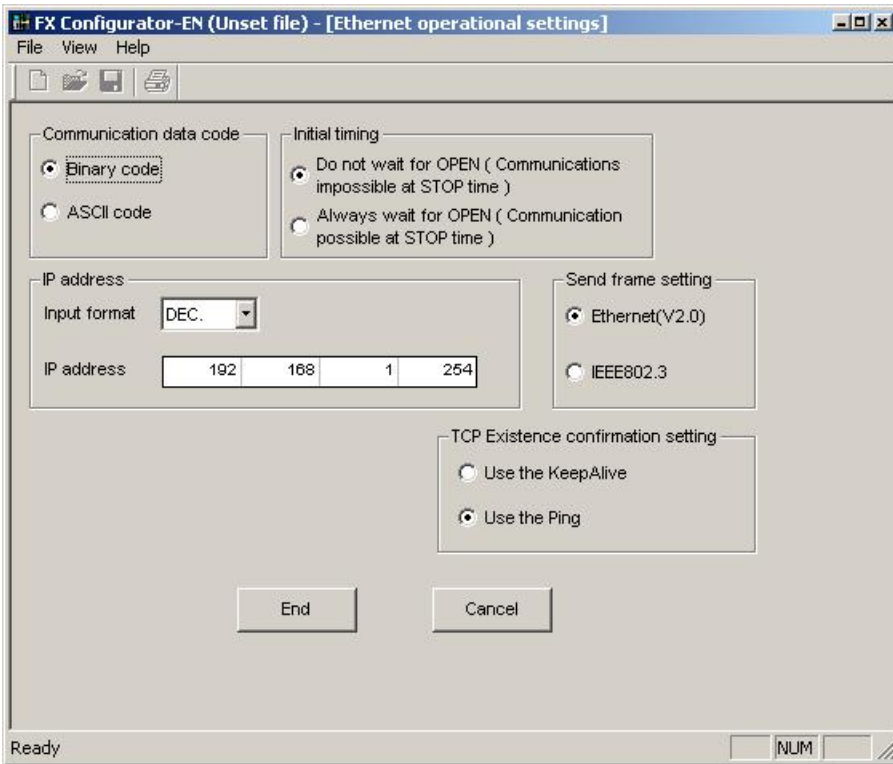


For a description of the editable parameters on the page see [PLC side Ethernet module parameters](#)<sup>[307]</sup>. Pressing the <Next> button opens a [page](#)<sup>[332]</sup> for specifying an optional additional network layer.

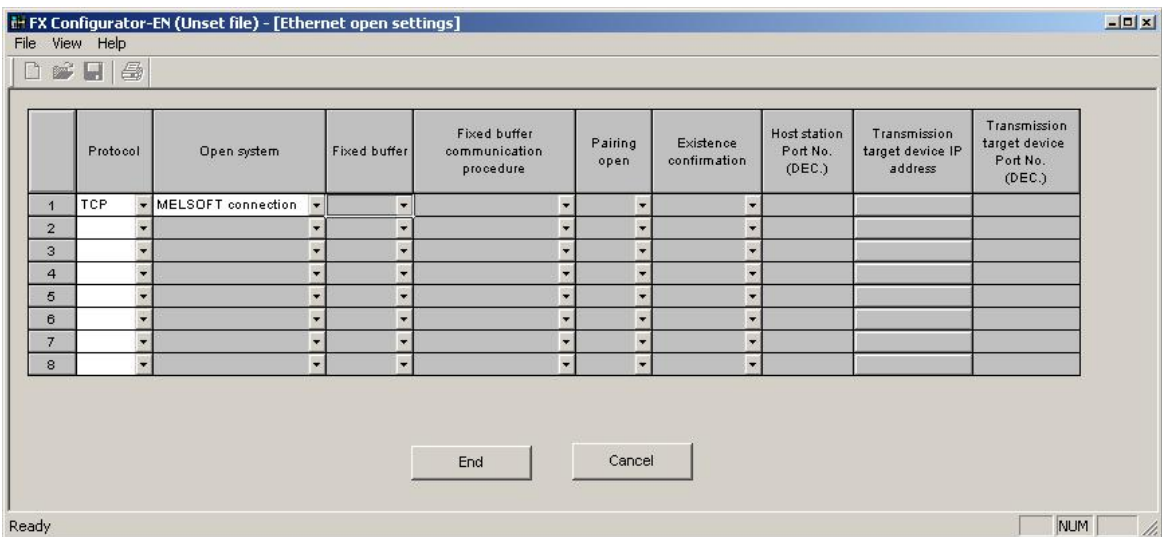
Because GX Developer and GX IEC Developer cannot configure the FX-ENET Ethernet module for FX3U(C) PLCs, there is another tool (FX Configurator-EN) to set up the communication parameters:



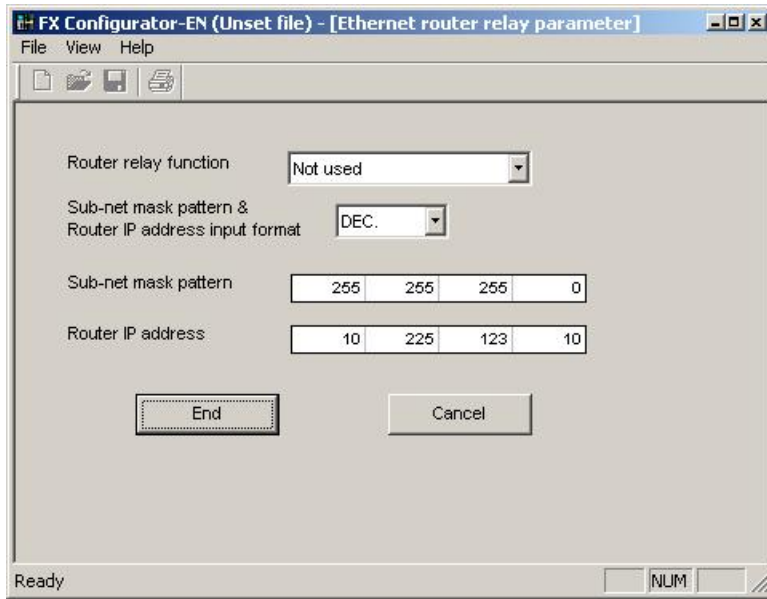
- (1) Select the module, up to 8 modules can be selected.
- (2) Dialog to select the operational setting:



- (3) Set the ethernet open setting:



(4) Dialog to select the router:



(5) Transfer setup is used to download the Ethernet parameter to the FX-ENET. The transfer happens through the PLC, so that the corresponding setting should be selected, for example serial communication to transfer the Ethernet parameters.

The ethernet communication via FX-ENET-ADP for FX1S, FX1N(C) and FX2N(C) is configured through settings in special PLC devices.

Set the ethernet parameters to nine data registers from D1000 to D1008. If these data registers are used for any other purpose, the ethernet parameters can be set to nine registers 'D' starting from D2000, D3000, D4000, D5000, D6000 or D7000.

Data registers	Setting Item	Default parameter	Description
Dn000, Dn001	Header	-	Set H454E4554 ("ENET")
Dn002, Dn003	IP address	192.168.0.100	Set the IP address of the FX-ENET module
Dn004, Dn005	subnet mask	255.255.255.0	Set the sub net mask of the FX-ENET module
Dn006, Dn007	Gateway address	192.168.0.1	Set the gateway address of the FX-ENET module
Dn008	TCP port number	1024	Set the TCP port number within the range from 1024 to 65535

n: Choose any number in the range between 1 and 7

### When a PLC program is used to set the ethernet parameters

The ethernet parameters for the FX-ENET-ADP can be set using the PLC program below:

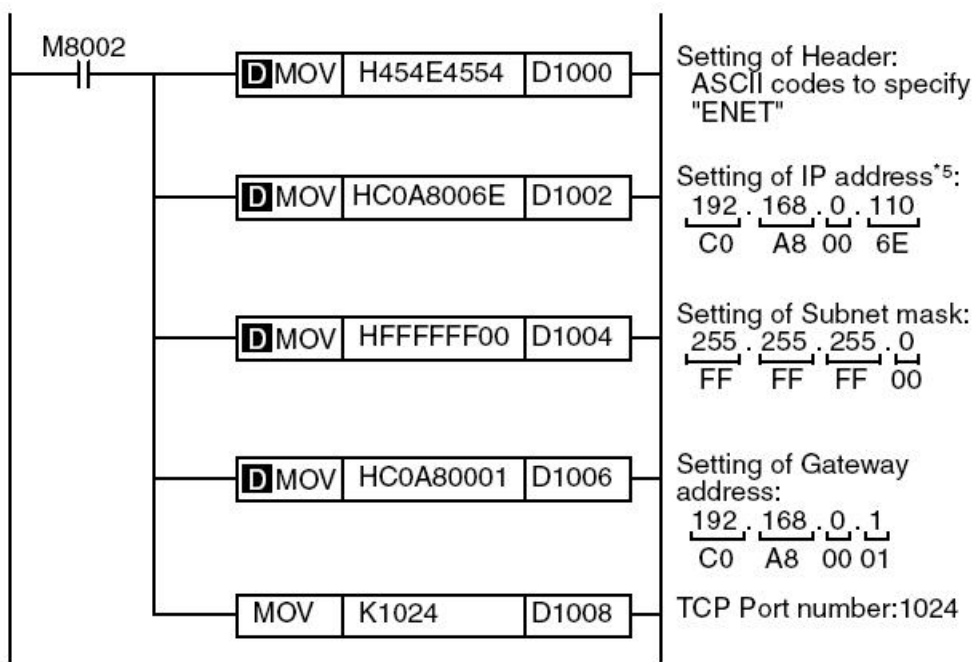
**Note:**

In the FX-ENET-ADP the ethernet parameters become valid only when the power is turned ON and the setting data is stored in specified data registers.



In any of the following cases, turn off the power of the PLC once, and turn it on again.

- when a parameter is set for the first time
- when the setting of a parameter is changed during operation



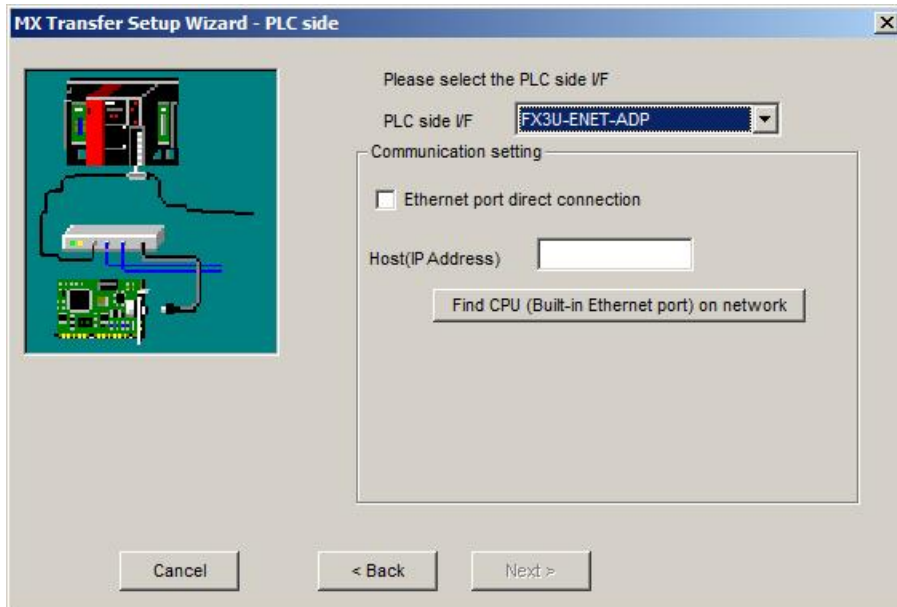
\*5: When connecting two or more PLCs to a network, do not use the IP address twice.

Example:	Personal Computer (GX IEC Developer)	192.168.0.10
	PLC1	192.168.0.110
	PLC2	192.168.0.111

### FX3 Ethernet Module

Use this setting to connect to the built-in ethernet port of an FX3GE CPU.

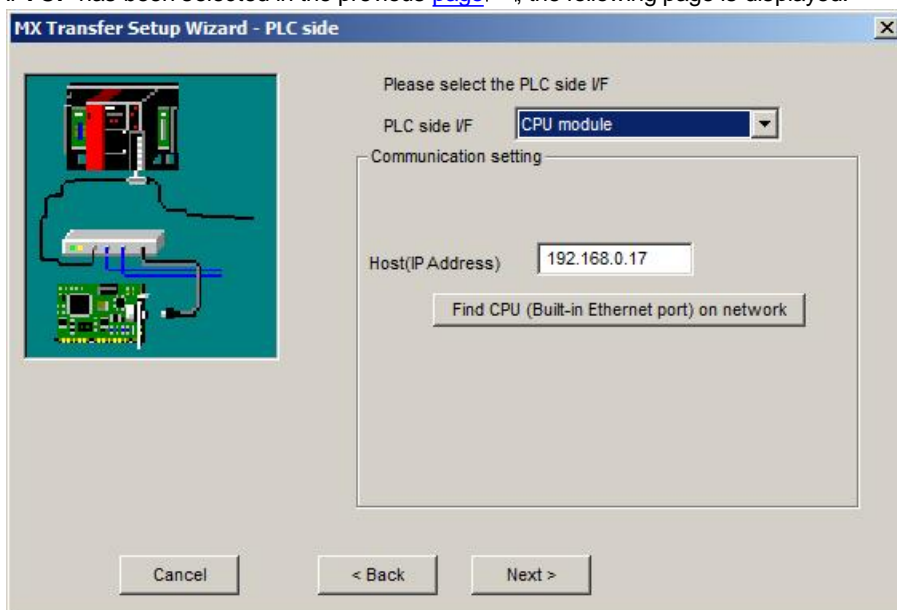




For a description of the editable parameters on the page see [PLC side Ethernet module parameters](#)<sup>[307]</sup>. Pressing the <Next> button opens a [page](#)<sup>[332]</sup> for specifying an optional additional network layer.

### PLC Ethernet CPU

If **TCP** has been selected in the previous [page](#)<sup>[247]</sup>, the following page is displayed:



The IP address of the Ethernet interface can be manually entered or looked up online. Pressing the 'Find CPU' button opens the '[Find CPU](#)<sup>[320]</sup>' dialog, in which the QnUDE and L-series CPUs in the local network are listed.

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
0	MELSECNET/H	o	x	x	x	x	o	x
	MELSECNET/10	o	o	o	o	x	o	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	o	x	o	x	x	o	x
	C24	o	x	x	x	x	o	o
	CC-Link	o	x	x	x	x	o	o
	CC-Link IE Field	*2	x	x	x	x	x	o
	CC-Link IE Control	o	x	x	x	x	x	o

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

\*2 : For the Redundant CPU, access is disabled

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
0	MELSECNET/H	x	x	x	x	x	x	x
	MELSECNET/10	x	x	x	x	x	x	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	x	x	x	x	x	x	x
	C24	o	x	x	x	x	x	o
	CC-Link	o	x	x	x	x	x	o
	CC-Link IE Field	*2	x	x	x	x	x	o
	CC-Link IE Control	x	x	x	x	x	x	x

o: Accessible

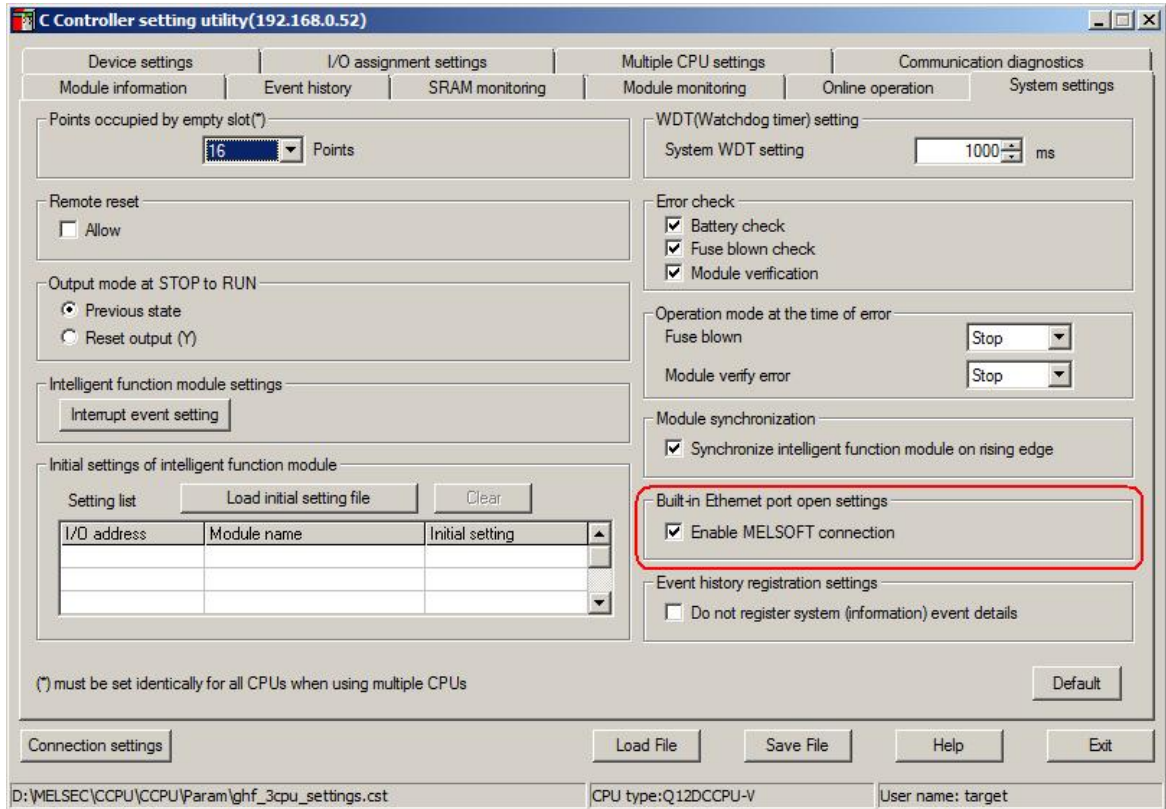
x: Inaccessible

\*1 : Including motion controller CPU

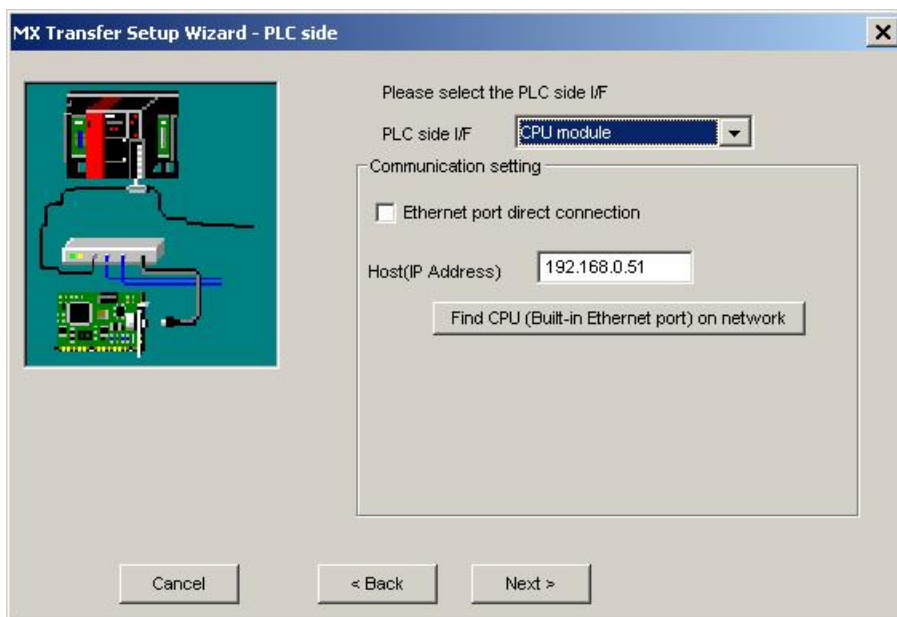
\*2 : For the Redundant CPU, access is disabled

### Connections to C-Controller (Q12DCCPU-V)

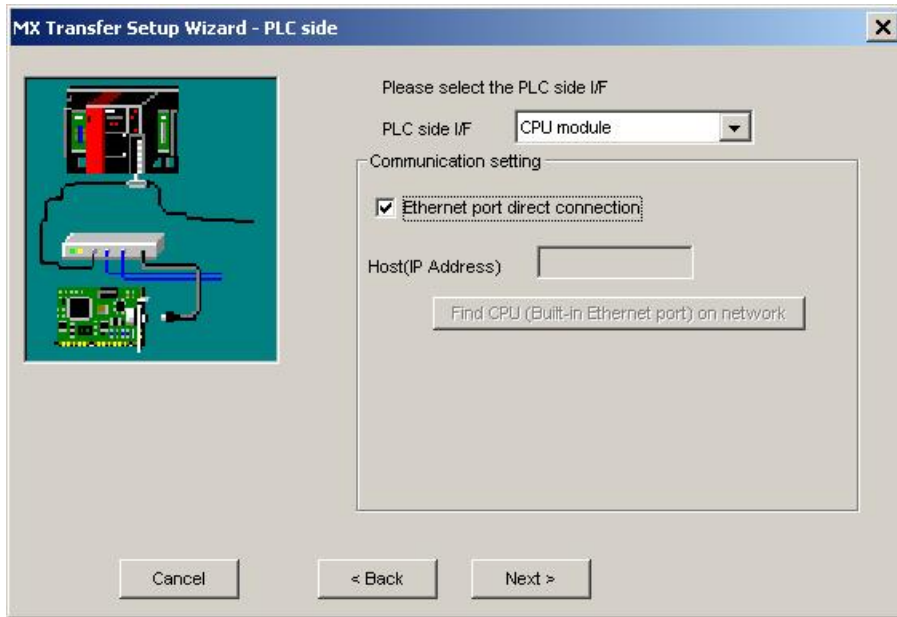
The 'Find CPU'<sup>[320]</sup> function Q12DCCPU-V does not show Q12DCCPU units in the network. Their IP-address must be entered manually and MELSOFT connections must be explicitly enabled with the 'C-Controller Setting Utility'.



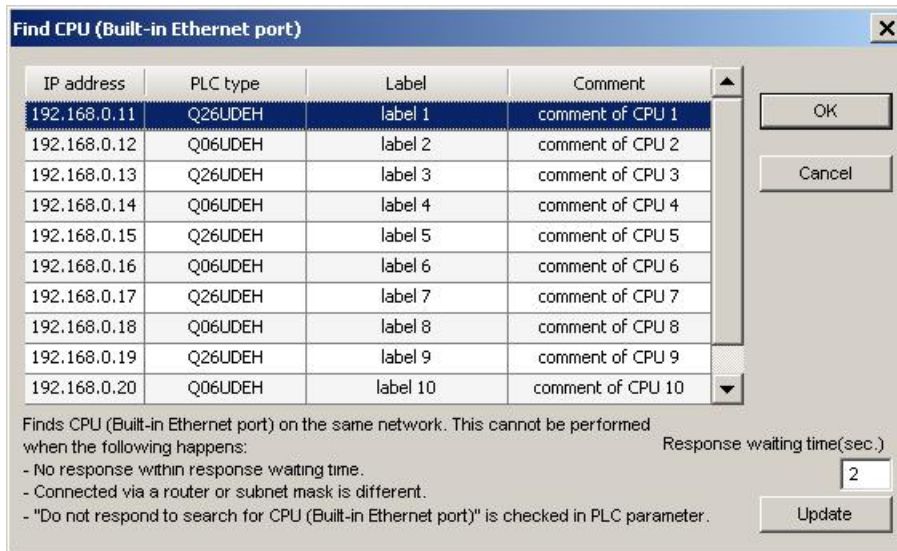
If **UDP** has been selected in the previous [page](#)<sup>[247]</sup>, the additional option 'Ethernet port direct connection' is displayed.



If the option 'Ethernet port direct connection' has been selected, the Ethernet interfaces of PC and PLC CPU must be connected point-to-point with a single cable. The IP address is not required for this type of connection and the corresponding input field is therefore disabled.

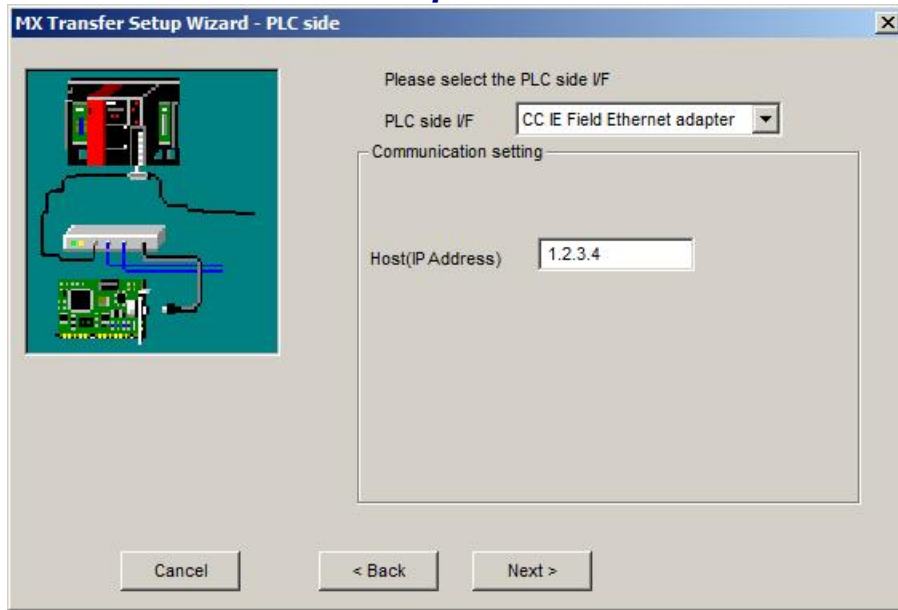


### Find CPU



This dialog lists the QnUDE, QnUDV, L-series and Q Motion series CPUs found in the local Ethernet network. When a CPU in the list is selected and the **OK** button pressed, the IP address of the selected CPU is copied to the 'Host(IP Address)' field. By pressing **Update** the local Ethernet network is scanned again for QnUDE, QnUDV, L-series and Q Motion series CPUs and the list is updated. The time to wait for CPUs to respond during the search is entered in the field **Response waiting time**.

### CC IE Field Ethernet Adapter



#### 11.2.23 PLC G4 Module

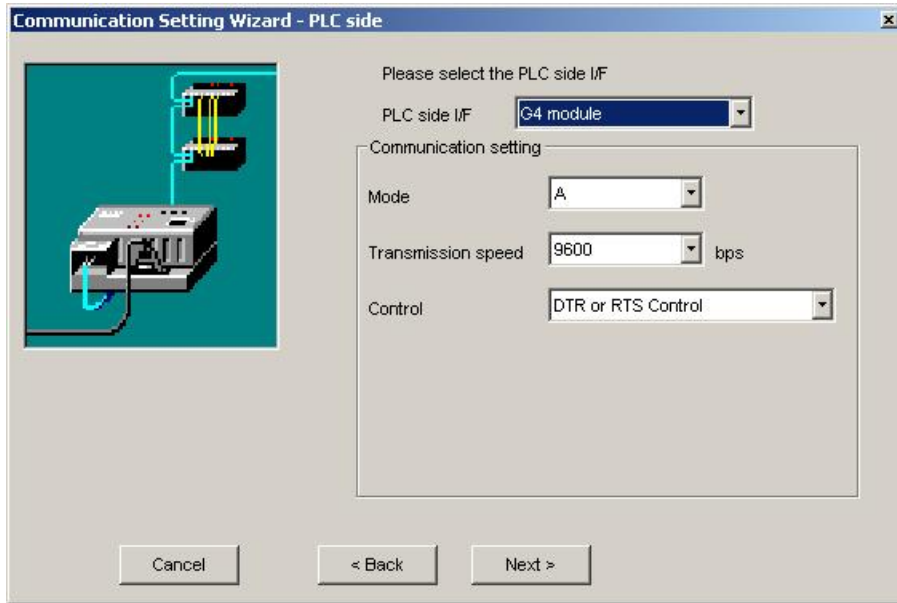
Enter the settings for a G4 module connected to the PC.

#### PLC side G4 module parameters

Parameter	Values	Description
Mode	A, QnA, Q	type of PLC C24 module
Transmission speed	9600 - 115200	baudrate on the serial link <b>available baudrates depend on selected mode</b>
Control	DTR only, RTS only, DTR and RTS, DTR or RTS	handshake signals

	<p><b>Note:</b></p> <p>The CC-Link G4 module used in CC-Link G4 communication (only when the AJ65BT-G4 is used) must have software version 'D' or later. Modules with software version 'C' or earlier will not operate properly.</p>
--	--

### G4 Module connected to PC



For a description of the editable parameters on the page see [PLC side G4 module parameters](#)<sup>[321]</sup>.

Pressing the <Next> button opens a page for specifying an optional additional network layer.

### Q mode

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
	MELSECNET/H	o	x	x	x	o	o	x
	MELSECNET/10	o	o	o	o	x	o	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	o	x	o	x	x	o	x
	C24	x	x	o	x	x	x	x
	CC-Link	x	x	x	x	x	x	x
	CC-Link IE Field	*2	x	x	x	x	x	o
	CC-Link IE Control	o	x	x	x	x	o	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

\*2 : partially supported, depends on CPU type

### QnA mode

Connected Station CPU		Relayed Network	Relayed Station CPU					
QnACPU	QCPU (Q mode)		QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
	MELSECNET/H	x	x	x	x	x	x	x
	MELSECNET/10	x	x	o	x	x	x	x
	MELSECNET(II)	x	x	o	x	x	x	x
	Ethernet	x	x	o	x	x	x	x
	C24	x	x	o	x	x	x	x
	CC-Link	x	x	x	x	x	x	x
	CC-Link IE Field	x	x	x	x	x	x	x
	CC-Link IE Control	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

### A mode

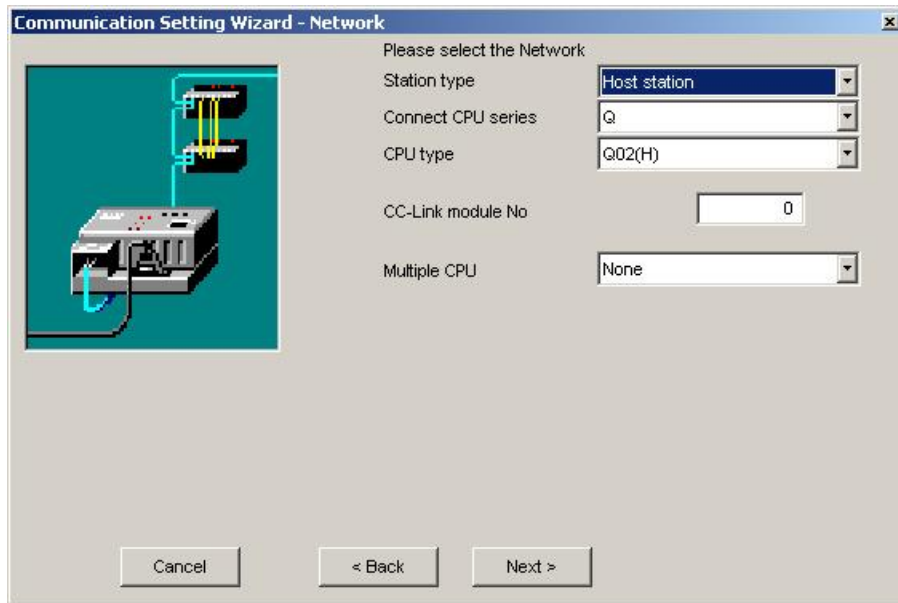
Connected Station CPU			Relayed Network	Relayed Station CPU						
QCPU (A mode)	QnA CPU	ACPU *1		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
o	x	o	MELSECNET/H	x	x	x	x	x	x	x
			MELSECNET/10	x	x	x	x	x	x	x
			MELSECNET(II)	x	x	x	x	x	x	x
			Ethernet	x	x	x	x	x	x	x
			C24	x	x	x	x	x	x	x
			CC-Link	x	x	x	x	x	x	x
			CC-Link IE	x	x	x	x	x	x	x

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

### Host station attached to G4 module

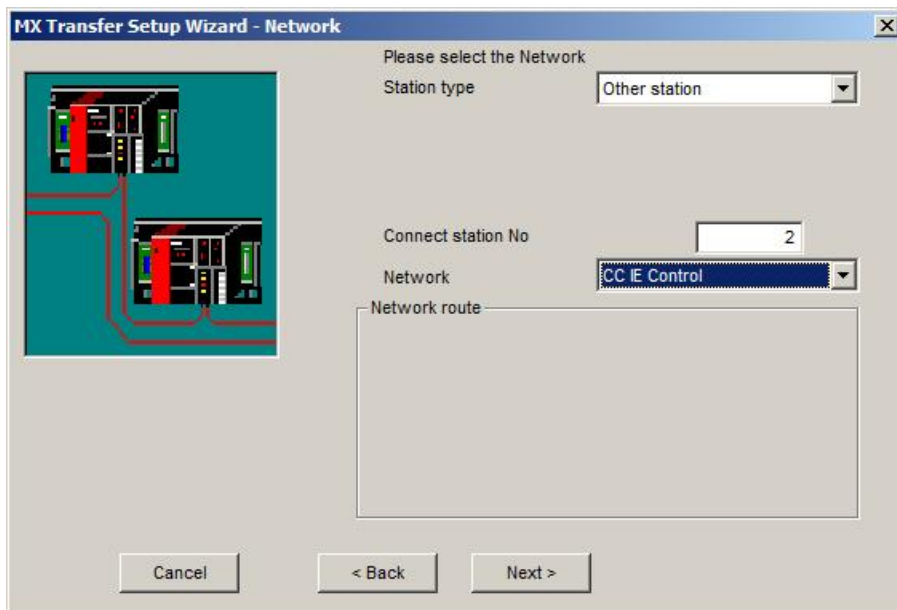


Pressing the <Next> button opens a [page<sup>350</sup>](#) for entering a comment for the connection.

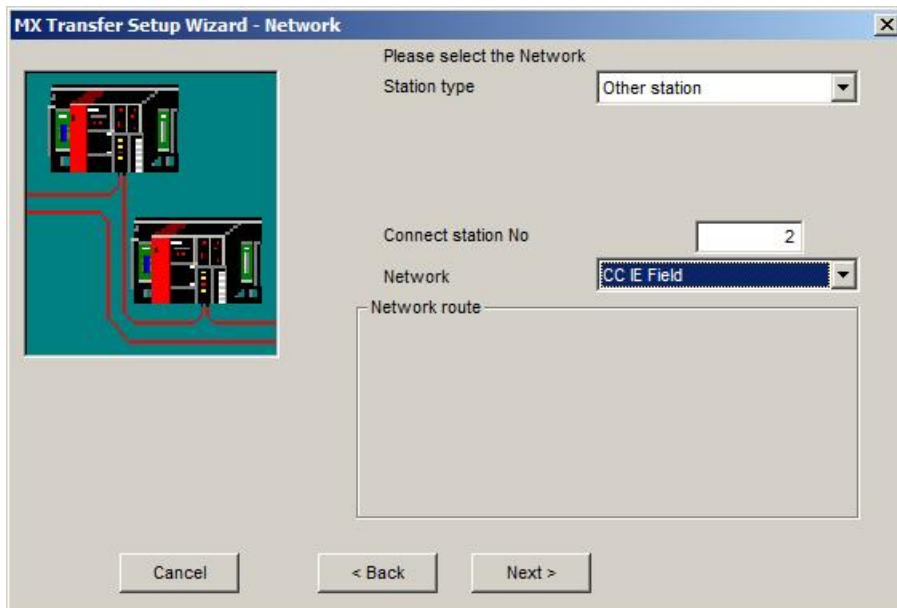


***Other station in a network attached to G4 module***

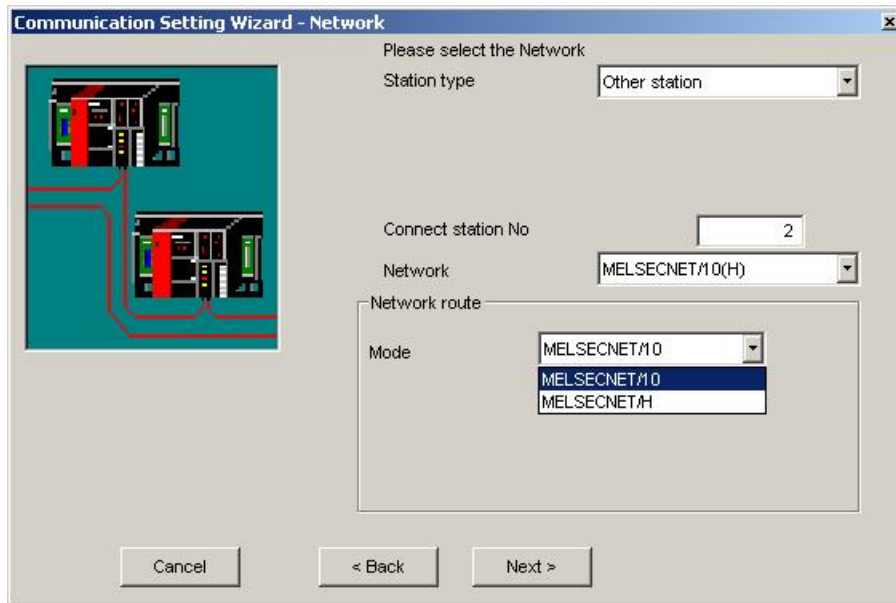
***In a CC IE Control network***



***In a CC IE Field network***

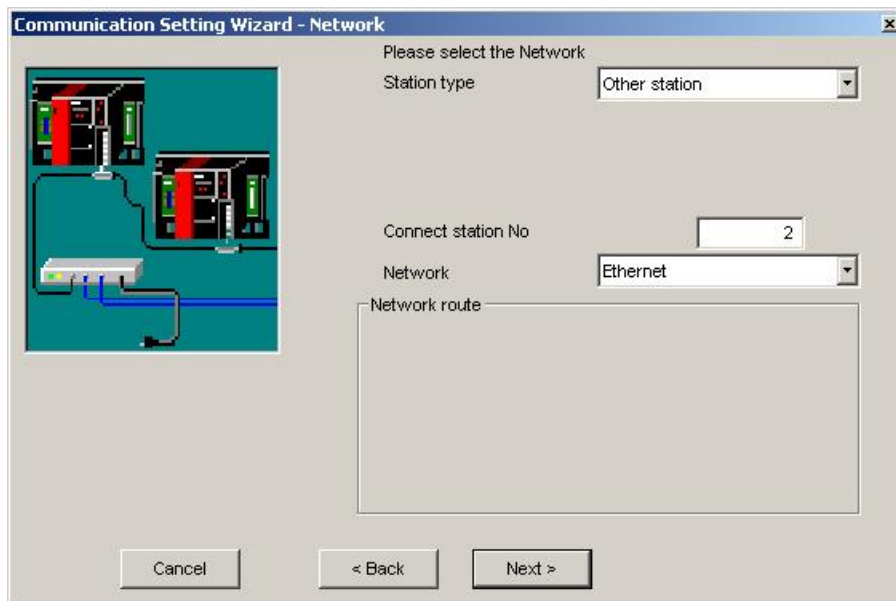


## In a MELSECNET/10 network



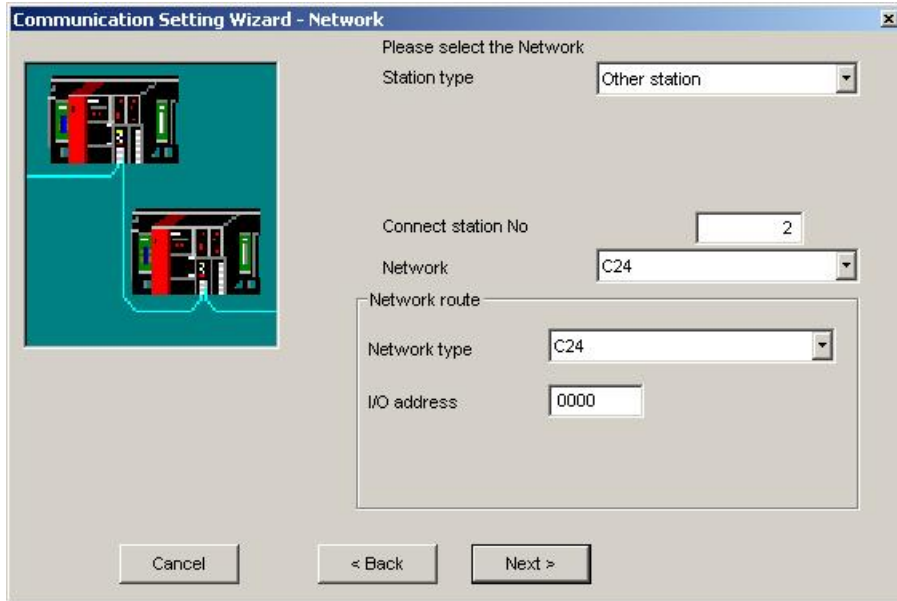
Pressing the **<Next>** button opens a [page<sup>\[345\]</sup>](#) for specifying the station type and address.

## In an Ethernet network



Pressing the **<Next>** button opens a [page<sup>\[345\]</sup>](#) for specifying the station type and address.

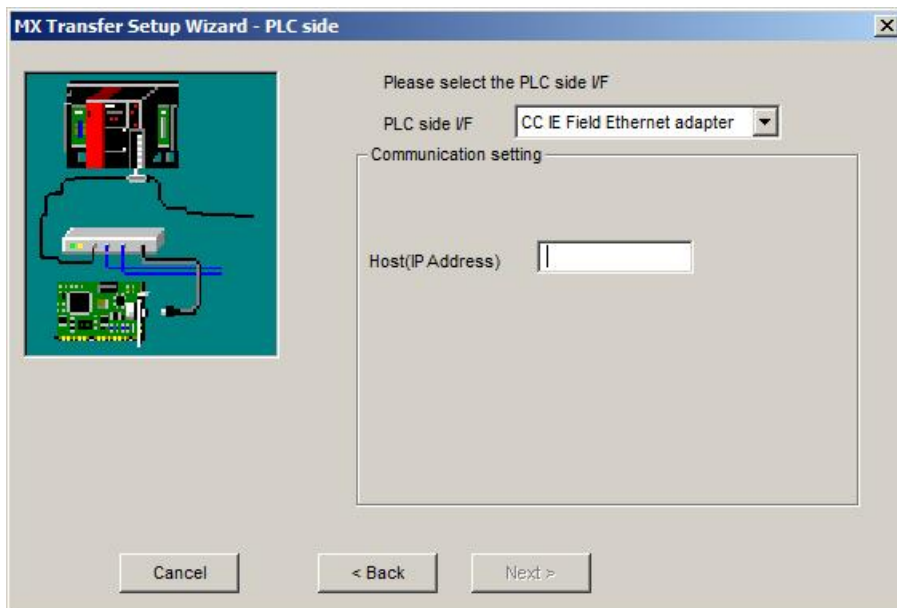
### In a C24 network



Pressing the <Next> button opens a [page<sup>\[343\]</sup>](#) for specifying the station type and address.

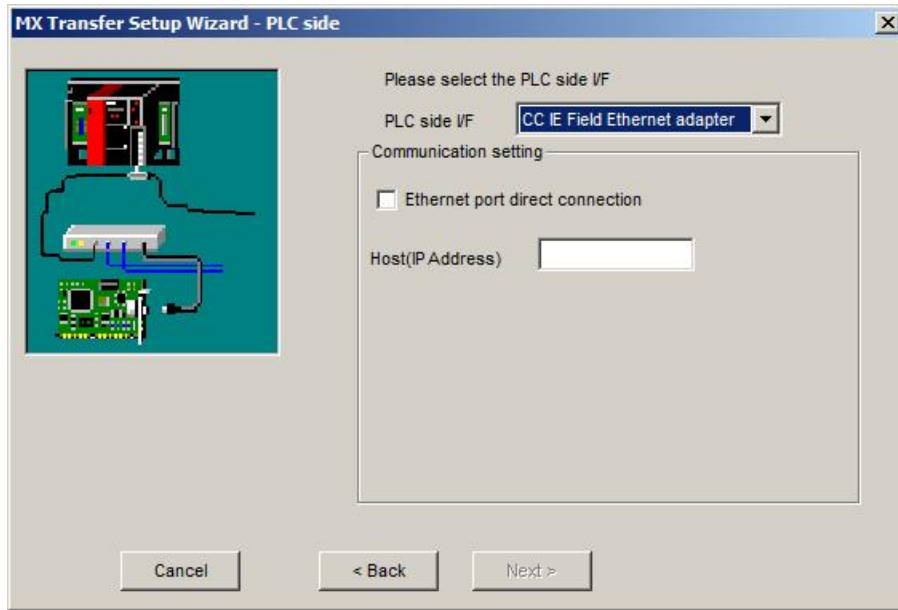
### 11.2.24 PLC CC-Link IE Field Ethernet Adapter

If TCP has been selected in the previous [page<sup>\[247\]</sup>](#), the following page is displayed. Enter the IP address or host name of the CC IE Field adapter.

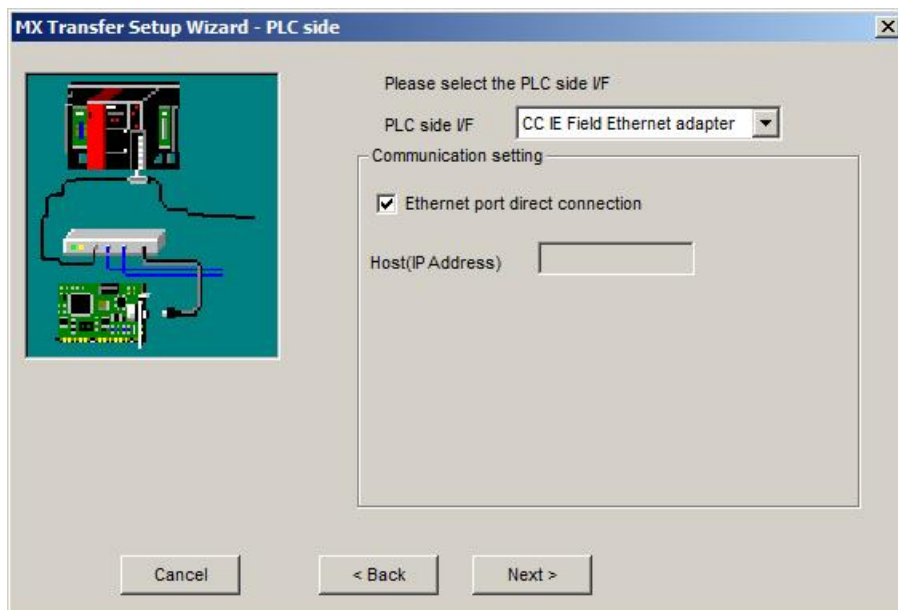


Parameter	Values	Description
Host (IP Address)	IP address or host name	IP address of the CC-Link IE Field Ethernet interface

If **UDP** has been selected in the previous [page<sup>\[247\]</sup>](#), the additional option 'Ethernet port direct connection' is displayed.



If the option 'Ethernet port direct connection' has been selected, the Ethernet interfaces of PC and PLC CPU must be connected point-to-point with a single cable. The IP address is not required for this type of connection and the corresponding input field is therefore disabled.



Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
*2	MELSECNET/H	o	x	x	x	x	x	x
	MELSECNET/10	o	x	x	x	x	x	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	o	x	x	x	x	x	x
	C24	o	x	x	x	x	x	o
	CC-Link	o	x	x	x	x	x	o
	CC-Link IE Field	*2	x	x	x	x	x	o
CC-Link IE Control	IE	o	x	x	x	x	x	

o: Accessible

x: Inaccessible

\*1 : Including motion controller CPU

\*2 : For the Redundant CPU, access is disabled

Connected Station CPU	Relayed Network	Relayed Station CPU						
		QCPU (Q mode)	QCPU (A mode)	QnA CPU	ACPU *1	FXCPU	Q Motion CPU	LCPU
o	MELSECNET/H	x	x	x	x	x	x	x
	MELSECNET/10	x	x	x	x	x	x	x
	MELSECNET(II)	x	x	x	x	x	x	x
	Ethernet	x	x	x	x	x	x	x
	C24	o	x	x	x	x	x	o
	CC-Link	o	x	x	x	x	x	o
	CC-Link IE Field	*2	x	x	x	x	x	o
	CC-Link IE Control	IE	x	x	x	x	x	x

o: Accessible

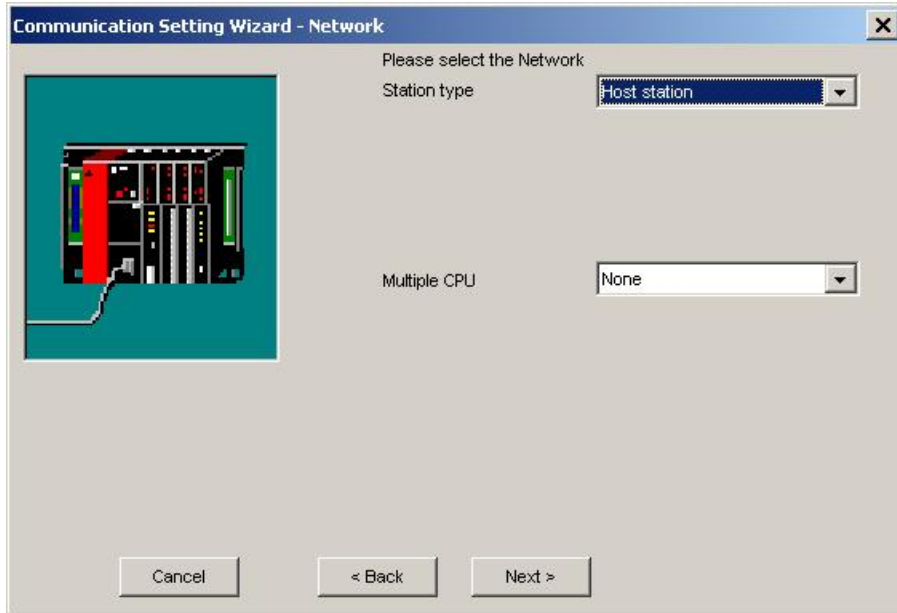
x: Inaccessible

\*1 : Including motion controller CPU

\*2 : For the Redundant CPU, access is disabled

11.2.25 Network

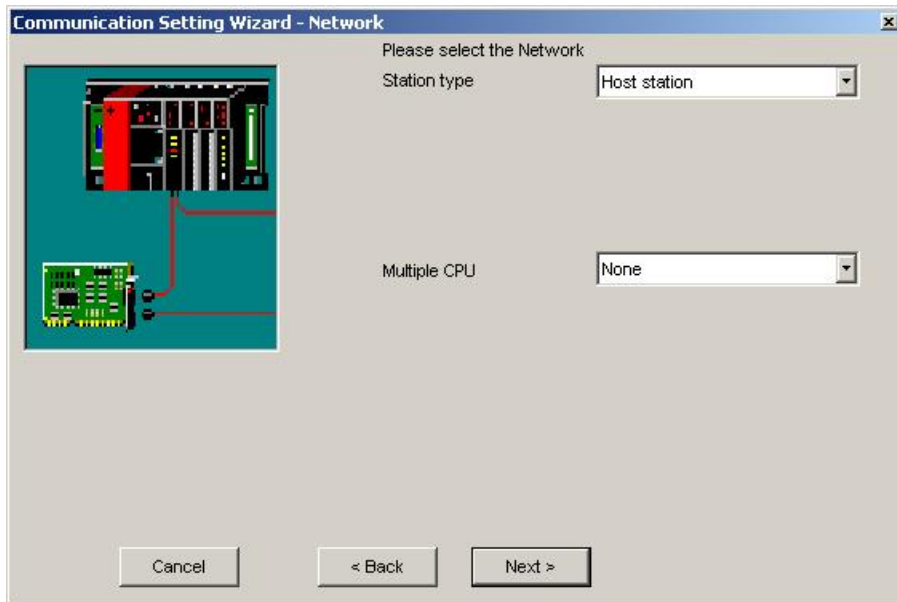
Host station for CPU connections



If the target PLC contains multiple CPUs, the number of the target CPU must be selected in the 'Multiple CPU' parameter. For details see '[Appendix Multi-CPU](#)'<sup>[351]</sup>.

Pressing the <Next> button opens a [page](#)<sup>[350]</sup> for entering a comment for the connection.

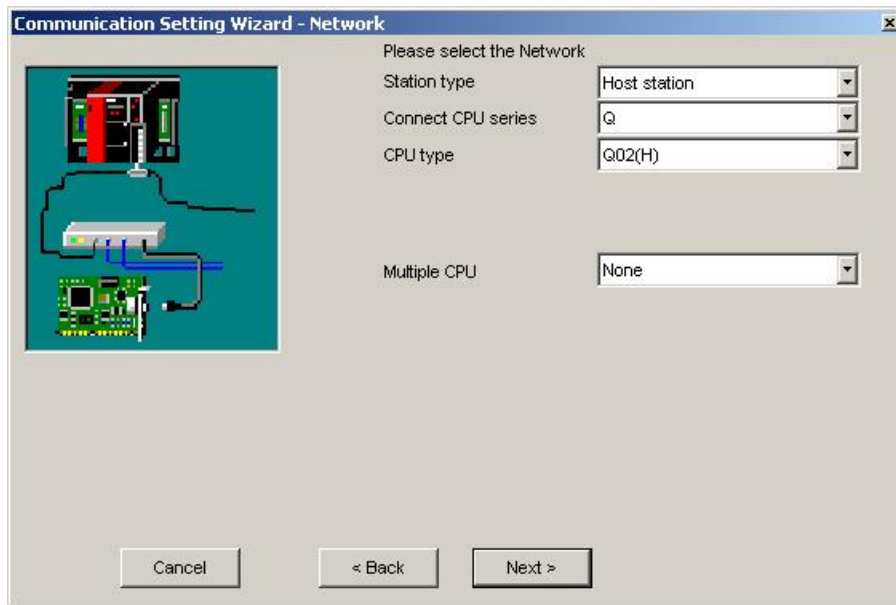
Host station for MELSECNET and CC-Link connections



If the target PLC contains multiple CPUs, the number of the target CPU must be selected in the 'Multiple CPU' parameter. For details see '[Appendix Multi-CPU](#)'<sup>[351]</sup>.

Pressing the <Next> button opens a [page](#)<sup>[350]</sup> for entering a comment for the connection.

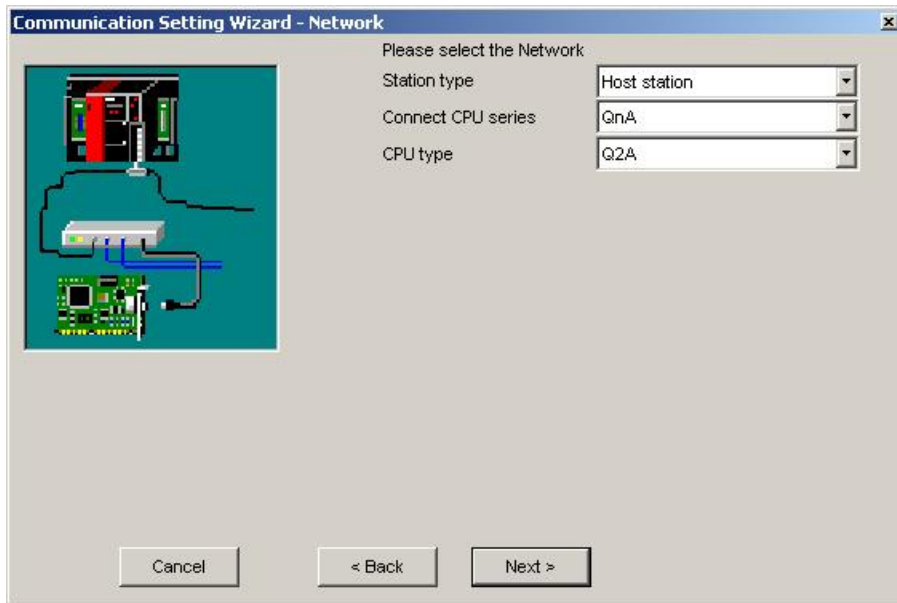
## Host station for C24 connections



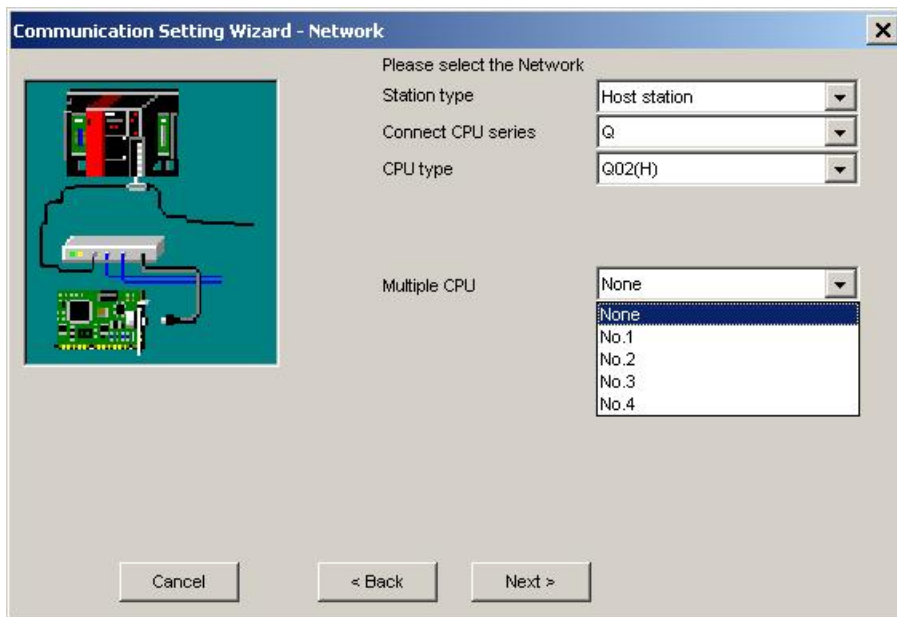
Select the type of the target CPU. If the target PLC contains multiple CPUs, the number of the target CPU must be selected in the 'Multiple CPU' parameter. For details see '[Appendix Multi-CPU](#)'<sup>[351]</sup>.

Pressing the <Next> button opens a [page](#)<sup>[350]</sup> for entering a comment for the connection.

### Host station for Ethernet connections



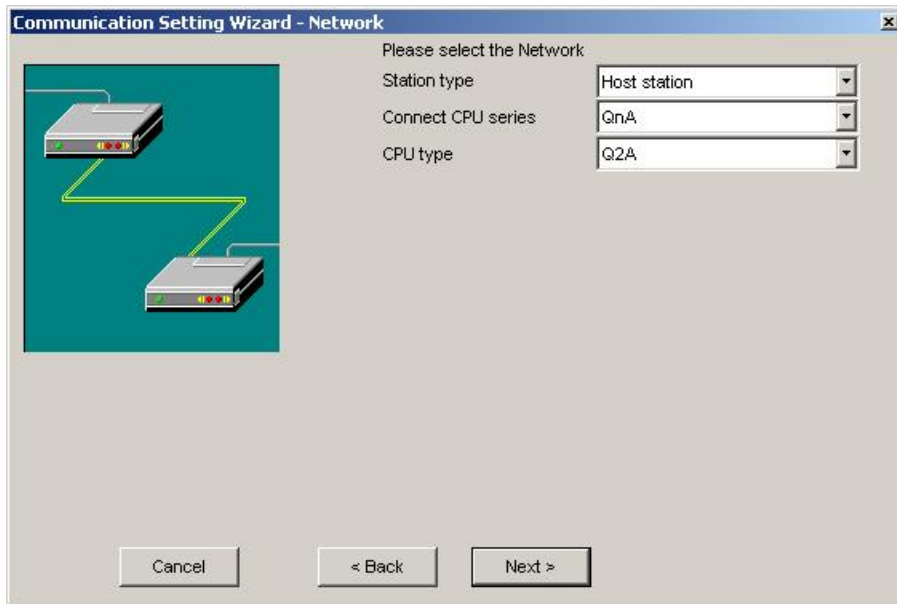
For Q-series PLCs the index of the CPU in a multi-CPU system can optionally be selected.



Select the type of the target CPU and press the **<Next>** button to open a [page](#)<sup>[350]</sup> for entering a comment for the connection.

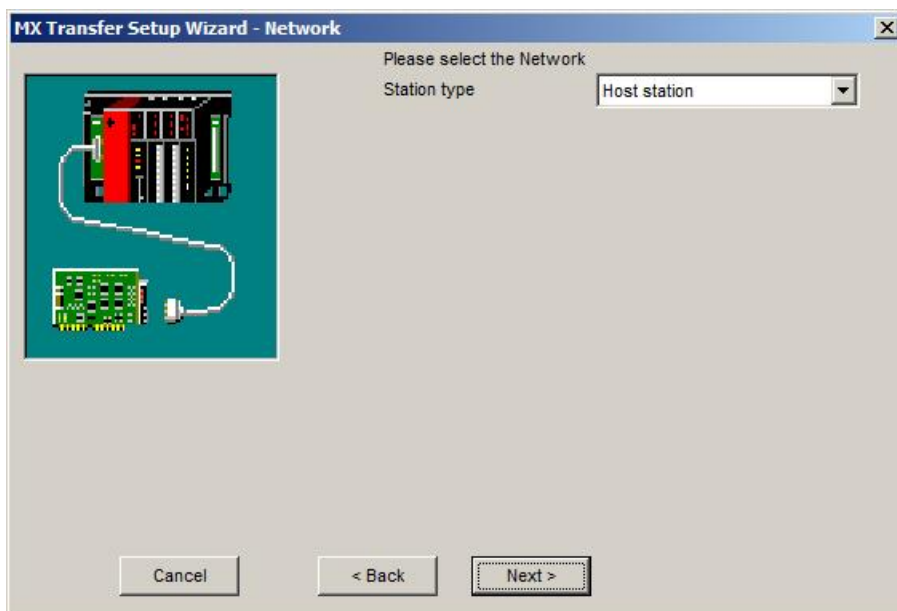


### Host station for modem connections

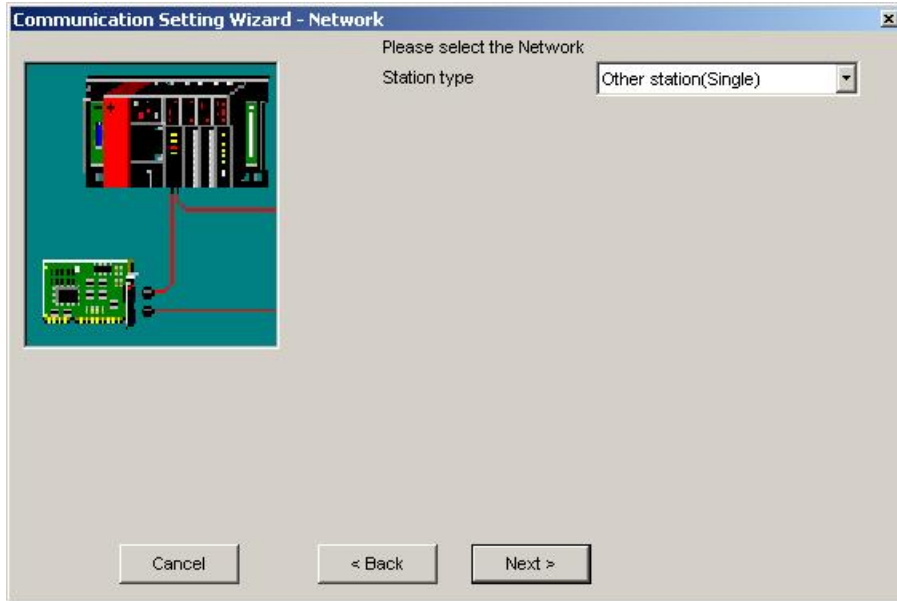


Pressing the <Next> button opens a [page](#)<sup>350</sup> for entering a comment for the connection.

### Host Station for CPU Board Connections



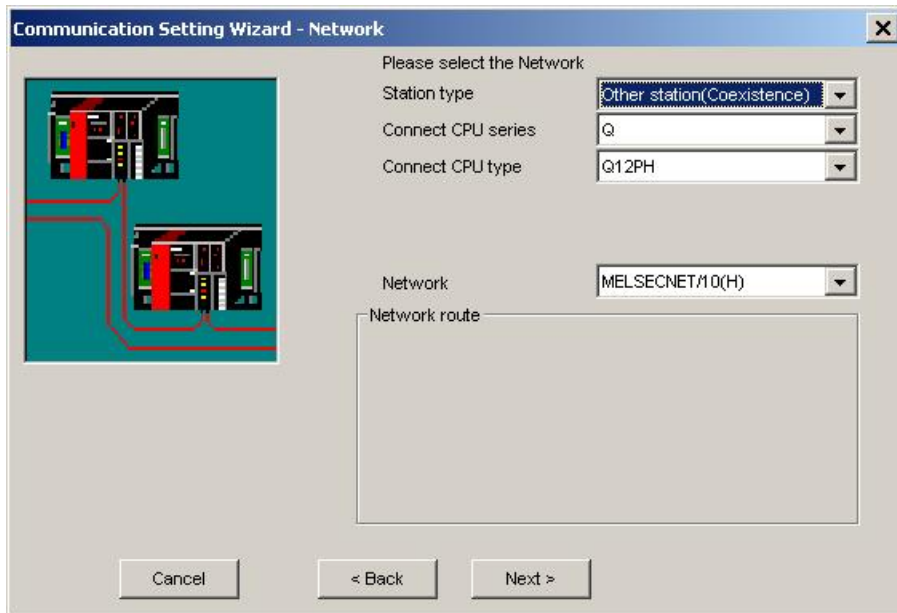
### Station in single network



Pressing the **<Next>** button opens a [page<sup>\[345\]</sup>](#) for entering the MELSECNET address of the target PLC.

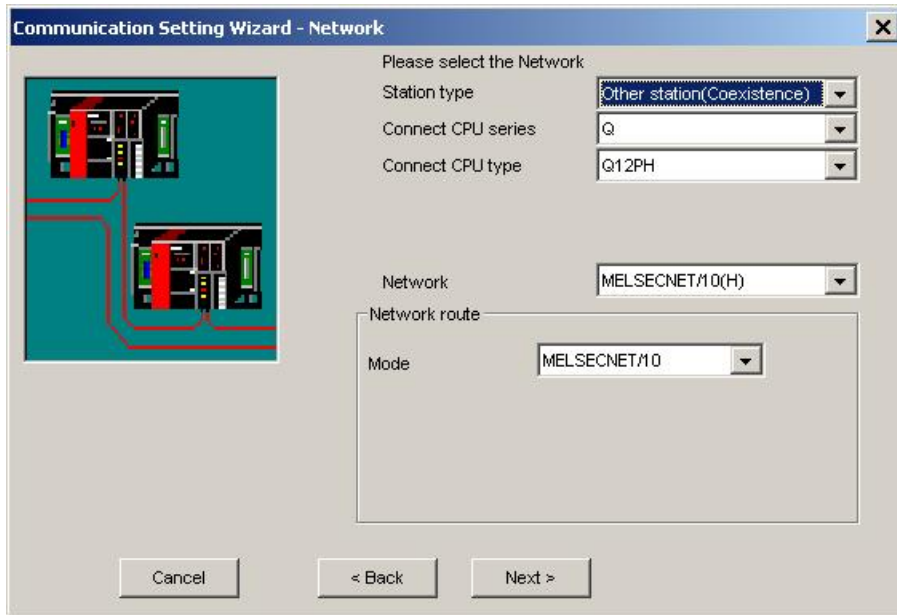
### Station in coexisting network

#### For MELSECNET/10 interfaces



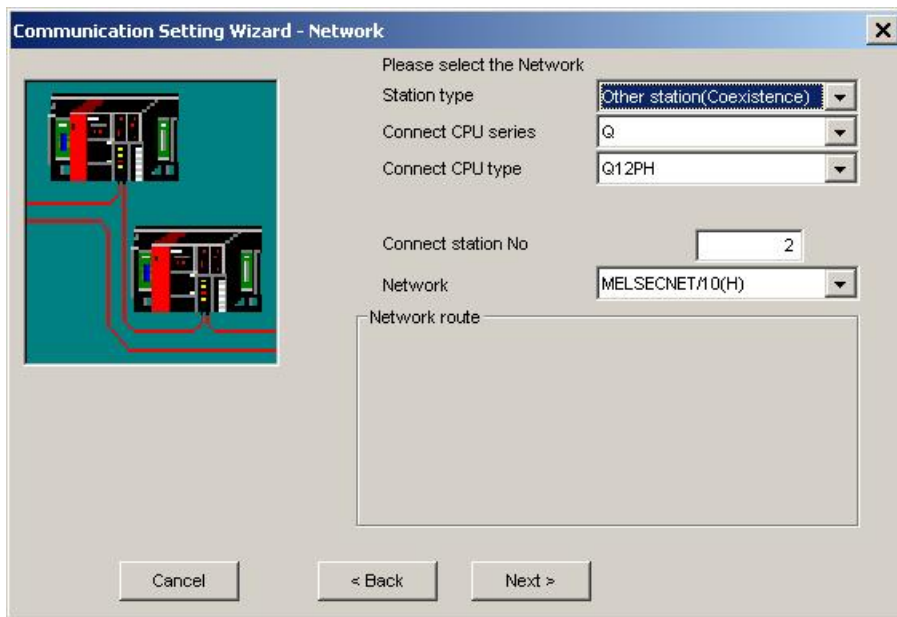
Select the type of the coexisting network and press the **<Next>** button to open a [page<sup>\[345\]</sup>](#) for entering parameters of the target PLC in the coexisting network.

**For MELSECNET/H interfaces**



Select the type of the coexisting network and press the **<Next>** button to open a [page<sup>\[345\]</sup>](#) for entering parameters of the target PLC in the coexisting network.

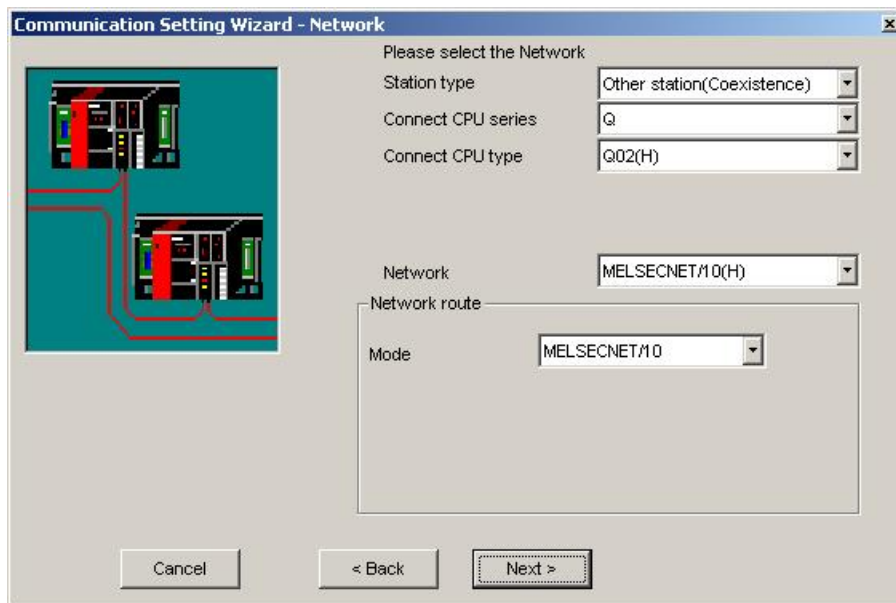
**For CC-Link interfaces**



Parameter	Values	Description
Connect station no.	0 - 64	station number of the CC-link module in the target station

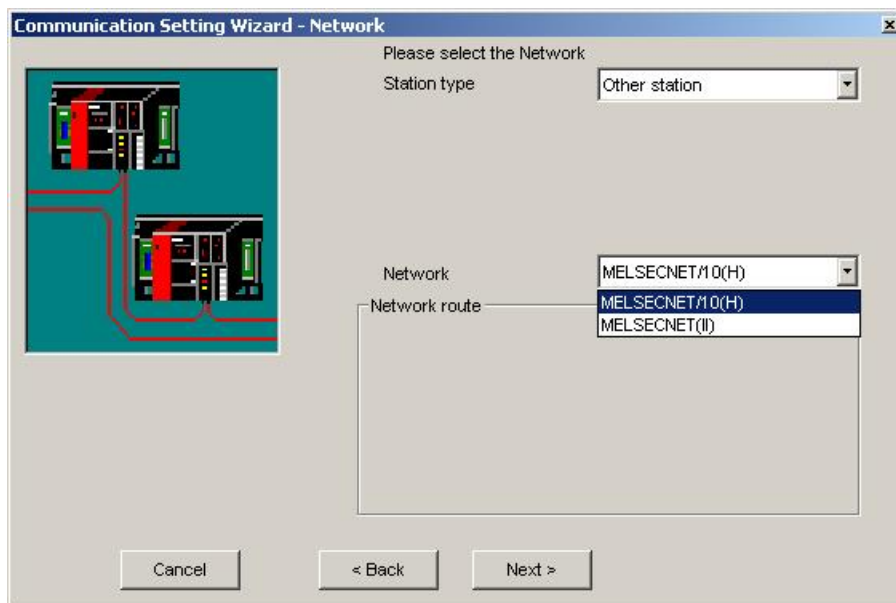
Select the type of the coexisting network and press the **<Next>** button to open a [page<sup>\[345\]</sup>](#) for entering parameters of the target PLC in the coexisting network.

### For CC-Link IE interfaces

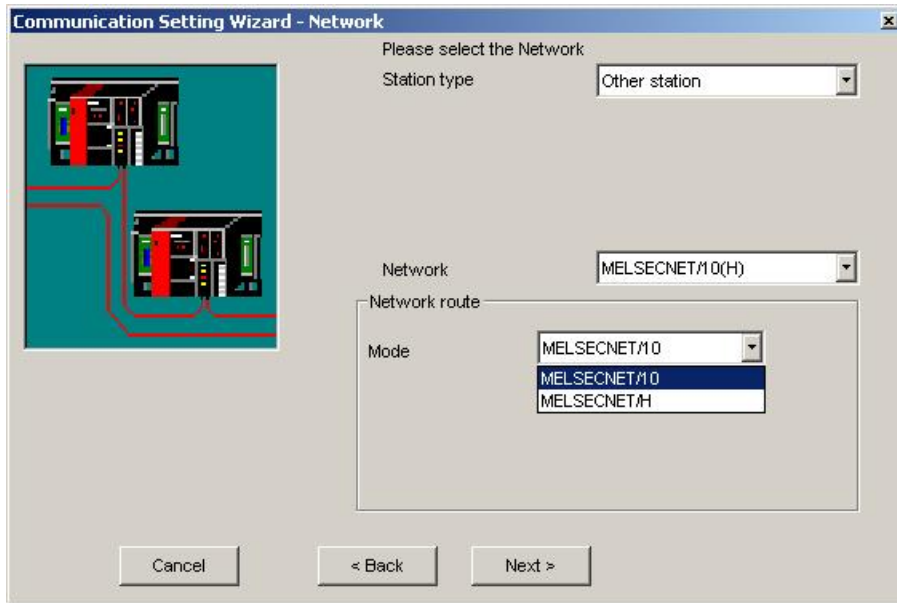


Select the type of the coexisting network and press the **<Next>** button to open a [page<sup>\[345\]</sup>](#) for entering parameters of the target PLC in the coexisting network.

### Station in MELSECNET/10 network

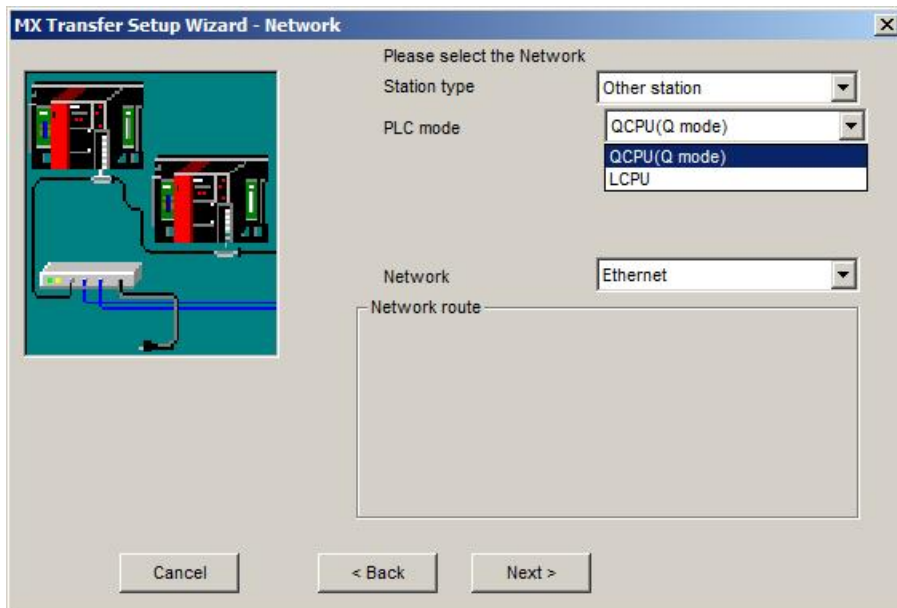


Select the type of the network, to which the target PLC is connected. Additionally the mode may also be selectable.

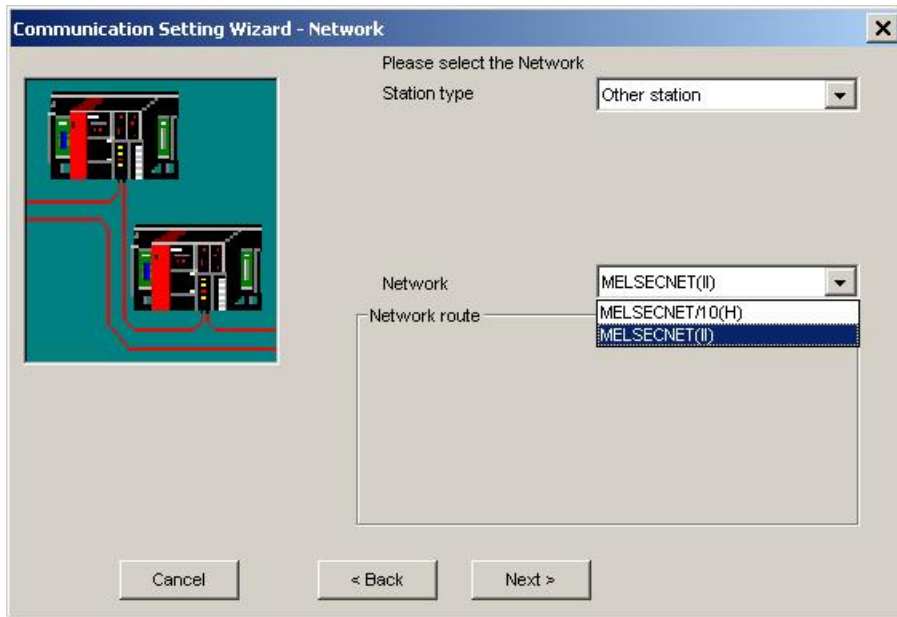


Pressing the <Next> button opens a [page](#)<sup>[342]</sup> for entering the MELSECNET address of the target PLC.


### Station in Ethernet network



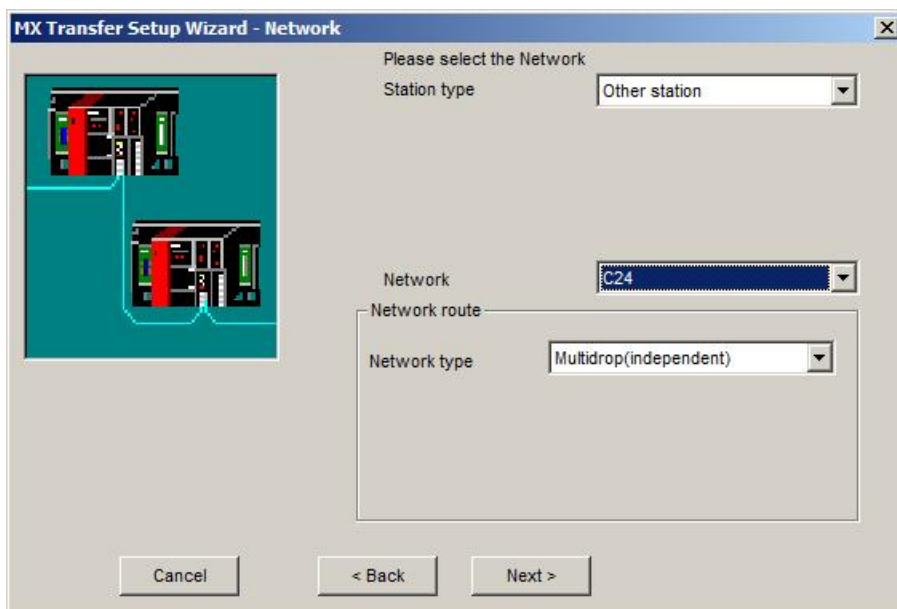
### Station in MELSECNET(II) network



Pressing the **<Next>** button opens a [page<sup>\[342\]</sup>](#) for entering the MELSECNET address of the target PLC.

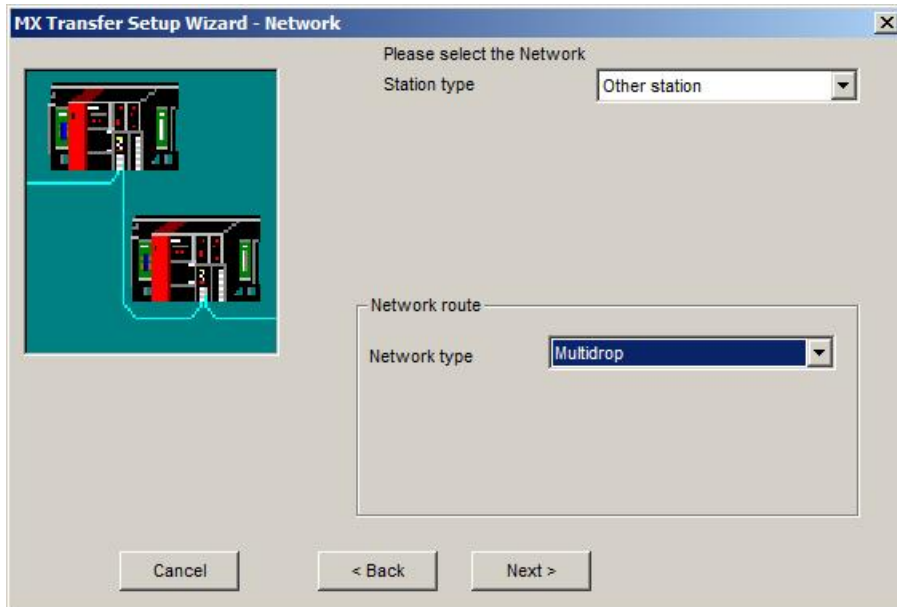
	<p><b>Note:</b></p> <p>When access is made to the QnACPU, AnUCPU, QCPU (A mode) or motion controller CPU via the MELSECNET(II), the device range is equivalent to that of the AnACPU.</p>
---	---

### Station in C24 multi drop network



Pressing the **<Next>** button opens a [page<sup>\[343\]</sup>](#) for entering the station address of the target PLC in multi drop network.

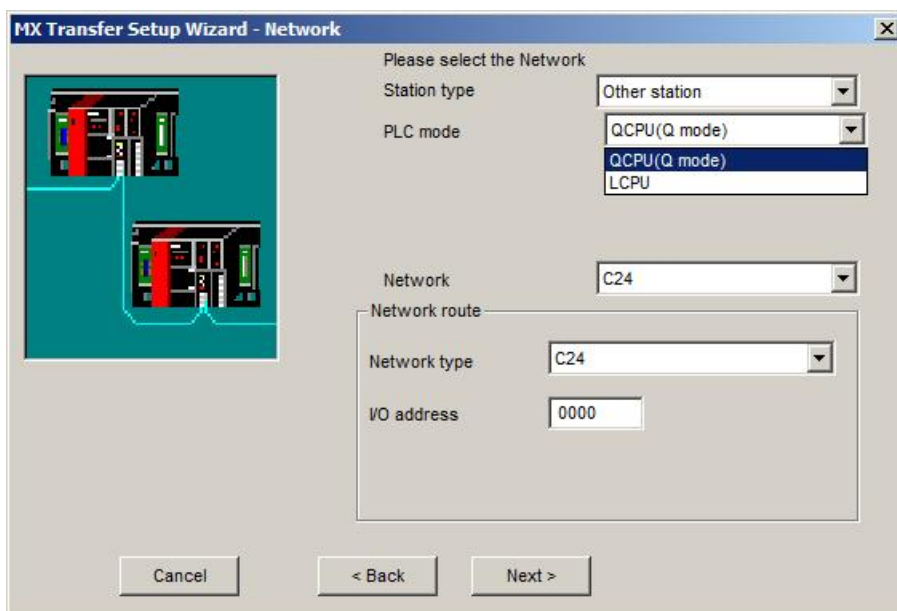
### Station in FX multi drop network



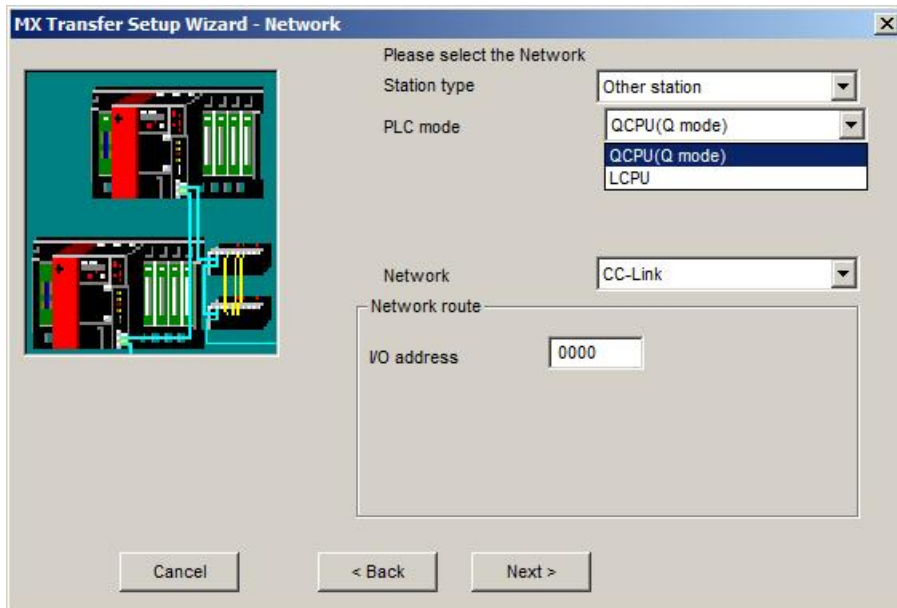
Pressing the **<Next>** button opens a [page](#)<sup>[343]</sup> for entering the station address of the target PLC in multi drop network.

### Network with C24 station

Parameter	Values	Description
Network type	C24	fixed
I/O address	I/O number divided by 0x10	module I/O address of connected station

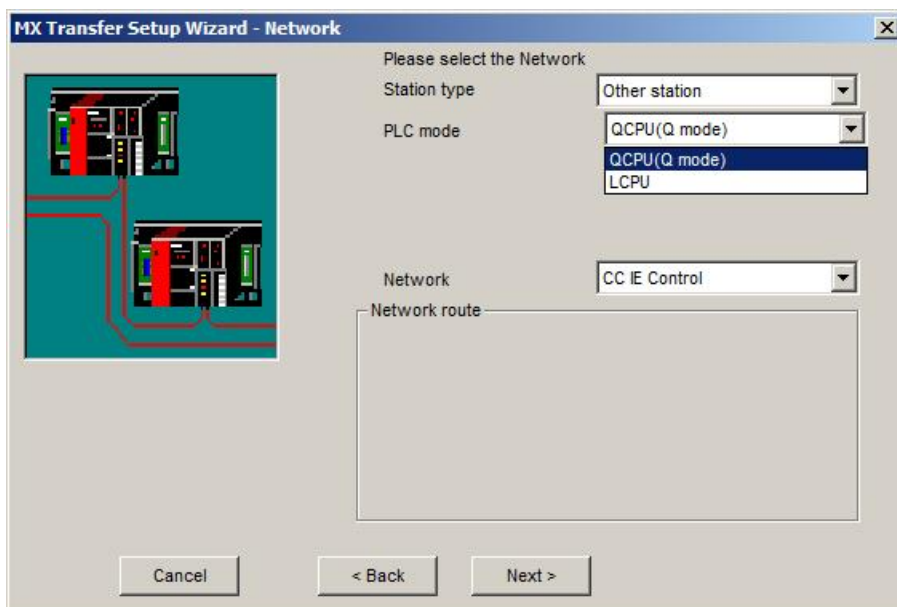


### Network with CC-Link station



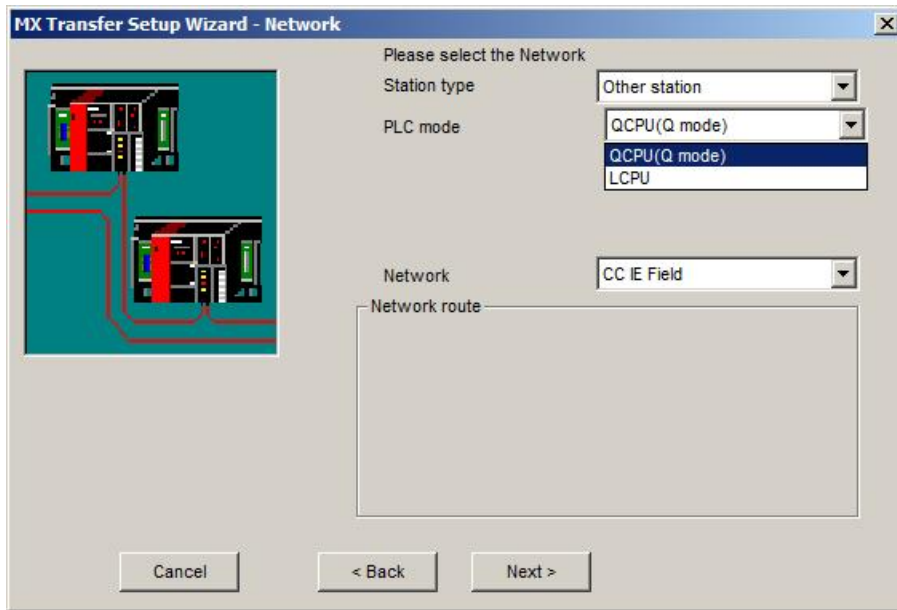
Parameter	Values	Description
I/O address	I/O number divided by 0x10	module I/O address of connected station

### Network with CC-Link IE Control





### Network with CC-Link IE Field

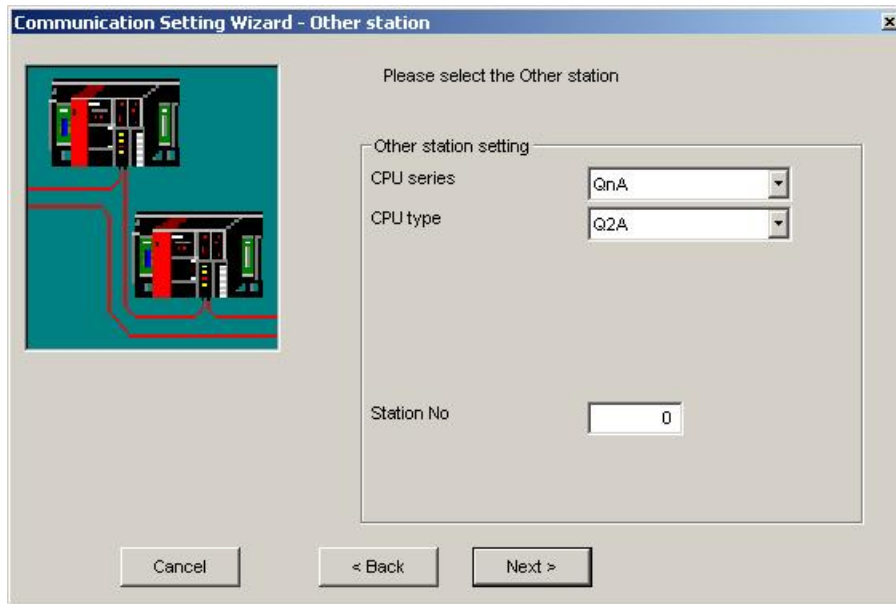


#### 11.2.26 Other Station

This page is used to provide type and network address information of the target PLC in a multi-layered network.

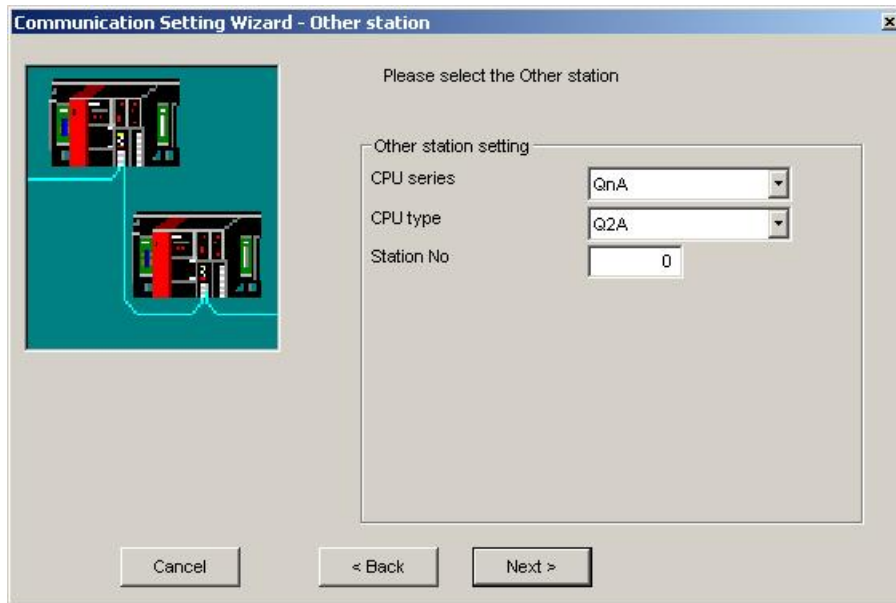
Parameter	Values	Description
CPU series	Q, QnA, A, FX, L and Q Motion depending on previous choices	series of the CPU type
CPU type	see list of supported CPU types	type of PLC CPU module
Network No	1 - 239	network number of the target interface
Station No.	1 - 64	station number of the target interface, limits may be different for certain protocols
Station No. for CC-Link network	0 - 63	station number of the target interface, 0 is not available for FX CPUs

## Station in same network for MELSECNET(II)

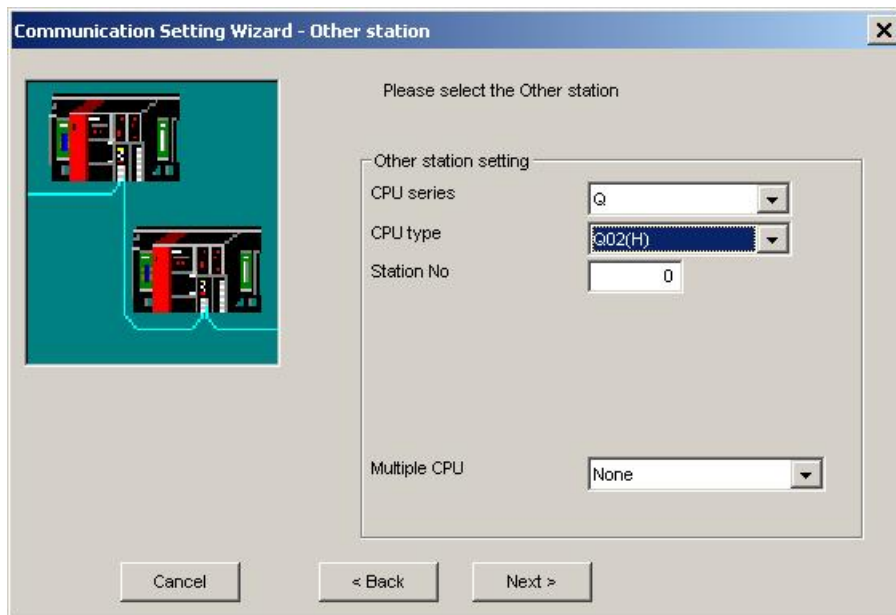


Pressing the <Next> button opens a [page<sup>350</sup>](#) for entering a comment for the connection.

## Station in same network for C24 or FX

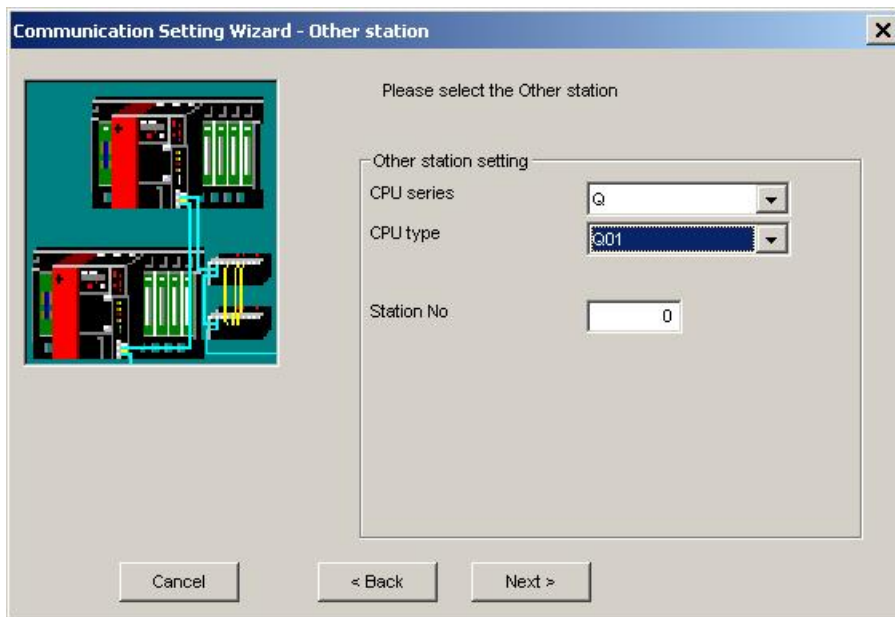


For PLCs, which support multiple CPUs, the index of the CPU in a multi-CPU system can optionally be selected.

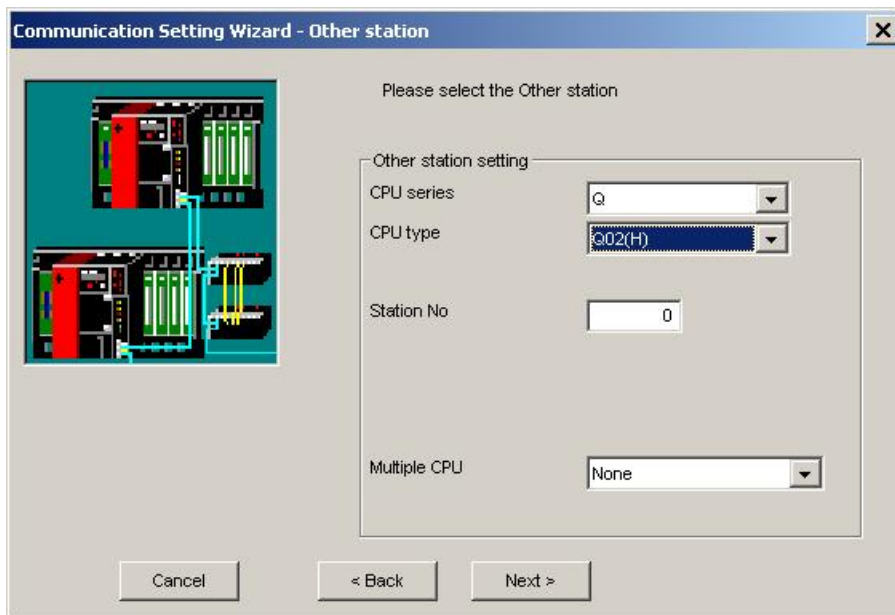


Pressing the **<Next>** button opens a [page<sup>350</sup>](#) for entering a comment for the connection.

### Station in same network for CC-Link

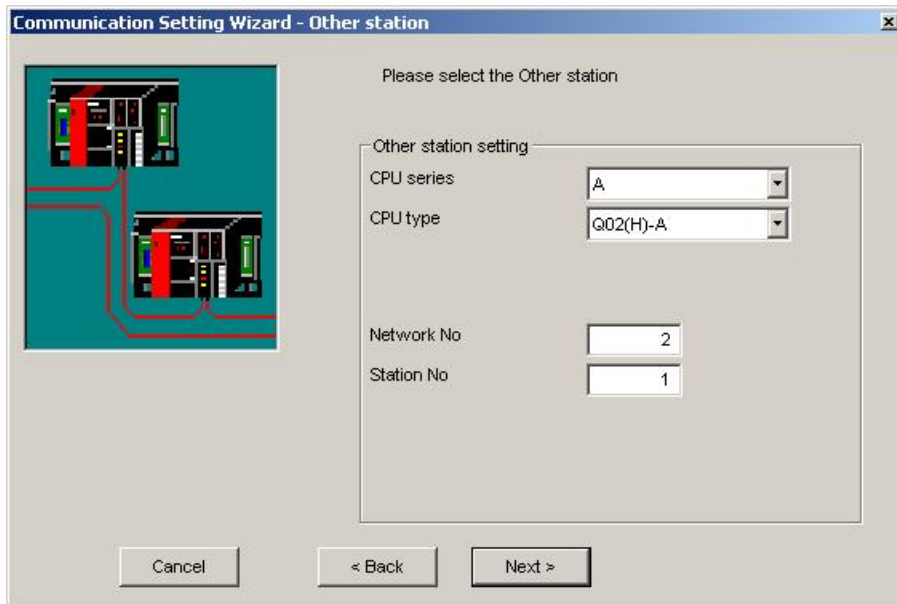


For Q-series PLCs the index of the CPU in a multi-CPU system can optionally be selected.

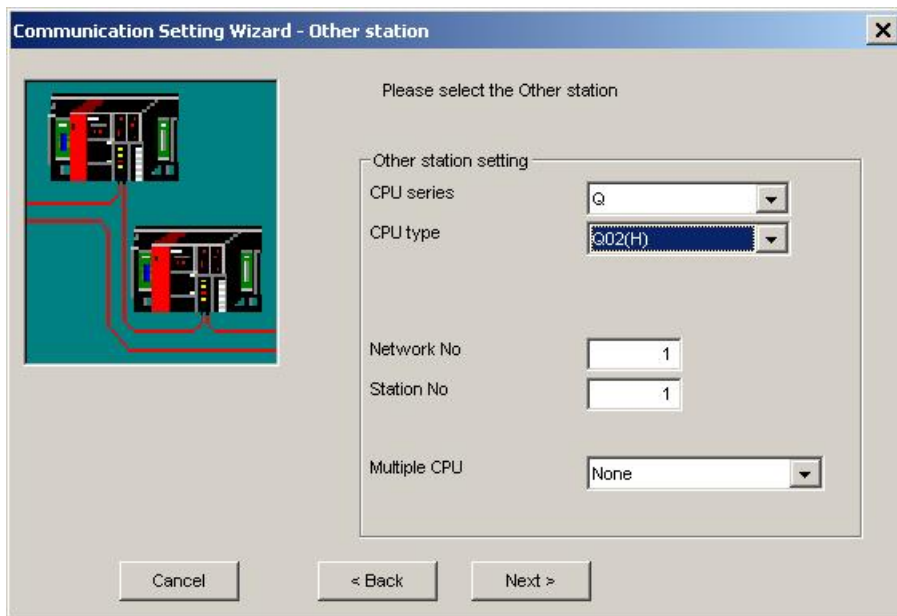


Pressing the **<Next>** button opens a [page](#)<sup>350</sup> for entering a comment for the connection.

### Station in same or different network



For Q-series PLCs the index of the CPU in a multi-CPU system can optionally be selected.



If the target PLC is in a different network, the network number of that network must be provided additionally to the station number of the target PLC.

Pressing the **<Next>** button opens a [page](#)<sup>[350]</sup> for entering a comment for the connection.

### 11.2.27 GOT Side Settings

#### ***PC Serial Interfaces (USB and RS232)***

For USB and RS232 serial connections between PC and GOT the following interfaces of the GOT can be used to connect to the PLC

- o [GOT Serial Interface](#)<sup>[346]</sup>
- o [GOT Bus Interface](#)<sup>[347]</sup>
- o [GOT Ethernet Interface](#)<sup>[347]</sup>

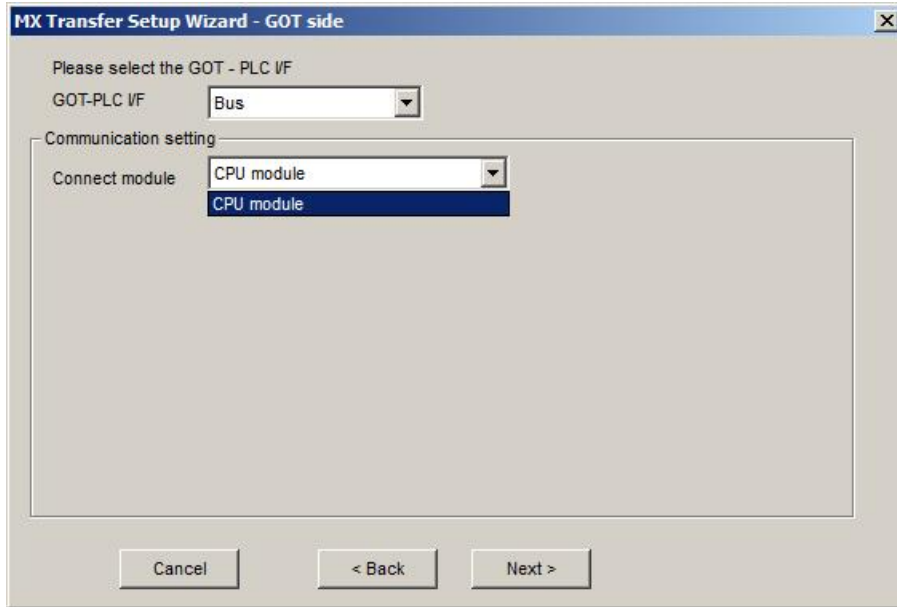
The interfaces, which are actually available on the GOT, depend on the specific GOT hardware used.

#### **GOT Serial Interface**

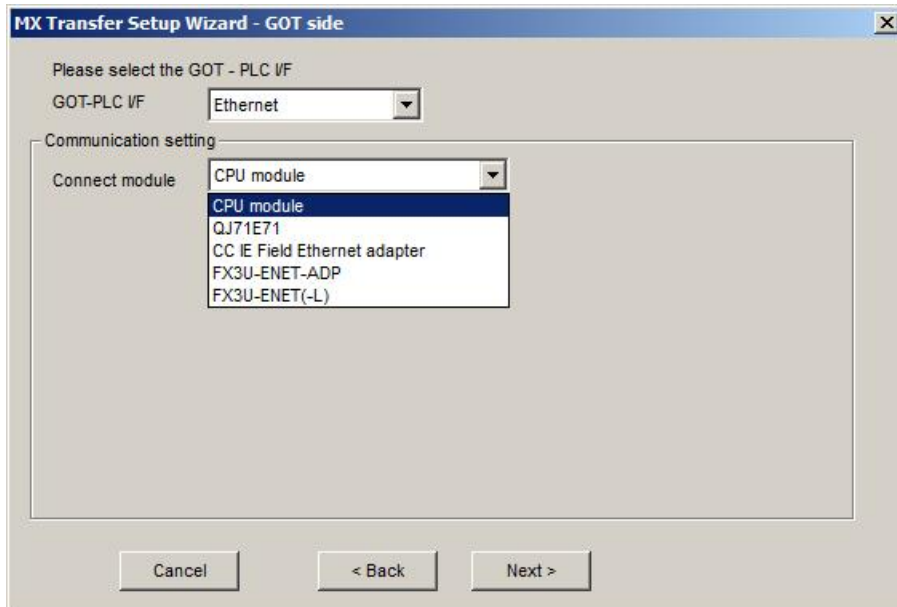


Parameter	Values	Description
Connect module	CPU module QJ71C24 LJ71C24	type of connected unit on the PLC side

### GOT Bus Interface



### GOT Ethernet Interface



Parameter	Values	Description
Connect module	CPU module QJ71E71 CC IE Field Ethernet adapter FX3U-ENET-ADP FX3U-ENET(-L)	type of connected unit on the PLC side

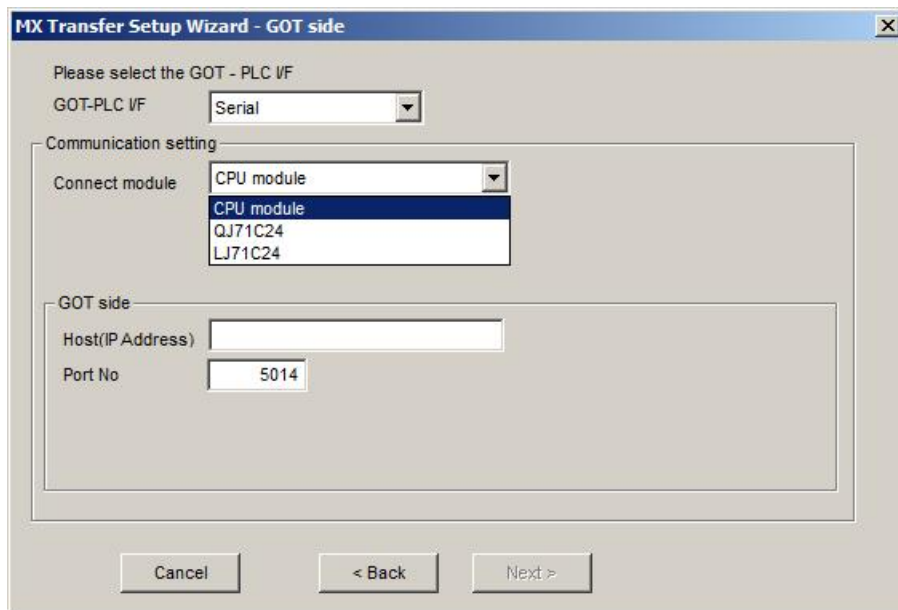
### PC Ethernet Interface

For Ethernet connections between PC and GOT the following interfaces of the GOT can be used to connect to the PLC

- o [GOT Serial Interface](#)<sup>[348]</sup>
- o [GOT Bus Interface](#)<sup>[349]</sup>

The interfaces, which are actually available on the GOT, depend on the specified GOT hardware used.

### GOT Serial Interface



Parameter	Values	Description
Connect module	CPU module QJ71C24 LJ71C24	type of connected unit on the PLC side
Host (IP Address)	IP address or host name	IP address of the GOT ethernet interface
Port No	1024 - 65534	port number on the GOT for the PC connection



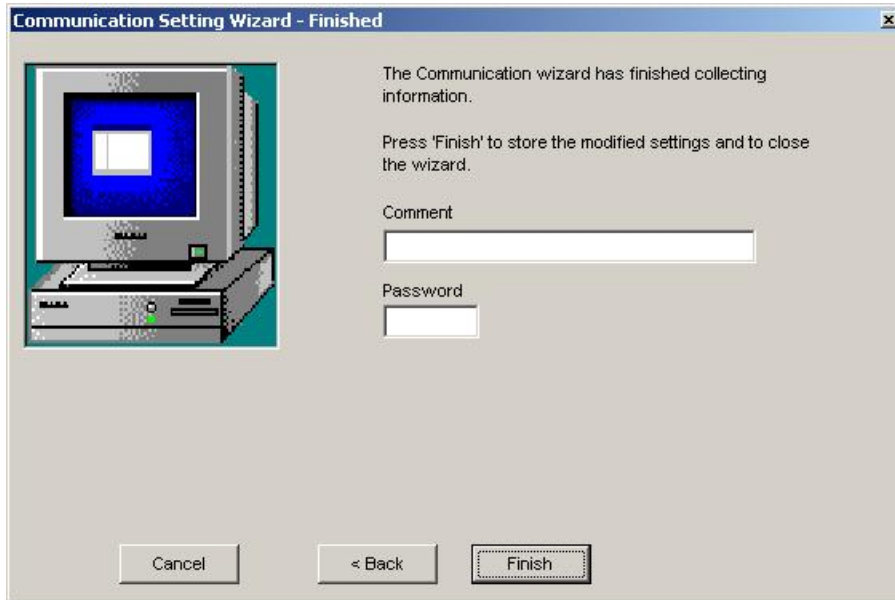
### GOT Bus Interface



Parameter	Values	Description
Connect module	CPU module	type of connected unit on the PLC side
Host (IP Address)	IP address or host name	IP address of the GOT ethernet interface
Port No	1024 - 65534	port number on the GOT for the PC connection

### 11.2.28 Enter Comment

Enter a comment to describe the configured PLC connection and test the PLC connection with the current settings.

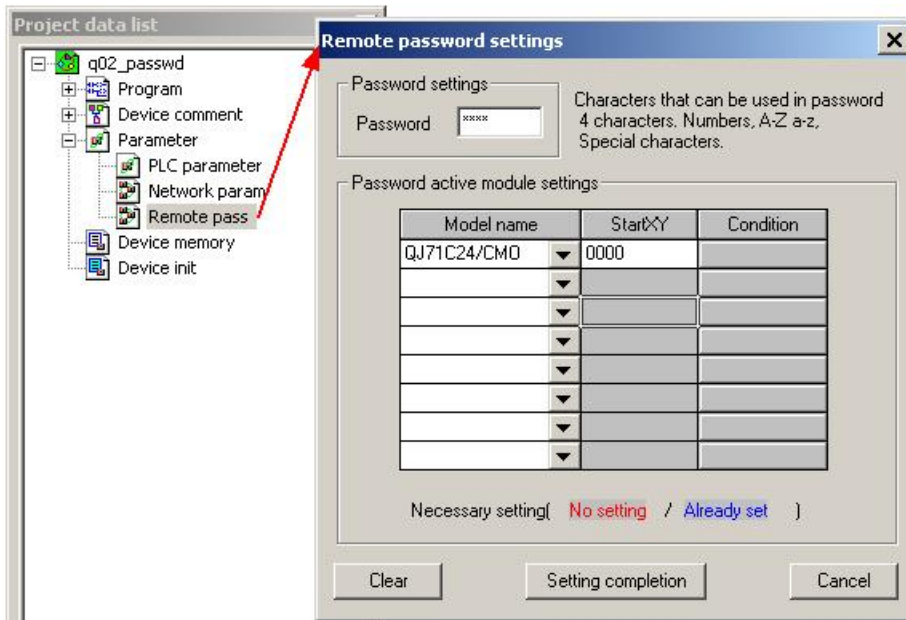


For modem connections to QJ71C24 and QJ71CMO modules a password may be required in order to gain access to the CPU. Enter the password, which has been set in the CPU parameters. For other connection types the password input field is disabled.

The password must not exceed four characters and can consist of alphanumerical characters and standard signs. Blanks are not allowed.

### Remote access password

The password for remote access via QJ71E71 and QJ71C24 is assigned in the CPU parameters of a GX Developer project and downloaded to the target PLC. Passwords set in any relaying units have no effect.



Enter the same password in the 'Password' input field for the connection to gain access to the target PLC.

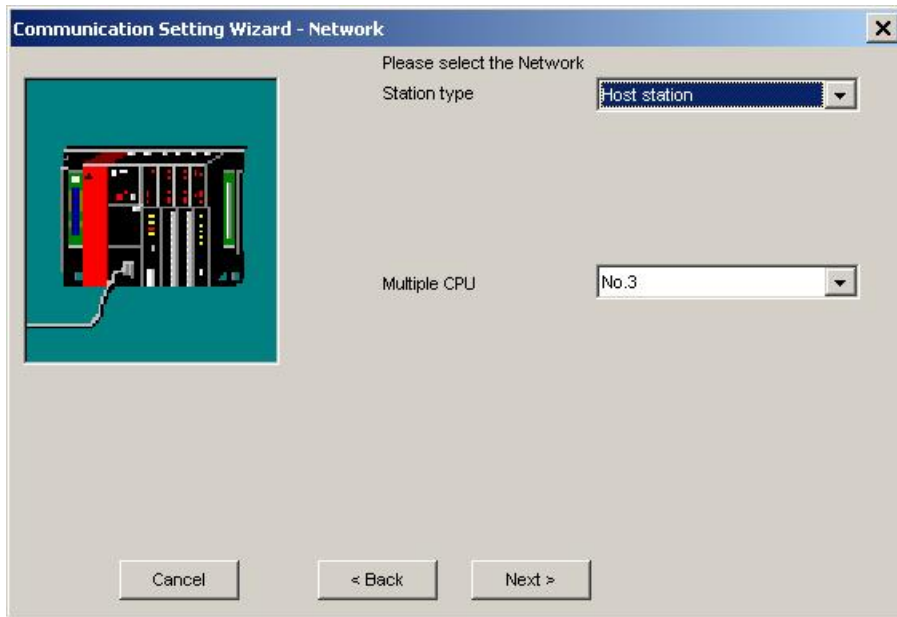
## 11.3 Special Setups

### 11.3.1 Communication with Multi-CPU Systems

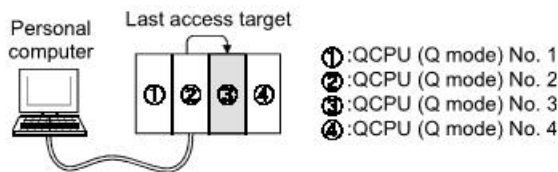
For a multi-CPU system the user must specify the CPU number of the target CPU, i.e. the last accessed station.

When making access to the non-control CPU of the relay module on the accessed station, use the modules of function version B as the relay modules and Q-CPU's (Q mode) on the own station, all relay stations and accessed station.

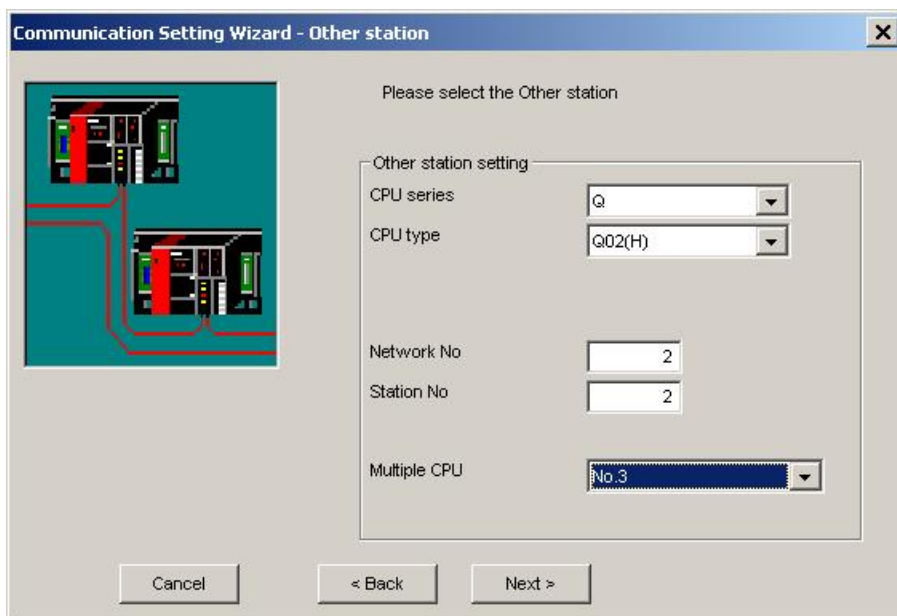
**(Example 1) CPU COM communication**



When the user has selected 'No. 3' (0x3E2) in the 'Multiple CPU' field, the connection is established to the third CPU in the rack, marked as (3) in the following figure.

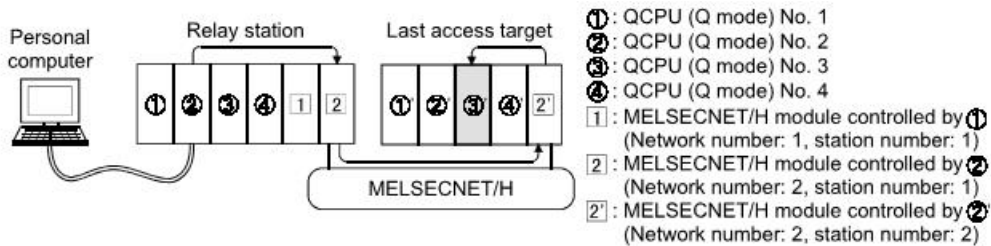


**(Example 2) CPU COM communication (via MELSECNET/H)**



When the user has selected 'No. 3' (0x3E2) in the 'Multiple CPU' field and entered '2' for network and station number, the connection is established to the third CPU in the 'Last access target' rack, marked as (3)' and grayed in the following figure.

The CPU number in the 'Relay station' rack cannot be specified. Therefore, if '1' has been entered as network number for the sample below, an error will occur, because the network number controlled by the CPU (2) is only '2'.

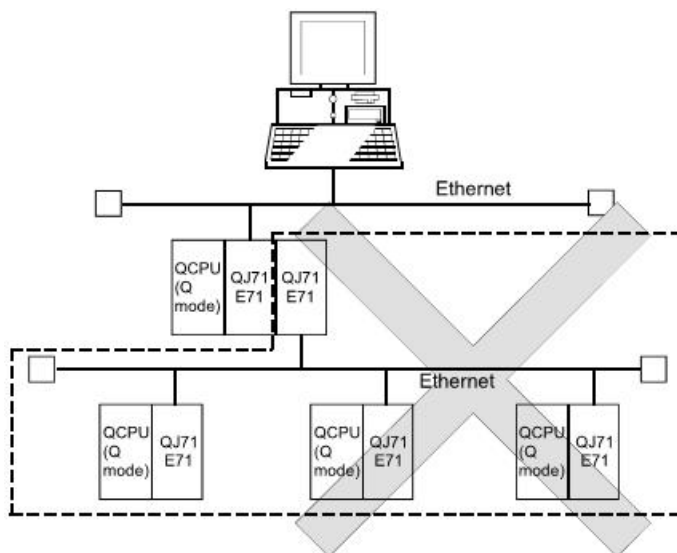


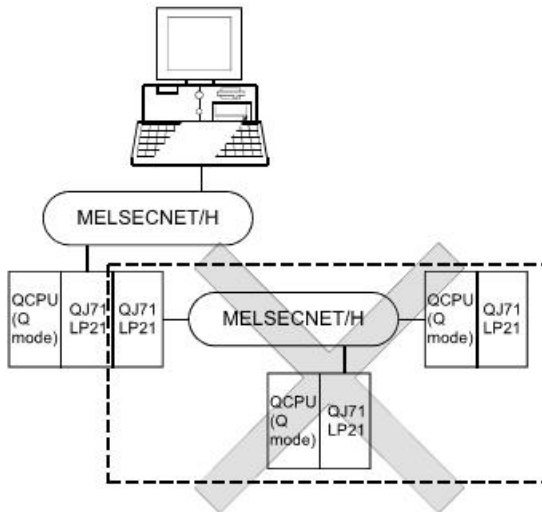
### 11.3.2 Restrictions with Q00, Q00J and Q01 CPUs

The number of network modules is restricted for Q00J, Q00 and Q01 CPUs.

Type of Network Module	Max. Number of Modules
MELSECNET/H module	1 module
Ethernet module	1 module
CC-Link module(Function version B or later)	2 modules

The following figures mark network architectures for Ethernet and MELSECNET/H, which are not supported in combination with Q00J, Q00 and Q01 CPUs.

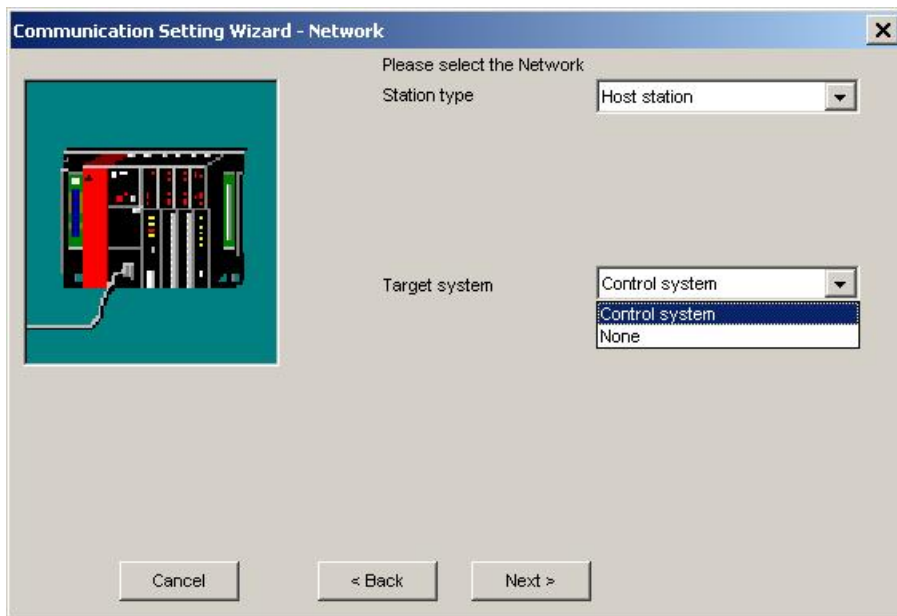




### 11.3.3 Redundant CPUs

The following text describes the configuration for accessing redundant Q series PLC systems, i.e. Q12PRH and Q25PRH.

#### (1) Target system



As 'Target system' either 'Control system' or 'None' can be selected.

Parameter	Values	Description
Target system	Control system	Connects to the control system and maintains access to the control system after switching control and standby CPU
	None	Connects to the target PLC CPU independently of whether it is control or standby system

**Note:** to determine which CPU in the redundant PLC system is being accessed, the following special relays can be checked.

(1) To distinguish, whether system A or B are accessed

SM1511	System identification flag	<ul style="list-style-type: none"> <li>identifies system A/system B of a redundant system.</li> <li>remains ON/OFF even if the tracking cable is disconnected while the redundant system is running.</li> </ul>
SM1512	System identification flag	

	System A	System B	during TRK.CABLE ERR. (= code 6120) (System not determined)
SM1511	ON	OFF	OFF
SM1512	OFF	ON	OFF

(2) When checking the operation system status

SM1515	Control/ Standby system status	<ul style="list-style-type: none"> <li>indicates the CPU module operation status</li> <li>remains ON/OFF even if the tracking cable is disconnected while the redundant system is running.</li> </ul>
SM1516		

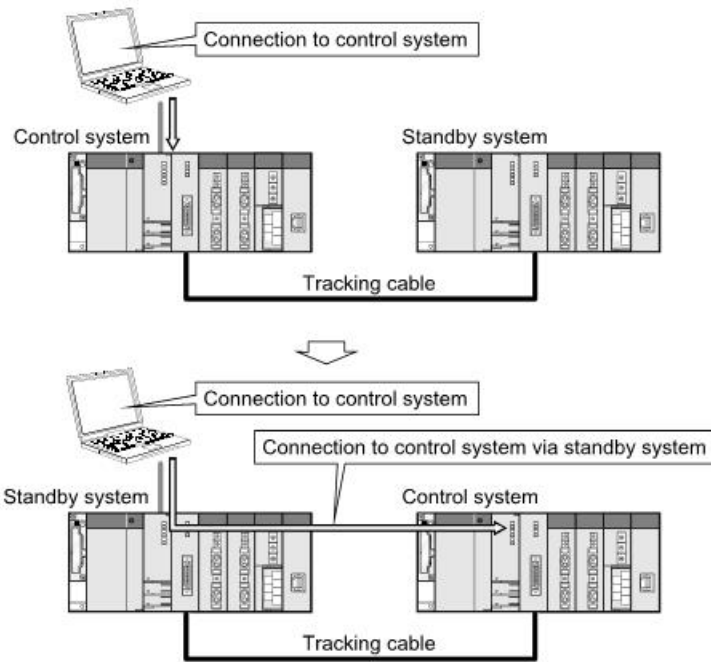
	Control system	Standby system	during TRK.CABLE ERR. (= code 6120) (System not determined)
SM1515	ON	OFF	OFF
SM1516	OFF	ON	OFF

## (2) Operation at occurrence of system switching

When system switching occurs during access to the redundant CPU and 'Control system' has been selected as target system, access is continued as described below.

(a) Connection via other than MELSECNET/H or Ethernet

Access to the control system after system switching is continued.

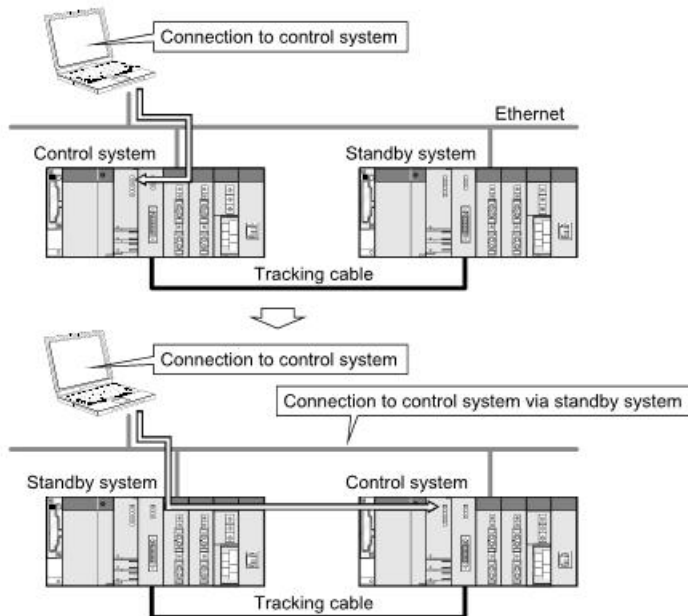


(b) Connection to MELSECNET/H or Ethernet

Access to the control system after system switching is continued as shown below, depending on whether a communication error occurs or not.

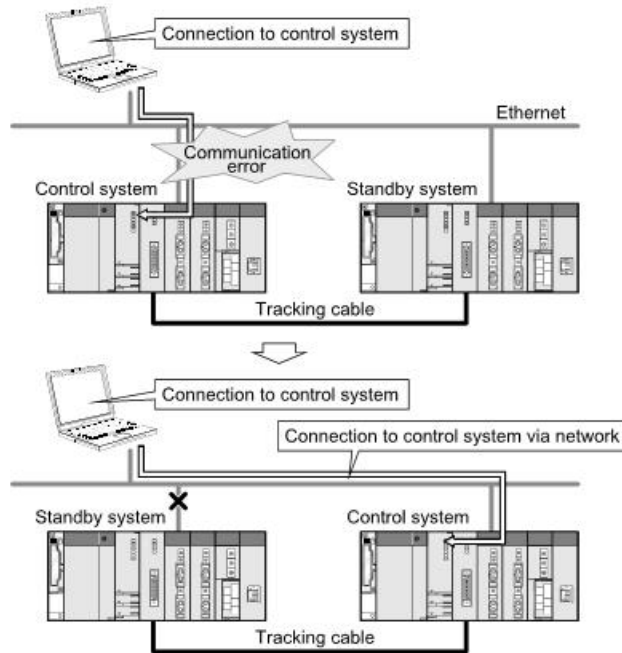
The following shows an example of Ethernet connection.

<When communication is normal>





<When communication error occurs>



**Note:** in the case of Ethernet connection, it may take time from when a communication error occurs until communication starts after connection to the control system.

### (3) Automatic switching of communication route

If a communication error occurs during access to the redundant CPU via a MELSECNET/H or Ethernet network, the

communication route is automatically switched to continue access to the control system.

In the following text this automatic switching of the communication route is referred to as 'route switching'.

The following describes the route switching conditions, how to check for route switching occurrence, and examples of access by route switching.

#### (a) Route switching conditions

When access is being made under the following conditions, access to the Redundant CPU is continued by route switching if a communication error occurs.

Mode	Conditions for continued access
Operation mode	Backup mode, Separate mode
Target system	Control system

However, if a tracking error had already occurred at a start of communication (caused for example by power off or reset of the redundant system), access to the control system is not continued by route switching even if tracking is recovered after that.

**(b) How to check for route switching occurrence and examples of access by route switching****1) How to check whether route switching occurred or not**

Special devices and registers allow to detect whether route switching has occurred.

SM1600 <sup>1</sup>	SD1590 <sup>2</sup>	SD1690 <sup>2</sup>	Possibility of route switching	Reference
OFF	either one is other than 0		Since a system switching request from the network module was detected, route switching may have been executed.	see example 1
ON	0	0	Since an other system fault occurred, route switching may have been executed.	see example 2
ON	either one is other than 0		Since an other system fault occurred or a system switching request from the network module was detected, route switching may have been executed.	see example 1 and 2

<sup>1</sup>: even if SM1600 is ON, route switching does not occur when the CPU is not accessed via the tracking cable.

<sup>2</sup>: when using SM1600, SD1590 and SD1690 to estimate whether route switching has occurred or not for the Redundant CPU connected to Ethernet, check the following items in the redundant setting of the network parameter dialog box of GX Developer.

- issue a system switching request at disconnection detection.
- issue a system switching request at communication error.

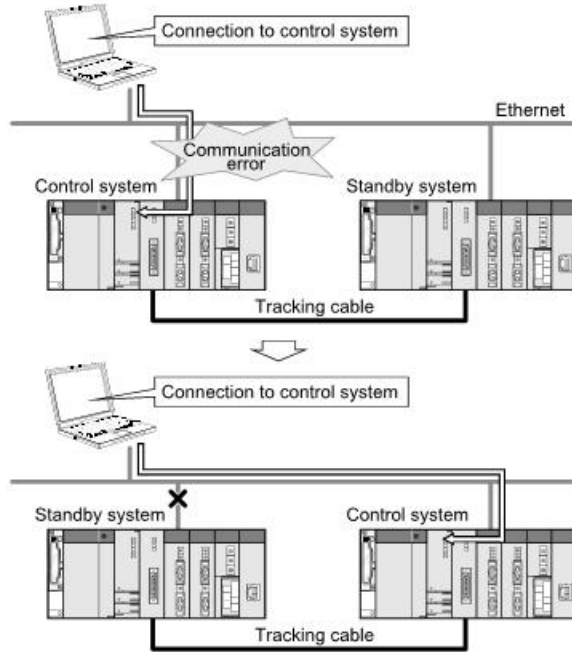
Check the following based on the states of the above special relay and special registers and remove the error cause.

- Check the Redundant CPU for an error.
- Check the tracking cable status and whether the tracking cable is correctly connected.
- Check the relevant network module for an error and the network where the relevant network module is connected for an error.

**2) Examples of access by route switching**

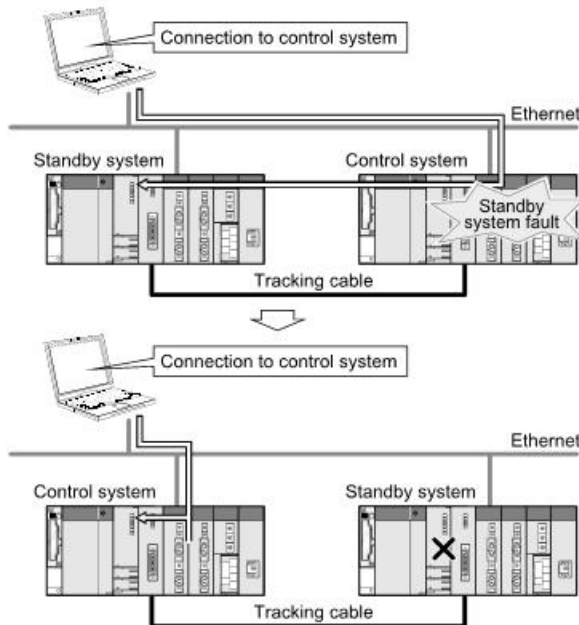
The following shows examples of route switching during access to the control system by Ethernet connection.

When system switching occurs at communication error:



*Route switching example 1*

When standby system fault occurs:



*Route switching example 2*

**Note:** the following table indicates details of the special relay and special registers to be monitored when estimating whether route switching occurred or not.

Device	Meaning	Explanation
SM1600	Other system error flag  OFF: No error  ON: Error	<ul style="list-style-type: none"> <li>turns on when an error occurs during redundant system error check. (Turns on when either of bits for SD1600.)</li> <li>remains off when no errors are present.</li> </ul>
SD1590	Module no. for network module requesting route switching in host system	<ul style="list-style-type: none"> <li>any of the following bits turns on corresponding to module No. for network module requesting route switching in host system.</li> <li>turns off by the system after recovery from error of the relevant module by user.</li> </ul> <div style="text-align: center;"> <p>Bit status 0: OFF 1: ON</p> <p>Module No. 0: Invalid, as CPU module uses 2 slots. Module No. 1: Indicates the module to the right of CPU module to Module No. 11: Indicates the module at the right end of a 12-I/O slot base (Q312B)</p> </div> <ul style="list-style-type: none"> <li>refer to SD1690 for module No. for network module requesting route switching in other system.</li> </ul>
SD1690	Module no. for network module requesting route switching in other system	<ul style="list-style-type: none"> <li>any of the following bits turns on corresponding to module No. for network module requesting route switching in other system.</li> <li>turns off by the system after recovery from error of the relevant module by user.</li> </ul> <div style="text-align: center;"> <p>Bit status 0: OFF 1: ON</p> <p>Module No. 0: Invalid, as CPU module uses 2 slots. Module No. 1: Indicates the module to the right of CPU module to Module No. 11: Indicates the module at the right end of a 12-I/O slot base (Q312B)</p> </div> <ul style="list-style-type: none"> <li>refer to SD1590 for module no. for network module requesting route switching in host system.</li> </ul>

### 11.3.4 Connection Types for Q24DHCCPU-V C-Controllers

The user interface of the transfer setup offers connection types for Q24DH CCPUs, which are not supported by this software.

The supported connection types are:

1. PC (USB) - Q24DHCCPU-V
2. PC (Ethernet) - Q24DHCCPU-V
3. PC (USB) via GOT (Bus) - Q24DHCCPU-V
4. PC (USB) via GOT (Ethernet) - Q24DHCCPU-V
5. PC (Ethernet) via GOT (Bus) - Q24DHCCPU-V

The following connection types can be selected in the transfer setup, but are presently not supported by the hardware except for multi-CPU numbers 2-4:

1. PC (Serial) - Q24DHCCPU-V
2. PC (C24) - Q24DHCCPU-V
3. PC (MELSECNET/H) - Q24DHCCPU-V
4. PC (CC-Link) - Q24DHCCPU-V
5. PC (CC-Link G4) - Q24DHCCPU-V
6. PC (CC-Link IE Control) - Q24DHCCPU-V
7. PC (CC-Link IE Field) - Q24DHCCPU-V
8. PC (Serial) - GOT (Serial) - Q24DHCCPU-V
9. PC (Serial) - GOT (C24) - Q24DHCCPU-V
10. PC (Serial) - GOT (Bus) - Q24DHCCPU-V
11. PC (Serial) - GOT (Ethernet) - Q24DHCCPU-V
12. PC (USB) - GOT (Serial) - Q24DHCCPU-V
13. PC (Ethernet) - GOT (Serial) - Q24DHCCPU-V

## 12 Appendix: Using OPC from Visual Basic

This section gives some simple examples of using OPC from a Visual Basic program. Examples will include browsing for installed OPC servers, browsing the defined item names and reading data from the server using synchronous calls.

### 12.1 Prerequisites

You will need:

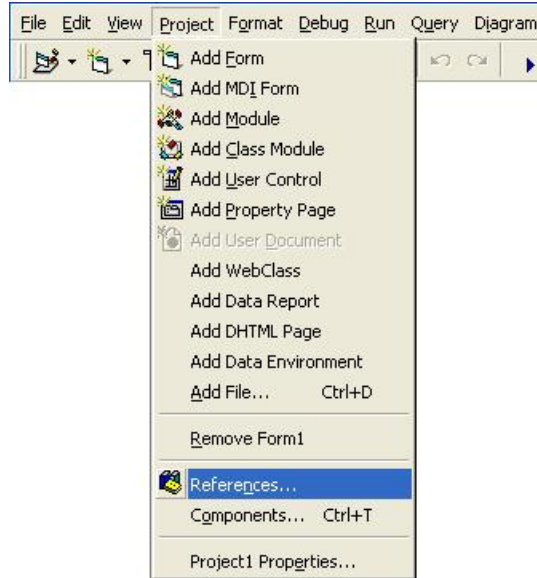
- Visual Basic - the examples in this document were developed with Visual Basic 6 Service Pack 5, but could also be used with other applications that will support VBA (Visual Basic for Applications).
- OPC automation support files (OPCdaAuto.dll and OPCenum.exe) – these must be correctly installed in e.g. the <WINDOWS\SYSTEM32> directory, and must be registered with REGSVR32.
- An installed OPC server, e.g. Mitsubishi.MXOPC.6

## 12.2 Terminology

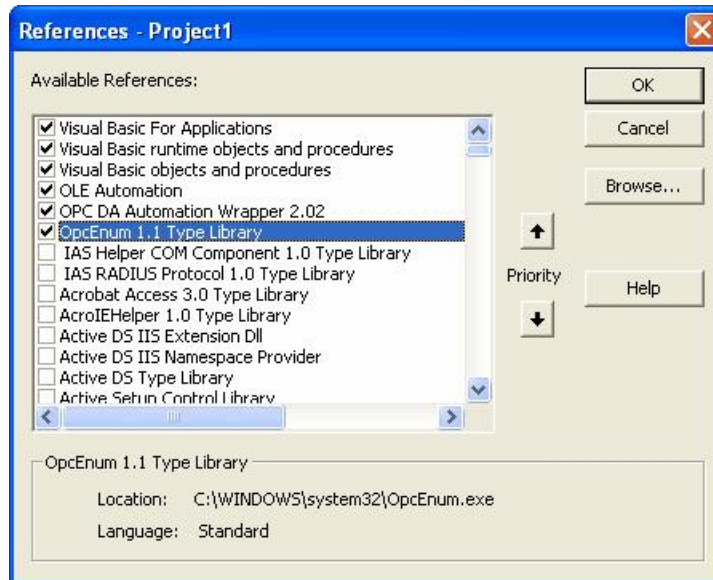
Term	Description
OPC	" <b>O</b> LE for <b>P</b> rocess <b>C</b> ontrol" – an open standard for connecting to process control equipment such as PLCs and SCADA databases. OPC is intended to solve the problem of every SCADA package needing a different driver for every PLC to which it connects. With OPC, only one driver needs to be developed, and each SCADA package connects to the OPC driver using the same interface.
Automation interface	A simplified OPC interface designed for integration with programs like Visual Basic and with scripting languages like VBA ( <b>V</b> isual <b>B</b> asic for <b>A</b> pplications).
OPCServer	The top level interface to OPC. This provides a way to find out which OPC servers are installed on a PC, and to connect to one of them.
OPCBrowser	This is the object used to list the items defined in the OPC server.
OPCGroup	A group of items to be read from the OPC server.
OPCGroups	Used to add groups to an OPC server and iterate through the available groups.
OPCItem	A single item of data in the OPC server, usually one I/O point.
OPCItems	Used to add items to an OPC group and iterate through the available items.
Server handle	A unique number used by the OPC server to identify a single item.
Client handle	A number used by the client to identify an OPC item.
Quality	This gives a measure of confidence in the data read from the server. In normal usage, most applications look for a quality value of 'OPC_QUALITY_GOOD' (&HC0) and treat any other value as an indication that the data is invalid.

### 12.3 Adding the OPC Libraries to Your Project

Before you can use the OPC automation libraries, they must be included in your project. To do this, select 'References' from the 'Project' menu:



In the list that appears, check 'OPC DA Automation Wrapper' and 'OPCenum' to include the OPC automation libraries in your project, then click OK.



## 12.4 OPC constants

Some of the more useful OPC constants are shown below, in a format that can be cut and pasted into a new module in a Visual Basic project.

```
' Data source constants - get cached value, or read direct from device
Public Const OPC_DS_CACHE = 1
Public Const OPC_DS_DEVICE = 2

' Quality and status constants
Public Const OPC_QUALITY_GOOD = &HC0
Public Const OPC_QUALITY_MASK = &HC0
Public Const OPC_STATUS_MASK = &HFC
Public Const OPC_LIMIT_MASK = &H3
Public Const OPC_QUALITY_BAD = &H0
Public Const OPC_QUALITY_UNCERTAIN = &H40
Public Const OPC_QUALITY_CONFIG_ERROR = &H4
Public Const OPC_QUALITY_NOT_CONNECTED = &H8
Public Const OPC_QUALITY_DEVICE_FAILURE = &HC
Public Const OPC_QUALITY_SENSOR_FAILURE = &H10
Public Const OPC_QUALITY_LAST_KNOWN = &H14
Public Const OPC_QUALITY_COMM_FAILURE = &H18
Public Const OPC_QUALITY_OUT_OF_SERVICE = &H1C
Public Const OPC_QUALITY_LAST_USABLE = &H44
Public Const OPC_QUALITY_SENSOR_CAL = &H50
Public Const OPC_QUALITY_EGU_EXCEEDED = &H54
Public Const OPC_QUALITY_SUB_NORMAL = &H58
Public Const OPC_QUALITY_LOCAL_OVERRIDE = &HD8

' Limit constants
Public Enum tOPClimits
    OPC_LIMIT_OK = 0
    OPC_LIMIT_LOW = 1
    OPC_LIMIT_HIGH = 2
    OPC_LIMIT_CONST = 3
End Enum ' tOPClimits

' Server states
Public Enum tServerState
    OPC_STATE_RUNNING = 1
    OPC_STATE_FAILED = 2
    OPC_STATE_NOCONFIG = 3
    OPC_STATE_SUSPENDED = 4
    OPC_STATE_TEST = 5
    OPC_STATE_DISCONNECTED = 6
End Enum ' tServerState
```



## 12.5 Listing installed OPC servers

To find out which OPC servers are installed, create a new OPCServer object and call the GetOPCServers() function. This returns an array of strings giving the names of installed servers. For Visual Basic purposes, the variable used should be a 'Variant'.

The name of the Mitsubishi OPC server is 'Mitsubishi.MXOPC.6', but when connecting to a server the shorter name 'Mitsubishi.MXOPC' can also be used. This will connect to the latest available version of the MX OPC server.

```
Dim objServer as OPCServer      ' OPC server object
Dim vAllServers as Variant      ' Temporary to build list of servers
Dim intLp as integer           ' Loop counter
Dim strS As String              ' String to build list of servers

' Create an OPC server object
Set objServer = New OPCServer

' Use the server object to populate the list of available OPC servers
vAllServers = objServer.GetOPCServers

' Display the name of each server
For intLp = LBound(vAllServers) To UBound(vAllServers)

    ' Add name to list
    strS = strS & vAllServers(intLp) & vbCrLf

Next ' Server

' Display results list
MsgBox strS, , "List of servers"

' Clean up
set objServer = Nothing
```

## 12.6 Connecting to an OPC server

To connect to an OPC server, create an OPCServer object and call the 'Connect' method, passing it the name of the server. This name can be found from the GetOPCServers() function described above.

```
Dim objServer as OPCServer      ' OPC server object

' Create server object if necessary
If objServer Is Nothing Then Set objServer = New OPCServer

' If already connected, disconnect first
If objServer.ServerState = OPC_STATE_RUNNING Then objServer.Disconnect

' Connect to the server
objServer.Connect "Mitsubishi.MXOPC"

' Check server is running
Debug.Assert objServer.ServerState = OPC_STATE_RUNNING

' Show server version and details
MsgBox "Server: " & objServer.VendorInfo & _
    " (" & Format(objServer.MajorVersion) & "." & _
    Format(objServer.MinorVersion) & ")", , _
    "Connected to server"
```

## 12.7 Browsing the OPC server items

The example below will show how to browse the list of items defined on the server hierarchically, writing the results into a treeview control called `twvBrowse`. A subroutine is used to browse the structure recursively, and for leaf (tag) items the tag's item ID (which is needed to read its value from the server) will be stored in the leaf node's `.Key` property.

Browsing for tags is not always necessary. If you already know which tags the application needs to read, you can use their IDs directly with the read and write functions (see sections [Reading data values from the server](#)<sup>[369]</sup> and [Writing data values to the server](#)<sup>[371]</sup>). For the MX OPC server, these item IDs are built from the device name followed by any group names and finally the tag name, with each part separated by period characters. Typical examples would be `'Dev01.Tag001'` (without groups) or `'Dev02.Group001.Tag001'` (with one group).

The IDs for dynamic tags are built differently, using a 'backwards slash' (`'\'`) as the separator, and with the tag name in a format which also gives the address and type, for example `'Dev01\D50.'` for a dynamic tag reading address D50 as an integer, or `'Dev01\G1\D100.S.10'` for a 10 character string under a group called 'G1'. For more examples, see ["Dynamic tag support"](#)<sup>[159]</sup>.

```
' Constants for treeview nodes - will be put in the 'Tag' property
Const TREE_ROOT As String = "0"
Const TREE_BRANCH As String = "1"
Const TREE_LEAF As String = "2"

' Constant for root node's key
Const strRootKey As String = "ROOT"

' Name:          FillBranches()
' Description:   Fill treeview with items and branches from OPC hierarchy.
'               Will call itself recursively to get the details.
' Inputs:       tvwT          - Treeview to fill
'               objBrw       - OPC browser object at current position
'               (initially root)
'               nodPar       - Parent node to fill under
'               (initially a new, root, node)
' Returns:      N/A
Private Sub FillBranches(ByRef tvwT As TreeView, _
                       ByRef objBrw As OPCBrowser, _
                       ByRef nodPar As Node)

    ' Declare variables
    Dim intLp As Integer    ' Loop counter
    Dim nodN As Node       ' Used for node creation

    ' Find branches at this level
    objBrw.ShowBranches
    For intLp = 1 To objBrw.Count
        ' Skip 'register hints' branch, no readable tags
        If objBrw.Item(intLp) <> "Register Hints" Then
            ' Add branch node (no key, not unique for branches)
            Set nodN = tvwT.Nodes.Add(nodPar, tvwChild, , _
                                     objBrw.Item(intLp))

            nodN.Tag = TREE_BRANCH
        End If ' Not register hints branch
    Next intLp ' Branch

    ' Find child nodes at this level
    objBrw.ShowLeafs
    For intLp = 1 To objBrw.Count
        ' Add leaf node
        Set nodN = tvwT.Nodes.Add(nodPar, tvwChild, _
                                  objBrw.GetItemID(objBrw.Item(intLp)), _
                                  objBrw.Item(intLp))

        nodN.Tag = TREE_LEAF
    Next intLp ' Leaf
```

```

' Now go back and recursively fill the branches we just created
' (can't do them as we go because each call to ShowBranches
'   overwrites the data)
Set nodN = nodPar.Child
While Not nodN Is Nothing

    ' Only interested in branch nodes
    If nodN.Tag = TREE_BRANCH Then

        ' Move down into this branch
        objBrw.MoveDown nodN.Text

        ' Fill in details recursively
        FillBranches tvwT, objBrw, nodN

        ' Move back up
        objBrw.MoveUp

    End If ' Branch node

    ' Next node
    Set nodN = nodN.Next

Wend ' Until all nodes checked

' Clean up object references
Set nodN = Nothing

End Sub ' FillBranches()

' When button is pressed, read list of items into treeview
Private Sub cmdGetItemsList_Click()

    Dim objBrowse As OPCBrowser ' OPC hierarchy browser object
    Dim nodRoot As Node ' Used to create root node

    ' Create a browser
    Set objBrowse = objServer.CreateBrowser

    ' Start at first item
    objBrowse.MoveToRoot

    ' Use '.' as path separator, so we get device.item notation
    ' when reading back
    tvwBrowse.PathSeparator = "."

    ' Fill in treeview (first adding a root node)
    tvwBrowse.Nodes.Clear
    Set nodRoot = tvwBrowse.Nodes.Add(, , strRootKey, _
        cmbServer.List(cmbServer.ListIndex))
    FillBranches tvwBrowse, objBrowse, nodRoot

    ' Close browser object
    Set objBrowse = Nothing

End Sub ' cmdGetItemsList_Click()

```

The routines below can be used to read the list of items back out of the treeview control. Again, a recursive subroutine is used to extract the details.

```

Dim strItemNames() as string ' Array for results
Dim lngItems as Long ' Number of items found

' List the items in the tree
Private Sub ListItems()

    lngItems = 0
    ReDim strItemNames(1 To 1)
    GetItemNames tvwBrowse.Nodes(strRootKey), strItemNames, lngItems

End Sub ' ListItems()

' Name: GetItemNames()
' Description: Routine to extract item names from the treeview
' Will traverse the tree recursively
' Inputs: nodPar - Parent node in treeview at which to start
' strDest() - Redimensionable array of item names to fill
' lngCnt - Number of items read so far
' Returns: N/A
Private Sub GetItemNames(ByRef nodPar As Node, _
    ByRef strDest() As String, _
    ByRef lngCnt As Long)
    Const intReallocStep = 10 ' Step threshold to resize target array

    ' Go through nodes, recursing as required until target array is full
    Dim nodN As Node ' Used for traverse/recurse

    ' Work through child nodes of parent node provided
    Set nodN = nodPar.Child

    ' Repeat until run out of nodes
    While Not nodN Is Nothing

        ' If this is a leaf node, collect its name. If a branch node,
        ' descend it
        If nodN.Tag = TREE_BRANCH Then
            ' Descend recursively
            GetItemNames nodN, strDest, lngCnt
        Else
            ' Leaf node, add it
            lngCnt = lngCnt + 1

            ' Redimension target array if required
            If lngCnt > UBound(strDest) Then
                ' Resize target array (but not above maximum)
                ReDim Preserve strDest(1 To lngCnt + intReallocStep)
            End If ' Array too small

            ' Fill in a single item name
            strDest(lngCnt) = nodN.Key

        End If ' Branch / leaf node

        ' Try next node (if any)
        Set nodN = nodN.Next
    Wend ' Until no more nodes or target array is full

    ' Clean up object references
    Set nodN = Nothing

End Sub ' GetItemNames()

```

## 12.8 Reading data values from the server

To read item values from the server, first create a new OPCGroup object to hold the items. You can define more than one group if you have different sets of items that are read at different times.

Once items are added to the group as OPCItems, they can be read individually or as a group. When the items are defined, their server 'handles' should be stored for use as parameters in the calls to the read functions.

In the example below (in which the error handling code is omitted for readability), a new group will be created and all the items found by the browse example in the previous section will be added to it. The items' values will then be read synchronously as a group.

```
Private Sub TestRead()

    Dim lngLp As Long           ' Loop counter
    Dim strGrpName As String    ' Group name
    Dim objGrp As OPCGroup      ' Group object
    Dim varValue As Variant     ' Used to read OPC item values
    Dim varQual As Variant     ' Used to read OPC quality
    Dim objItem As OPCItem     ' Used to get server handles for items

    ' Arrays for results
    Dim lngSvrHandles() As Long ' Server handles
    Dim lngErrors() As Long     ' Error values
    Dim arrvarValues() As Variant ' Item values

    ' Result string for message box (first 20 items only)
    Dim strS As String

    ' Resize arrays for required number of items
    ReDim lngSvrHandles(1 To lngItems)
    ReDim lngErrors(1 To lngItems)
    ReDim arrvarValues(1 To lngItems)

    ' Create group in server. Name will be e.g. 'Items200'
    strGrpName = "Items" & Format(lngItems)
    Set objGrp = objServer.OPCGroups.Add(strGrpName)

    ' Create an OPC item for each item name
    For lngLp = 1 To lngItems

        ' Add object, get reference
        Set objItem = objGrp.OPCItems.AddItem(strItemNames(lngLp), _
            CLng(lngLp))

        ' Check that item was defined correctly
        If objItem Is Nothing Then
            ' Failed to define item, probably missing
            MsgBox "Failed to create server item for item '" & _
                strItemNames(lngLp) & "' - does the item exist?"
            Exit For
        Else
            ' Item defined correctly - get server handle
            lngSvrHandles(lngLp) = objItem.ServerHandle

            ' Clean up reference
            Set objItem = Nothing
        End If ' Item failed to define / defined OK
    Next lngLp ' Test item to create

    ' Read all items as a group
    objGrp.SyncRead OPC_DS_DEVICE, lngItems, lngSvrHandles, _
        arrvarValues, lngErrors, varQual
```

```
' Check results for each item
For lngLp = 1 To lngItems

    ' Check quality - if quality is bad, data is invalid
    If varQual(lngLp) <> OPC_QUALITY_GOOD Then

        ' Note reason and offending item name
        MsgBox "Bad quality (0x" & Hex(varQual(lngLp)) & _
            ") for item '" & strItemNames(lngLp) & "'"
        Exit For

    Else

        ' Show value on debug console
        Debug.Print strItemNames(lngLp) & " = " & _
            Format(arrvarValues(lngLp))

        ' Add to list (first 20 only)
        If lngLp <= 20 Then strS = strS & strItemNames(lngLp) & _
            " = " & Format(arrvarValues(lngLp)) & vbCrLf

    End If ' Quality bad / OK

Next lngLp ' Item to check

' Show results
MsgBox strS, , "Item values"

'
' Must clean up object references to release
' memory and resources in the OPC server
'

' Remove items from group
objGrp.OPCItems.Remove lngItems, lngSvrHandles, lngErrors

' Release group object to prevent memory/resource leaks
Set objGrp = Nothing

' Remove created group from OPC server
objServer.OPCGroups.Remove strGrpName: strGrpName = ""

' Clean up array variables
Erase arrvarValues
Erase lngSvrHandles
Erase lngErrors

End Sub ' TestRead
```

## 12.9 Writing data values to the server

The example below shows how to set the value of an item in the server. Although only one value is written, an array of values and item handles is used, so that it can easily be adapted to write multiple items.

```
Dim varVal() As Variant      ' Value to write as a variant
ReDim varVal(1 To 1)        ' Must use resizeable arrays
varVal(1) = 10              ' Sample value

' Now ready to write the value back to the server
Dim objGrp As OPCGroup     ' Group object
Dim objItem As OPCItem     ' Item object
Dim lngSvrHandles() As Long ' Required to remove items
Dim lngErrors() As Long   ' Error values
ReDim lngSvrHandles(1 To 1) ' Redimension arrays to right size
ReDim lngErrors(1 To 1)   ' (must be resizable arrays)

' Create a group for the items to write
Set objGrp = objServer.OPCGroups.Add("WriteGroup")

' Add object, get reference
Set objItem = objGrp.OPCItems.AddItem(strID, 1)

' Check that item was defined correctly
If objItem Is Nothing Then
    ' Failed to define item, probably missing
    MsgBox "Failed to create server item for item '" & _
        strID & "' - does it exist?"
Else
    ' Get server handle, required to remove the item
    ' from the group later
    lngSvrHandles(1) = objItem.ServerHandle

    ' Write values for all items in this group
    objGrp.SyncWrite lngSvrHandles, varVal, lngErrors

    ' (N.b. an alternative method to write a single item is:
    '   objItem.Write varVal(1)
    ' This is less efficient than a group write for many items)

    ' Show success
    MsgBox "Wrote " & Format(varVal(1)) & " to " & strID

    ' Release item reference
    Set objItem = Nothing

    ' Remove items from group
    objGrp.OPCItems.Remove lngSvrHandles, lngErrors
End If ' Not defined / OK

' Release group object to prevent memory/resource leaks
Set objGrp = Nothing

' Remove created group from OPC server
objServer.OPCGroups.Remove "WriteGroup"
```

## 12.10 Disconnecting from an OPC server

To disconnect from an OPC server, first check that the connection is not already disconnected, then call the 'Disconnect' method. After disconnecting, set the server variable to the special value 'Nothing' to ensure that the object is destroyed cleanly. If this is not done, Visual Basic will not destroy the object, causing memory and resource leaks until the module or program containing the object is unloaded. It is also recommended that the 'End' statement is used in the last Form\_Unload of the program, to make sure every resource used is released.

```
' Disconnect from the server if not already disconnected
If Not objServer Is Nothing Then

    ' Close connection if connected
    If objServer.ServerState <> OPC_STATE_DISCONNECTED Then
        objServer.Disconnect
    End If ' Connected

    ' Clean up OPC server object
    Set objServer = Nothing

End If ' OPC server object was created

' Normally used in the Form_Unload event - unload all forms, release
' all memory and terminate the program.
End
```

## 12.11 References

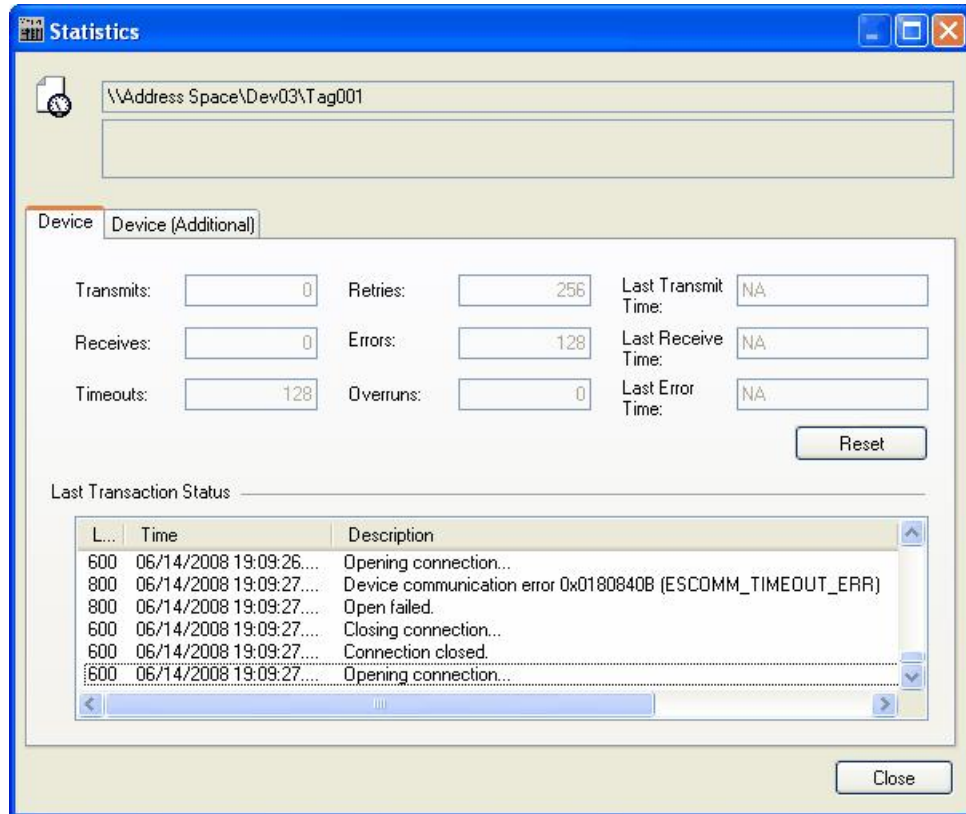
The references below are recommended for further information about using OPC from Visual Basic, and for information about OPC in general.

- OPC Data Access Automation Specification
- <http://www.opcfoundation.org>
- [http://www.iconics.com/support/free\\_tools.asp](http://www.iconics.com/support/free_tools.asp)



## 13 Appendix: Understanding ACT error codes

When there are problems with the communication to an MX device, the statistics window may show a more detailed error code, as shown in the example screenshot below.



Statistics window showing error number

This appendix will describe the error codes that may be returned for MX devices when using the MX Component communications layer. These error codes are divided into three main types:

- HRESULT error codes, which are general failure codes and do not necessarily indicate a problem with the device – see "[HRESULT error codes](#)"<sup>[374]</sup>
- ACT error codes, returned by the communications layer – see "[ACT error codes](#)"<sup>[375]</sup>
- CPU, module and network board error codes, returned by a particular device – see "[CPU, module and network board errors](#)"<sup>[419]</sup>

### 13.1 HRESULT error codes

These are general ActiveX errors, which are not necessarily related to the device settings or status.

Value	Name	Type	Description
0x00000000	S_OK	Normal termination	Function processing terminated normally.
0x00000001	S_FALSE	Normal termination	Function processing (as ActiveX control) terminated normally, but operation (access to PLC) failed.
0x80004003	E_POINTER	Abnormal termination	The pointer passed to the function is invalid.
0x80070006	E_HANDLE	Abnormal termination	A supplied handle value was invalid.
0x8007000E	E_OUTOFMEMORY	Abnormal termination	Memory allocation or object creation failed.
0x80070057	E_INVALIDARG	Abnormal termination	An incorrect argument was provided.
0x80070005	E_ACCESSDENIED	Abnormal termination	Insufficient permissions to use the object.
0x8000FFFF	E_UNEXPECTED	Abnormal termination	An unexpected error occurred.
0x80004005	E_FAIL	Abnormal termination	An unspecified error occurred.

## 13.2 ACT error codes

These error codes will often help to diagnose connection problems, as they can show issues such as the wrong parameters being set for the connection.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x01802002	Device number error. The device character string number specified in the method is an unauthorised device number.	Review the device number.
0x01802003	Program Type Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802004	Sumcheck error. The sumcheck value of the received data is abnormal.	Check the module side sumcheck setting. Check the sumcheck property of the control. Check the cable. Exit the program and restart the computer. Reinstall MX Component.
0x01802005	Size error. The number of points specified in the method is unauthorised.	Check the number of points specified in the method. Confirm that the address range being read does not exceed the range configured in the PLC. Review the system, e.g. PLC CPU, module setting and cable status. Exit the program and restart the computer. Reinstall MX Component.
0x01802006	Block number error. The block specifying number in the device character string specified in the method is unauthorised.	Review the block specifying number in the device character string specified in the method.
0x01802007	Receive data error. The data received is abnormal.	Review the system, e.g. PLC CPU, module setting and cable status. Check the cable. Exit the program and restart the computer.
0x01802008	Write Protect Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802009	Reading Parameters error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180200A	Writing Parameters error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180200B	PLC type mismatch The CPU type set to the property and the CPU type set on the communication settings utility do not match the CPU type on the other end of communication.	Set the correct CPU type as the CPU type of the property. Set the correct CPU type on the communication settings utility. Review the system, e.g. PLC CPU, module setting and cable status.
0x0180200C	Request Cancel Error. The request was cancelled while being processed.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x0180200D	Drive Name Error. The specified drive name is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180200E	Beginning Step Error. The beginning step specified is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180200F	Parameter Type Error. The parameter type is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802010	File Name Error. The file name is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802011	Status Error. The status of Registration / Cancellation / Setting is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802012	Detailed Condition Field Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802013	Step Condition Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802014	Bit Device Condition Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802015	Parameter Settings Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802016	Error in specifying telephone exchange number. Method does not support the operations corresponding to the specified telephone exchange number.	Check the telephone exchange number. Check if the method being executed is supported or not. Check the system configuration such as PLC, unit, etc.
0x01802017	Keyword Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802018	Read/Write Flag Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802019	Refresh Method Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180201A	Buffer Access Method Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x0180201B	Start Mode/Stop Mode Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180201C	Written clock data error. Clock data specified for write cannot be written properly since that data is in error.	Review the clock data to be written.
0x0180201D	Online clock data write error. Write of clock data failed. Clock data cannot be written since the PLC CPU is in RUN.	Place the PLC CPU in the STOP status.
0x0180201E	ROM drive Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180201F	While Tracing error. Invalid operation was carried out during trace.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802020	First I/O number error. The first I/O number specified in the method is an unauthorised value.	Check the value of the first I/O number specified in the method. Using the GPP function, check the PLC CPU parameters (I/O assignment). Exit the program and restart the computer.
0x01802021	First address error. The buffer address specified in the method is an unauthorised value.	Check the value of the buffer address specified in the method. Exit the program and restart the computer.
0x01802022	Pattern Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802023	SFC Block No. Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802024	SFC Step No. Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802025	Step No. Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802026	Data Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802027	System Data Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802028	Error in number of TC settings Value	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x01802029	Clear Mode Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180202A	Signal Flow Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180202B	Version Control Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180202C	Monitor Not Registered error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180202D	PI Type Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180202E	PI No Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180202F	Error in Number of PIs	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802030	Shift Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802031	File Type Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802032	Specified Unit error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802033	Error check flag Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802034	Step RUN operation error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802035	Step RUN data error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802036	During Step RUN error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802037	Write error while running program corresponding to E2PROM	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x01802038	Clock data read/write error The clock data read/write method was executed for the PLC CPU which does not have the clock devices.	Do not execute clock data read/write.
0x01802039	Trace not completed error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180203A	Registration Clear Flag Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180203B	Operation error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180203C	Error in the number of exchanges	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180203D	Error in number of loops specified	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180203E	Retrieve data selection	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180203F	Error in number of SFC cycles	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802040	Motion PLC Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802041	Motion PLC Communication error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802042	Fixed execution time setting error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802043	Error in number of functions	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802044	System information specification error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802045	Registration Condition Not Formed error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x01802046	Function No. Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802047	RAM drive error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802048	ROM drive error at the booting side	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802049	Transfer mode specification error at the booting side	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180204A	Insufficient memory error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180204B	Back up drive ROM error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180204C	Block size error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180204D	Detached during RUN state error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180204E	Unit Already Registered error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0180204F	Password Registration Data Full error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802050	Password Not Registered error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802051	Remote Password Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802052	IP Address Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802053	Timeout value out of range error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802054	Command not detected error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.



<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x01802055	Trace execution type error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802056	Version error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01802057	Tracking cable error. The tracking cable is faulty. The PLC CPU status is error.	Reexamine the system such as the PLC CPU, module setting and cable status.
0x0180205C	Keyword protection error PLC is protected by the key word.	Disable the keyword and execute again.
0x0180205D	Keyword disable error. The inputted keyword is wrong.	Input a correct keyword.
0x0180205E	Keyword protecting error. PLC did not accept the protecting command.	Execute again or re-switch the power of the PLC.
0x0180205F	Keyword entry error. An illegal character is included in the inputted keyword.	Input a correct keyword.
0x01802060	Keyword deletion error. The inputted keyword is wrong.	Input a correct keyword.
0x01808001	Multiple Open Error. Open method was executed while it was open	Exit the program and restart the computer. Execute any method other than Open.
0x01808002	Channel number specifying error. The port number set to the property and the port number set on the communication settings utility are unauthorised values.	Set the correct value to the port number of the property. Make communication settings again on the communication settings utility.
0x01808003	Driver not yet started. The network board driver is not started.	The network board driver is not started. Start the driver.
0x01808004	Error in overlap event generation	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01808005	MUTEX generation error. Creation of MUTEX to exercise exclusive control failed.	Exit the program and restart the computer. Reinstall MX Component.
0x01808006	Error in socket object generation. Socket object could not be created	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x01808007	Socket object generation error. Creation of the Socket object failed.	Check for a running application which uses the same port number. Retry after changing the port number value of the property. Retry after changing the port number value on the communication settings utility. Make Ethernet board and protocol settings on the control panel of the OS. Exit the program and restart the computer.
0x01808008	Port connection error. Establishment of connection failed. The other end does not respond.	Review the IP address and port number values of the properties. Review the port number value on the communication settings utility. Review the system, e.g. PLC CPU, module setting and cable status. Exit the program and restart the computer.
0x01808009	COM port handle error. The handle of the COM port cannot be acquired. The COM port object cannot be copied. The SOCKET object cannot be copied.	Check for an application which uses the COM port. Exit the program and restart the computer.
0x0180800A	Buffer size setting error. Setting of the COM port buffer size failed.	Check for an application which uses the COM port. Make COM port setting on the control panel of the OS. Exit the program and restart the computer.
0x0180800B	DCB value acquisition error. Acquisition of the COM port DCB value failed.	Check for an application which uses the COM port. Make COM port setting on the control panel of the OS. Exit the program and restart the computer.
0x0180800C	DCB setting error. Setting of the COM port DCB value failed.	Check for an application which uses the COM port. Make COM port setting on the control panel of the OS. Exit the program and restart the computer.
0x0180800D	Time-out value setting error. Setting of the COM port time-out value failed.	Review the time-out value of the property. Review the time-out value on the communication settings utility. Check for an application which uses the COM port. Make COM port setting on the control panel of the OS. Exit the program and restart the computer.
0x0180800E	Shared memory open error. Open processing of shared memory failed.	Check whether the GX Simulator has started. Exit the program and restart the computer.
0x01808101	Duplex close error	Exit the program and restart the computer.
0x01808102	Handle close error. Closing of the COM port handle failed.	Exit the program and restart the computer.
0x01808103	Driver close error. Closing of the driver handle failed.	Exit the program and restart the computer.
0x01808104	Overlap Event Close Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01808105	Mutex Handle Close Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x01808106	COM Port Handle Close Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01808201	Send error Data send failed.	Review the system, e.g. PLC CPU, module setting and cable status. Make COM port setting on the control panel of the OS. Make Ethernet board and protocol settings on the control panel. Retry the method. Exit the program and restart the computer.
0x01808202	Send data size error. Data send failed.	Exit the program and restart the computer.
0x01808203	Queue clear error. Clearing of the COM port queue failed.	Exit the program and restart the computer. Perform Close once and execute Open again.
0x01808301	Receive error. Data receive failed.	Review the system, e.g. PLC CPU, module setting and cable status. Review the time-out value of the property. Review the time-out value on the communication settings utility. Retry the method. Exit the program and restart the computer.
0x01808302	Not Sent error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01808303	Error in retrieving Overlap Event	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01808304	Receive buffer size shortage. Receive data was larger than the receive buffer size prepared for the system.	Exit the program and restart the computer.
0x01808401	Control error. Changing of the COM port communication control failed.	Exit the program and restart the computer.
0x01808402	Signal Line Control Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01808403	Signal line specifying error. Changing of the COM port communication control failed.	Exit the program and restart the computer.
0x01808404	Open not yet executed	Execute Open. Exit the program and restart the computer.
0x01808405	Communication parameter error. The data bit and stop bit combination of the properties is unauthorised.	Review the data bit and stop bit values of the properties. Set them again on the communication settings utility.
0x01808406	Transmission speed value specifying error. The transmission speed of the property is unauthorised.	Review the transmission speed value of the property. Set it again on the communication settings utility.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x01808407	Data length error. The data bit value of the property is unauthorised.	Review the data bit value of the property. Set it again on the communication settings utility.
0x01808408	Parity specifying error. The parity value of the property is unauthorised.	Review the parity value of the property. Set it again on the communication settings utility.
0x01808409	Stop bit specifying error. The stop bit value of the property is unauthorised.	Review the stop bit value of the property. Set it again on the communication settings utility.
0x0180840A	Communication control setting error. The control value of the property is unauthorised.	Review the control value of the property. Set it again on the communication settings utility.
0x0180840B	Time-out error. Though the time-out period had elapsed, data could not be received.	Review the time-out value of the property. Set it again on the communication settings utility. Review the system, e.g. PLC CPU, module setting and cable status. Perform Close once and execute Open again. Exit the program and restart the computer.
0x0180840C	Connect error	Exit the program and restart the computer.
0x0180840D	Duplex connect error	Exit the program and restart the computer.
0x0180840E	Attach failure. Attaching of the socket object failed.	Exit the program and restart the computer.
0x0180840F	Signal line status acquisition failure. Acquisition of the COM port signal line status failed.	Exit the program and restart the computer.
0x01808410	CD signal line OFF. The CD signal on the other end of communication is in the OFF status.	Review the system, e.g. PLC CPU, module setting and cable status. Exit the program and restart the computer.
0x01808411	Password mismatch error	Check the remote password of the property.
0x01808412	TEL Communication Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01808501	USB driver load error. Loading of the USB driver failed.	Exit the program and restart the computer. Reinstall MX Component.
0x01808502	USB driver connect error. Connection of the USB driver failed.	Exit the program and restart the computer. Reinstall MX Component.
0x01808503	USB driver send error. Data send failed.	Review the system, e.g. PLC CPU, module setting and cable status. Make USB setting on the control panel (device manager) of the OS. Retry the method. Exit the program and restart the computer.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x01808504	USB driver receive error. Data receive failed.	Review the system, e.g. PLC CPU, module setting and cable status. Make USB setting on the control panel (device manager) of the OS. Retry the method. Exit the program and restart the computer.
0x01808505	USB Driver Timeout Error	Recheck the timeout value. Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x01808506	USB driver initialisation error. Initialisation of the USB driver failed.	Make USB setting on the control panel (device manager) of the OS. Exit the program and restart the computer.
0x01808507	Other USB error. Error related to data send/receive occurred.	Disconnect the cable once, then reconnect. Exit the program and restart the computer. Reinstall MX Component.
0x02000001	Points Exceeded error. The number of points registered in the monitoring server is very high.	Reduce the no. of points registered by the monitor. Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x02000002	Shared memory creation error. Failed in creating shared memory.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x02000003	Shared memory access error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x02000004	Memory Secure error. Failed in securing memory for the monitoring server.	Close the other applications. Increase the system memory. Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x02000005	Device Not Registered error. Monitor has not been registered	Register the monitor in the monitoring server. Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x02000006	Monitoring Server Startup Error. Monitoring Server is not started.	Start the Monitoring Server. Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x02000010	Yet to retrieve Device Value error. Monitoring is not yet completed	Try to retrieve the value again after waiting for a fixed amount of time. Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03000001	Command not Supported. Command is not supported.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03000002	Memory Lock Error Failed while locking memory.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03000003	Error Securing Memory. Failed in securing the memory.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x03000004	DLL read error. Failed in reading DLL.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03000005	Error in securing Resources. Failed in securing the resources.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03010001	File Creation Error. Failed in creating the file.	Check if there is enough space on the hard disk. Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03010002	File Open Error. Failed to open the file.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03010003	Buffer Size Error. The buffer size specified is either incorrect or not enough.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03010004	SIL Sentence formation error. SIL sentence formation is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03010005	Filename Error. The specified filename is too long.	Specify a shorter filename. Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03010006	File does not exist error. The specified file does not exist.	Check the filename. Check if the file exists or not. Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03010007	File Structure Error. The data structure in the specified file is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03010008	File already exists error. The specified file already exists.	Check the filename. Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03010009	File does not exist error. The specified file does not exist.	Check the filename. Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0301000A	File Deletion Error. The specified file could not be deleted.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0301000B	Multiple Open Error. The specified project has been opened twice.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0301000C	Filename Error. The specified filename is incorrect.	Check the filename. Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0301000D	File Read Error. Failed in reading the file.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x0301000E	File Write Error. Failed in writing the file.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0301000F	File Seek Error. File seek failed.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03010010	File Close Error. Failed while closing the file.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03010011	Folder Creation Error. Failed while creating the folder.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03010012	File Copy Error. Failed while copying the file.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03010013	Project Path Error. The length of the project path is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03010014	Project Type Error. The project type is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03010015	File Type Error. The file type is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03010016	Sub-File Type Error. The sub-file type is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03010017	Insufficient Disk space error. The disk space is insufficient.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03020002	Multiple Open Error. Tried to open DBProduct more than once.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03020003	Not Opened error. DBProduct is not opened.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03020004	Extract Error. DBProduct is not extracted.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03020010	Parameter Error. The parameters of DBProduct are incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03020011	Language Error. The language parameter is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x03020012	Error in specifying Maker. The maker parameter is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03020013	Error in specifying Unit. The unit parameter is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03020014	SQL Parameter Error SIL, SQL Parameter of DBProduct is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03020015	SIL Sentence formation error. SIL sentence formation is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03020016	Field Key Input Error. The field key entered is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03020050	Record Data Construction Error. Failed in reconstructing the record data of DBProduct.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03020060	Error Retrieving Record Data. Failed while retrieving DBProduct record data.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03020061	Last Record error. Cannot retrieve the next record since the current record is the last record.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FF0000	Initialization error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FF0001	Not Initialized error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FF0002	Multiple Initialization error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FF0003	Workspace Initialization Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FF0004	Database Initialization Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FF0005	Recordset Initialization Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FF0006	Error Closing Database	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.



<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x03FF0007	Error Closing Recordset	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FF0008	Database Not Opened error. Database is not opened.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FF0009	Recordset Not Opened error. Recordset is not opened.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FF000A	Table Initialization Error. Failed in initializing TtableInformation table	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FF000B	Table Initialization Error. Failed in initializing TfieldInformation table	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FF000C	Table Initialization Error. Failed in initializing TrelationInformation table	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FF000D	Table Initialization Error. Failed in initializing Tlanguage table	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FF000E	Table Initialization Error. Failed in initializing Tmaker table	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FF000F	Table Initialization Error. Failed in initializing TOpenDatabase table	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FF0010	Field Value Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FF0011	Field Value Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FF0012	Exit Error. Failed to exit the database.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FF0100	Moving Record error. Failed while moving the record.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FF0101	Retreiving Record Count Error. Failed to retrieve the record count.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FF0110	Retreiving Field Value Error. Failed in retrieving the field value.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x03FF0111	Setting Field Value Error. Failed in setting the field value.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x03FFFFFF	Other Errors	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04000001	No command error. The specified CPU type cannot be used to perform processing.	Check the CPU type set to ActCpuType. Check whether the system configuration is supported or not. Exit the program and restart the computer. Reinstall MX Component.
0x04000002	Memory lock error. Failed in locking memory.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04000003	Securing Memory Error. Failed in securing the memory.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04000004	Internal server DLL load error. Start of the internal server failed.	Check for the deleted or moved installation file of MX Component. Exit the program and restart the computer. Reinstall MX Component.
0x04000005	Securing Resources Error. Failed in securing the resources.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04000006	Error Loading Main Object. Failed in reading the file.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04000007	Error Loading Conversion Table. Failed in reading table data.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04000100	Incorrect Intermediate Code Size error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04010001	Intermediate Code Not Converted error. The converted machine code for one command is more than 256 bytes.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04010002	Intermediate Code Completion Error. Intermediate code area of the code to be converted ended abruptly.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04010003	Insufficient Intermediate Code error. The intermediate code of the code to be converted was insufficient.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04010004	Intermediate Code Data Error. The intermediate code to be converted is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x04010005	Intermediate Code Structure Error. The number of steps in the intermediate code is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04010006	Error in Number of Steps. The number of steps in comment intermediate code is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04010007	Insufficient Storage Space for Machine Code error. The storage space for machine code is insufficient.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04010008	Other Errors (Other errors generated during the conversion of Intermediate code to machine code.)	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04011001	Machine Code Not Converted error. The converted intermediate code for one command is more than 256 bytes.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04011002	Machine Code Completion Error. The machine code area to be converted ended abruptly.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04011003	Abnormal Machine Code. Could not convert since the machine code to be converted was abnormal.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04011004	Insufficient Storage Space for Intermediate Code error. The storage area for intermediate code is insufficient.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04011005	Other Errors. Other errors generated while converting machine code to Intermediate code.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04020001	Text Code Not Converted error. The converted intermediate code for one command is more than 256 bytes.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04020002	No Input error. The input list code is insufficient.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04020003	Command Error. The command name of list code to be converted is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04020004	Device Error. The device name of list code to be converted is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x04020005	Device Number Error. The device number of the list code to be converted is out of range.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04020006	Conversion Error. The list code to be converted conversion could not be identified.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04020007	Text Data Error. The list code to be converted is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04020008	Error in SFC Operation Output. The output command of SFC operation is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04020009	SFC Shift Condition Error. SFC shift condition command is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0402000A	Error in Statements between lines. The statements entered between lines are incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0402000B	P.I Statement Error. The P.I statement entered is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0402000C	Note Error. The Note entered is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0402000D	Comment Error. The comment entered is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0402000E	Other Errors (Other errors generated during the conversion of list to Intermediate code)	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04021001	Intermediate Code Not Converted error. The converted list code for one command has exceeded 256 bytes.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04021002	Intermediate Code Area Full error. Intermediate code area to be converted is full.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04021003	Command Error. The command specified by the intermediate code to be converted is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04021004	Device Error. The device specified in the intermediate code to be converted is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x04021005	Intermediate Code Error. The structure of intermediate code to be converted is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04021006	Insufficient List Storage Space error. The space for storing the converted list code is insufficient.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04021007	Other Errors (Other errors generated during the conversion of intermediate code to list)	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04030001	Not Converted error. The storage space for converted intermediate code is insufficient.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04030002	Bad Circuit Creation error. The character memory circuit is not completed in a sequence.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04030003	Specified Circuit Size Exceeded. Specified circuit size is too big.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04030004	Incorrect Return Circuit error. There is no consistency before and after the return circuit. The setting for the return circuit is too high.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04030005	Other Errors (Other errors generated while converting from Character Memory to Intermediate Code)	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04031001	Not Converted error. The size (vertical/horizontal) of the character memory specified is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04031002	Abnormal Command Code error. The command intermediate code to be converted is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04031003	Bad Circuit Creation error. Could not be converted to Sequence Circuit. There is no END command.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04031004	Specified Circuit Size exceeded error. Specified circuit size is too big.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04031005	Fatal Error. Fatal Error has occurred.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x04031006	Insufficient number of storage blocks error. The space to store the converted character memory circuit blocks is not sufficient.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04031007	Circuit Block Search Error. Data is broken off in the circuit block.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04031008	Other Errors (Other errors generated during the conversion of intermediate code to character memory)	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04040001	CAD Data Error. There is no CAD data to be converted. The CAD data format is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04040002	Output Data Error. The input CAD data type and the output CAD data type are not matching.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04040003	Library Load Error. Failed to load the library.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04040004	Storage Space Secure Error. The space secured to store the converted data is not sufficient.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04040005	No END Command error. There is no END command in the CAD data to be converted.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04040006	Abnormal Command Code. There is abnormal command code in the CAD data to be converted.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04040007	Device No. Error. The device number is out of range.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04040008	Step No. Error. The step number is out of range.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04040009	The specified circuit size exceeded error. 1 circuit block is too big.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0404000A	Return Circuit Error. The return circuit is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0404000B	Bad Circuit Creation error. The circuit data is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x0404000C	SFC Data Error. The SFC data to be converted is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0404000D	List Data Error. The list data to be converted is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0404000E	Comment Data Error. The comment data to be converted is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0404000F	Statement Error. The statement data to be converted is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04040010	Other Errors (Other errors generated during the conversion of CAD code to Intermediate code.)	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04041001	Intermediate Code Data Error. There is no intermediate code to be converted. The format of the intermediate code is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04041002	CAD Data Type Error. The input CAD data type and the output CAD data type are not matching.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04041003	Library Error. Failed to load the library.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04041004	Insufficient Input Data error. Data to be converted is insufficient.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04041005	Insufficient Storage Space error. There is not enough space to store the CAD data to be converted.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04041006	No END Command error. There is no END command in the CAD data to be converted.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04041007	Abnormal Command Code. There is abnormal command code in the CAD data to be converted.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04041008	Device No. Error. The device number is out of range.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04041009	Step No. Error. The step number is out of range.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x0404100A	The specified circuit size exceeded error. 1 circuit block is too big.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0404100B	Return Circuit Error. The return circuit is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0404100C	Bad Circuit Creation error. The circuit data is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0404100D	SFC Data Error. The SFC data to be converted is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0404100E	List Data Error. The list data to be converted is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0404100F	Comment Data Error. The comment data to be converted is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04041010	Statement Error. The statement data to be converted is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04041011	Other Errors (Other errors generated during the conversion of Intermediate code to CAD code.)	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x040A0001	Insufficient Intermediate Code Storage Space. The space to store the data after conversion is insufficient.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x040A0002	The space to store addition SFC information is not sufficient	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x040A0003	Conversion Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x040A0004	Non-SFC Program Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x040A1001	Step Not Used / No Output error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x040A1002	Step No out of range error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.



<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x040A1003	Step Not Used / No Output error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x040A1004	Transition No out of range.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x040A1005	Maximum Number Exceeded error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x040A1006	Microcontroller Program space Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x040A1007	Non-SFC Program Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x040B0001	Insufficient Intermediate Code Storage Space. The space to store the data after conversion is insufficient.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x040B0002	Conversion Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x040B1001	Failed in creating Step Start position table	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x040B1002	Error Reading Step Information	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x040B1003	Step No. Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x040B1004	Failed in reading the output of operation/Transition condition intermediate code error.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x040B1005	Securing Internal Work Area Failed error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x040B1006	Error in setting the maximum value of X direction for character memory	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x040B1007	Insufficient Internal Work Area error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x040B1008	Stack Overflow, Abnormal Character Memory	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x040B1009	Insufficient No of Storage Blocks error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x040B100A	Non-SFC Program Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04050001	Abnormal Character String Specified error. Device character string specified is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04050002	Device Points Error. Device points are out of range	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04050003	Other Errors (The errors generated during the conversion of the Device Character String to Device Intermediate Code)	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04051001	Device Name Error. The classification specified for the device intermediate code is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04051002	Device Name Error. The classification specified for the extended specification device intermediate code is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04051003	Other Errors (The errors generated during the conversion of the Device Intermediate Code to Device Character String)	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04052001	Abnormal Character String Specified error. Device character string specified is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04052002	Device Points Error. Device points are out of range.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04052003	Other Errors (The errors generated during the conversion of the Device Character String to Device Representation Code)	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04053001	Device Representation Error. The classification specified for the device intermediate code is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x04053002	Device Representation Error. The classification specified for the extended specification device intermediate code is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04053003	Device Representation Error. The rectification part specified for the device is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04053004	Device Representation Error. The rectification part specified for the extended device is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04053005	Other Errors (The errors generated during the conversion of the Device Representation Code to Device Character String)	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04064001	Abnormal Device Intermediate Code error. The intermediate code for the device is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04064002	Other Errors (Other errors generated during the conversion of the Intermediate code for the Device to Device Name)	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04065001	Abnormal Device Name error. The classification specified for the intermediate code of the device is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04065002	Abnormal Device Name error. The classification for the intermediate code of the extended specification device is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04065003	Other Errors (Other errors generated during the conversion of the device name to Intermediate code)	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04066001	Device Intermediate Code Error. The intermediate code for the device is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04066002	Other Errors (Other errors generated during the conversion of the device intermediate code to device representation code.)	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04067001	Device Representation Error. The classification specified for the intermediate code of the device is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04067002	Device Representation Error. The classification for the intermediate code of the extended specification device is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x04067003	Device Representation Error. The rectification part specified for the device is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04067004	Device Representation Error. The rectification part specified for the extended device is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04067005	Other Errors (Other errors generated during the conversion of device representation code to the device intermediate code)	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04070001	Common Data Conversion Error. The input data of the device comment conversion is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04070002	Insufficient Common Data. The data to be converted is insufficient.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04070003	Insufficient Storage Area. The area where the conversion data is stored is insufficient.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04071001	Error in PLC Data Conversion. The input data of the device comment conversion is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04071002	Insufficient PLC Data error. The data to be converted is insufficient.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04071003	Insufficient Storage Area. The area where the conversion data is stored is insufficient.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04072001	Open Error. Failed in creating conversion object	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04072002	PLC Type Error. The specified PLC type does not exist.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04072003	Not Converted error. Converted object does not exist	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04072004	Input Data Error. The input data is incorrect	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04073001	Program Common Data Conversion Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x04073002	Program Common Data Conversion Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04073101	Program PLC Data Conversion Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074001	Common Data Parameter Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074002	Network Parameter Common Data Error. The parameter block exists, but the data inside is not set.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074101	Parameter PLC Data Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074102	Network Parameter PLC Data Error. The parameter block exists, but the data inside is not set.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074103	Offset Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074201	Error in Specifying Network Type. The PLC specified does not support the network type.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074202	Parameter Block Number Error. The Block corresponding to the parameter block number specified does not exist.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074203	Parameter Block Content Error. It is different from the content supported by the specified.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074204	Parameter Block Information Error. The specified block number does not exist.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074205	Default Parameter Block is Abnormal. The specified block number does not exist.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074301	Error in Conversion of the Common Parameter Block	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074302	Error in Common Parameter Block No. 1001. The value of the RUN-PAUSE settings existence flag is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x04074303	Error in Common Parameter Block No. 1003	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074304	Error in Common Parameter Block No. 1008	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074305	Error in Common Parameter Block No. 1100	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074306	Error in Common Parameter Block No. 2001 The device intermediate code specified does not exist.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074307	Error in Common Parameter Block No. 3000	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074308	Error in Common Parameter Block No. 3002	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074309	Error in Common Parameter Block No. 3004. The settings for the annunciator display mode is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407430A	Error in Common Parameter Block No. 4000. I/O Allotment Data is not created.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407430B	Error in Common Parameter Block No. 5000. The specified network is not supported.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407430C	Error in Common Parameter Block No. 5001. Valid unit No is not set while accessing other exchange.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407430D	Error in Common Parameter Block No. 5002	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407430E	Error in Common Parameter Block No. 5003	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407430F	Error in Common Parameter Block No. 5NM0	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074310	Error in Common Parameter Block No. 5NM1	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x04074311	Error in Common Parameter Block No. 5NM2	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074312	Error in Common Parameter Block No. 5NM3	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074313	Error in Common Parameter Block No. 6000	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074314	Error in Common Parameter Block No. FF18. Link parameter Capacity is not set.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074315	Error in Common Parameter Block No. FF25. Calculation circuit check is not set.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074316	Error in Common Parameter Block No. FF30. Sampling Trace Data is not created.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074317	Error in Common Parameter Block No. FF31. Status latch data is not created.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074318	Error in Common Parameter Block No. FF42. Timer processing points are not set.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074319	Error in Common Parameter Block No. FF30. Setting value device for specified extended timer does not exist.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407431A	Error in Common Parameter Block No. FF44	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407431B	Error in Common Parameter Block No. FF45	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407431C	Error in Common Parameter Block No. FF60. Terminal Settings are not set.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407431D	Error in Common Parameter Block No. FF70. User Release area is not set.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074401	Error in Conversion of PLC Parameter Block	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x04074402	Error in PLC Parameter Block No.1001	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074403	Error in PLC Parameter Block No.1003	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074404	Error in PLC Parameter Block No.1008	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074405	Error in PLC Parameter Block No.1100	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074406	Error in PLC Parameter Block No.2001	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074407	Error in PLC Parameter Block No.3000	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074408	Error in PLC Parameter Block No.3002	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074409	Error in PLC Parameter Block No.3004	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407440A	Error in PLC Parameter Block No.4000	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407440B	Error in PLC Parameter Block No.5000. The specified network type is not supported.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407440C	Error in PLC Parameter Block No.5001	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407440D	Error in PLC Parameter Block No.5002	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407440E	Error in PLC Parameter Block No.5003	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407440F	Error in PLC Parameter Block No.5NM0. The specified network type is not supported.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074410	Error in PLC Parameter Block No.5NM1	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.



<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x04074411	Error in PLC Parameter Block No. 5NM2. The specified network type is not supported.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074412	Error in PLC Parameter Block No. 5NM3	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074413	Error in PLC Parameter Block No. 6000	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074414	Error in PLC Parameter Block No. FF18	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074415	Error in PLC Parameter Block No. FF25	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074416	Error in PLC Parameter Block No. FF30	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074417	Error in PLC Parameter Block No. FF31	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074418	Error in PLC Parameter Block No. FF42	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04074419	Error in PLC Parameter Block No. FF43	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407441A	Error in PLC Parameter Block No. FF44	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407441B	Error in PLC Parameter Block No. FF45	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407441C	Error in PLC Parameter Block No. FF60	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407441D	Error in PLC Parameter Block No. FF70	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04075001	Common Data Conversion Error. Failed while converting the device memory settings portion.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04075002	Common Data Conversion Error. Failed while converting the device memory data portion.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x04075003	Common Data Conversion Error. Device memory data portion did not exist.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04075101	PLC Data Conversion Error. Failed while converting the settings portion of the device memory.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04075102	PLC Data Conversion Error. Failed while converting the data portion of the device memory.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04076001	Common Data Conversion Error. Failed while converting the settings portion of the device comments.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04076002	Common Data Conversion Error. Failed while converting the data portion of the device comments.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04076101	PLC Data Conversion Error. Failed while converting the settings portion of the device comments.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04076102	PLC Data Conversion Error. Failed while converting the settings portion of the device comments.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04077001	Common Data Conversion Error. Failed during the conversion of sampling trace settings portion.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04077002	Common Data Conversion Error. Failed during the conversion of sampling trace data portion.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04077101	PLC Data Conversion Error. Failed during the conversion of sampling trace settings portion.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04077102	PLC Data Conversion Error. Failed during the conversion of sampling trace data portion.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04078001	Common Data Conversion Error. Failed in the conversion of the status latch settings portion.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04078002	Common Data Conversion Error. Failed in the conversion of the status latch data portion.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04078101	PLC Data Conversion Error. Failed in the conversion of the status latch settings portion.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x04078102	PLC Data Conversion Error. Failed in the conversion of the status latch data portion.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04079101	Failure history PLC Data Conversion error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407A101	File List PLC Data Conversion Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407B101	Error Information PLC Data Conversion Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407C001	Error in Conversion of Indirect Address to Device Name. The device name storage area is not secured.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407C002	Error in Conversion of Device Name to Indirect Address. Indirect Address storage area is not secured.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407C003	Error in Conversion of Indirect Address to Device Representation. The device representation storage area is not secured.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407C004	Error in Conversion of Device Representation to Indirect Address. Indirect Address storage area is not secured.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407C005	Error in Conversion of Indirect Address to Device Character String. Device Character String storage area is not secured.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407C006	Error in Conversion of Device Character String to Indirect Address. Indirect Address storage area is not secured.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407C007	Error in Conversion of Intermediate Code to Device Name. Device Name storage area is not secured.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407C008	Error in Conversion of Device Name to Intermediate Code. Intermediate Code storage area is not secured.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x0407C009	Error in Conversion of Intermediate Code to Device representation. Device Representation storage area is not secured.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407C00A	Error in Conversion of Device Representation to Intermediate Code. Intermediate Code storage area is not secured.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407C00B	Error in Conversion of Intermediate Code to Indirect Address. Indirect Address storage area is not secured.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407C00C	Error in Conversion of Indirect Address to Intermediate Code. Intermediate Code storage area is not secured.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407C00D	PLC Type Error The specified PLC type is not supported.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407C00E	Device Character String Error. The specified device is not supported.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407C00F	Device Character String Error. The specified device character string, type is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407C010	Device Error. The specified device is not supported by the specified PLC	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407C011	PLC Type Error. The specified PLC is not supported.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407C012	Device out of Range Error. For AnA system, a device out of AnA system range was specified.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407D001	Common Data Conversion Error. Error in Conversion of SFC trace condition settings portion.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407D002	Common Data Conversion Error. Error in Conversion of SFC trace condition data portion.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x0407D101	PLC Data Conversion Error. Error in Conversion of SFC trace condition settings portion.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x0407D102	PLC Data Conversion Error. Error in Conversion of SFC trace condition data portion.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04080001	Intermediate Code classification out of range error. The intermediate code classification specified is out of range.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04080002	Extended specification Intermediate Code classification out of range error. The extended specification intermediate code specified is out of range.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04080003	Device Points check absent error. The device does not check the device points.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04090001	GPP Project Error. The specified PLC type and GPP project type are not matching.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04090002	File Type Error. The specified GPP project type and file type are not matching.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04090010	Insufficient GPP Data to be converted. There is no data to be converted. The data size specified is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04090011	Insufficient Storage Space for Converted Data. The space for storing converted data is insufficient.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04090012	Error in GPP Data to be converted. The GPP data to be converted is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04090110	Insufficient Data to be converted error. There is no data to be converted. The data size specified is insufficient.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04090111	Insufficient Storage Space for Converted Data error. The storage space for converted data is insufficient.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04090112	Error in data to be converted. The data to be converted is incorrect.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x04FFFFFF	Other Errors	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x10000001	No Command error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000002	Start of communication DLL of MX Component failed.	Exit the program and restart the computer. Reinstall MX Component.
0x10000003	Open failed. (DiskDrive)	Exit the program and restart the computer. Reinstall MX Component.
0x10000004	Duplex open error	Exit the program and restart the computer.
0x10000005	File Access Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000006	Incorrect Folder Name error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000007	File Access Denied error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000008	Disk Full Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000009	File Delete Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x1000000A	Incorrect File Name error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x1000000C	Execution failed since another application or thread is making a request.	Execute again after some time. Perform programming according to the multithread rules of COM and ActiveX. Exit the program and restart the computer.
0x1000000D	Folder Creation Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x1000000E	Folder/ File Type Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x1000000F	Offset Address Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000010	Request Cancel. Cancel Process has occurred.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000011	Memory securing error	Exit the program and restart the computer. Reinstall MX Component.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x10000012	Open not yet executed	Exit the program and restart the computer.
0x10000013	Attach Not Executed error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000014	Object Invalid error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000015	Request Cancel Failed error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000016	Failed in Reading Status error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000017	The specified size (number of devices) is unauthorised.	Check the number of points specified in the method. Exit the program and restart the computer.
0x10000018	There is no registered device.	Exit the program and restart the computer.
0x10000019	Dataset Not Executed	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x1000001A	Read Not Executed error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x1000001B	Incorrect Create Flag error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x1000001C	Operation Over Access	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x1000001D	Redundant Device error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x1000001E	Registry search failed.	Exit the program and restart the computer. Exit other programs and secure free memory area. Reinstall MX Component.
0x1000001F	File Type Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000020	Device Memory Type Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000021	Program Range Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x10000022	TEL Type Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000023	TEL Access Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000024	Cancel Flag Type Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000030	Multiple Device Registration Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000031	Device Not Registered error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000032	Specified device error	Review the specified device data. Exit the program and restart the computer. Exit other programs and secure free memory area.
0x10000033	Specified device range error	Review the specified device data. Exit the program and restart the computer. Exit other programs and secure free memory area.
0x10000034	File Write Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000040	Server start failed.	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000041	Server Stop Error. Failed while stopping the server	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000042	Server Started Twice error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000043	Server Not Started error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000044	Resource Timeout Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000045	Server Type Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000046	Failed to Access Server error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.



<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0x10000047	Server Already Accessed error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000048	Failed in Simulator Startup	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x10000049	Failed in exiting Simulator	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x1000004A	Simulator Not Started error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x1000004B	Simulator Type Error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x1000004C	Simulator Not Supported error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x1000004D	Simulator Started Twice error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0x1000004E	Shared Memory Not Started error	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0xF0000001	No-license error. The license is not given to the IBM-PC/AT compatible.	Using the license FD, give the license to the IBM-PC/AT compatible.
0xF0000002	Set data read error. Reading of the set data of the logical station number failed.	Specify the correct logical station number. Set the logical station number on the communication settings utility.
0xF0000003	Already open error. The Open method was executed in the open status.	When changing the communication target CPU, execute the Open method after performing Close.
0xF0000004	Not yet open error. The Open method is not yet executed.	After executing the Open method, execute the corresponding method.
0xF0000005	Initialisation error. Initialisation of the object possessed internally in MX Component failed.	Exit the program and restart the computer. Reinstall MX Component.
0xF0000006	Memory securing error. Securing of MX Component internal memory failed.	Exit the program and restart the computer. Exit other programs and secure free memory area.
0xF0000007	Function non-support error. The method does not support.	Can not use because the corresponding method is not supported.

Error code	Error Definition	Action
0xF1000001	Character code conversion error. Character code conversion (UNICODE ASCII code or ASCII code UNICODE) failed.	Check the character string specified in the method. The ASCII character string acquired from the PLC CPU is abnormal. Review the system, e.g. PLC CPU, module setting and cable status. Exit the program and restart the computer. Retry the GetCpuType method.
0xF1000002	First I/O number error. The first I/O number specified is an unauthorised value. A matching first I/O number does not exist.	Check the value of the first I/O number specified in the method. Using the GPP function, check the PLC CPU parameters (I/O assignment).
0xF1000003	Buffer address error. The buffer address specified is an unauthorised value. The buffer address is outside the range.	Check the value of the buffer address specified in the method.
0xF1000004	Buffer read size error. As a result of buffer read, the specified size could not be acquired.	Perform reopen processing. Review the system, e.g. PLC CPU, module setting and cable status. Retry. Exit the program.
0xF1000005	Size error. The size specified in the read/write method is abnormal. The read/write first number plus size exceeds the device or buffer area.	Check the size specified in the method.
0xF1000006	Operation error. The operation specified for remote operation is an abnormal value.	Check the operation specifying value specified in the method.
0xF1000007	Clock data error. The clock data is abnormal.	Check the clock data specified in the method. Set the correct clock data to the clock data of the PLC CPU.
0xF1000008	Monitored device registration count excess. The number of device points registered in the EntryDeviceStatus method was 0 or less. The number of device points registered in the EntryDeviceStatus method was more than 20.	Register the device points between 1 and 20 in the EntryDeviceStatus method.
0xF1000009	Monitored device data registration error	After making deregistration in the FreeDeviceStatus method, execute the EntryDeviceStatus method again.
0xF1000010	Device status monitor processing failed to start. Device status monitor processing failed to end.	Start/end the device status monitor processing again in the EntryDeviceStatus method.
0xF1000011	The VARIANT argument data type is wrong.	Reexamine the data type specified for the VARIANT argument. <ul style="list-style-type: none"> <li>• Check whether the array variable size is large enough.</li> <li>• Check whether the data type specified in the corresponding method has been set.</li> </ul>

Error code	Error Definition	Action
0xF1000012	The device status monitoring time interval is a value outside the range 1 second to 1 hour (1 to 3600).	Specify the device status monitoring time between 1 and 3600.
0xF1000013	Already Connected error. Connect was executed again after it was executed for the same object.	Execute the Connect method after executing the Disconnect method.
0xF1000014	Invalid Telephone Number error. Characters other than "0123456789-*#" that are allowed for telephone numbers are included.	Rectify the Telephone number and try to Connect again.
0xF1000015	Exclusive Control Failure error. There was failure in the exclusive control process while executing the Connect and Disconnect method.	In case if Connect/Disconnect method is being executed for any other object, execute the failed method (Connect/Disconnect) again after the completion of the Connect/Disconnect method of that object. If the Connect/Disconnect process is in progress only for the self object, perform the following. Exit the program. Restart the computer. Reinstall MX Component.
0xF1000016	While connecting to the telephone line error. The telephone line is connected to some other application, other than the one using MXComponent.	Try Connecting again after disconnecting the application that is using the telephone line.
0xF1000017	Telephone line not connected error. Telephone line is not connected. Connect was executed and the telephone line was connected, but it got disconnected due to some reason.	(When Connect method has failed) Execute Connect again after executing Disconnect method. (When method other than Connect has failed) Execute Disconnect method, Execute Connect and connect to the telephone line. After connecting, execute the method that failed once again.
0xF1000018	No Telephone number error. The telephone No. is not set. The telephone No. or call back No. is not set, if the connection method is Automatic (when specifying the call back No.), call back connection (when specifying the number), or call back Request (when specifying the number).	In case of program settings type, set the telephone No. to the property ActDialNumber. (Set the telephone No. to the properties ActDialNumber and ActCallbackNumber, if the connection method is automatic (when specifying the call back No.), call back connection (when specifying the telephone No.), or call back request (when specifying the number).) In case of utility settings type, set the telephone No. using the wizard. (Set the telephone No. and call back No. , if the connection method is automatic (when specifying the call back No.), call back connection (when specifying the telephone No.), or call back request (when specifying the number).)
0xF1000019	Not Closed error. Disconnect was executed while in Open state.	Try Disconnect again after executing Close.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0xF100001A	Target telephone line connection mismatch error. Connect was tried for a different telephone number using the port which is already connected to a telephone line. (When the method of connection is a callback reception, it is considered that the telephone number is different from methods of connection in other than the callback reception.)	If you want to connect to a different telephone number, Execute Disconnect with respect to the telephone line that is already connected and executes Connect after it gets disconnected. In case of connecting the telephone line with callback reception, use the Connect of the connection method that is executed at the earliest in the same port as callback reception.
0xF100001B	Control Type Mismatch error. An object, whose control type is different from that of the object already connected to the telephone line, tried to Connect.	Execute Disconnect for the object currently connected to the telephone line and execute Connect once again after the telephone line gets disconnected.
0xF100001C	Not Disconnected error. When Disconnect method is executed for the object connected to the telephone line, it is found that other objects are in connected state.	Execute Disconnect for all the Connected objects. Try Disconnect again for the object that actually performed the telephone line connection.
0xF100001D	Not Connected error. Open was executed before Connect Or, Disconnect was executed.	Execute Open again after executing Connect. Or execute Disconnect again after executing Connect.
0xF100001E	Fatal Error.	Exit the program. Restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0xF100001F	Open time setting error. There is some difference in telephone number and the port number settings used during Connect and Open. There is some error in Connect way.	Check the telephone number and the port number. Check the Connect way.
0xF2000002	There is an error response from the target telephone. Normally caused by a communication error.	Check the value of the properties set in case of program settings type and check the contents of the detailed settings that were set using the wizard in case of utility settings type.
0xF2000003	Invalid data was received. Causes can include an incorrect data packet received due to noise, or communicating with a device other than A(Q)6TEL/C24.	Retry. Check the communication device used at the other end.
0xF2000004	There is no response from the modem. Causes can include abnormality in the modem, or the wrong telephone number.	Check the status of the modem. Check the telephone number. If the problem persists even after checking the above points, change the value of the properties set (Properties such as ActConnectionCDWaitTime etc. , which set the timings) in case of program settings type and change the contents of the detailed settings that were set using the wizard in case of utility settings type.
0xF2000005	There are chances that the line is not disconnected.	Check the line.

<b>Error code</b>	<b>Error Definition</b>	<b>Action</b>
0xF2000006	The PC modem did not receive the AT command. Causes can include specifying an invalid AT command, or an abnormality in the modem.	Check the contents of the AT command. Check the status of the modem.
0xF2000007	Modem did not respond properly to the standard escape command.	Check the modem. Confirm whether the value of the time-out is too small. ( 5000ms or more is recommended. )
0xF2000009	Modem does not respond properly to the line Disconnect command.	Check the modem.
0xF200000A	Target did not receive the signal. The Receive settings of the modem at the other end may be incorrect, the other end may be busy or the telephone number may be incorrect.	Check the Receive settings of the modem at the other end. Check if the other end is busy. Check the telephone number.
0xF200000B	Timeout reached for the call back receive waiting time.	Increase the call back receive waiting time ActCallbackReceptionWaitingTimeOut and execute connect again.
0xF200000C	Password of A6TEL, Q6TEL, QJ71C24 units could not be resolved.	Set the password to ActPassword property and execute the failed method again.
0xF2010001	The callback line disconnect wait time is other than 0 -180 Seconds. The callback execution delay time is other than 0 -1800 Seconds. The telephone number is more than 62 characters.	Check whether the callback line disconnect wait time is with in 0 – 180 Seconds. Check whether the callback execution delay time is with in 0 - 1800 Seconds. Check whether the telephone number is less than or equal to 62 characters. Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
0xF2010002	QJ71C24 did not receive the specified connection method. Causes can include an incorrect Connection method or an incorrect telephone number for Call back.	Check whether the settings of QJ71C24 and the MXComponent are matching.
0xF2010003	QJ71C24 does not permit the automatic connection (during fixed Call back or when the number is specified.)	Check the settings of QJ71C24.
0xF2100005	There are chances that the line is not disconnected.	If there is no problem with the modem or the telephone line, change the value of the properties set (Properties like ActConnectionCDWaitTime etc. , which set the timings) in case of program settings type and change the contents of the detailed settings that were set using the wizard in case of utility settings type.
0xF2100008	There was no response from the modem for the data sent from the PC.	Change the value of the properties set (Properties such as ActConnectionCDWaitTime etc. , which set the timings) in case of program settings type and change the contents of the detailed settings that were set using the wizard in case of utility settings type.

Error code	Error Definition	Action
0xF2100006	Modem did not receive the startup command AT.	Change the settings of the property ActATCommand. in case of program settings type and change the command AT that were set using the wizard in case of utility settings type.
0xF2100007	The PC modem does not respond to the Escape command.	If there is no problem with the modem or the telephone line, change the value of the properties set (Properties like ActConnectionCDWaitTime etc., which set the timings) in case of program settings type and change the contents of the detailed settings that were set using the wizard in case of utility settings type.
0xF21000**	There is no response from the modem. Causes can be the following. Abnormality in the modem. Wrong telephone number.	Check the status of the modem. Check the telephone number. If the problem persists even after checking the above points, change the value of the properties set (Properties such as ActConnectionCDWaitTime etc., which set the timings) in case of program settings type and change the contents of the detailed settings that were set using the wizard in case of utility settings type.
0xF21001**	There is no response from A(Q)6TEL/C24. Causes can be the following. Setting mistake w.r.t. A(Q)6TEL/C24 A(Q)6TEL/C24 got connected to a non-existent modem.	Re-examine the settings of A(Q)6TEL/C24. Confirm whether the modem exists. If the problem persists even after checking the above points, change the value of the properties set (Properties such as ActConnectionCDWaitTime etc., which set the timings) in case of program settings type and change the contents of the detailed settings that were set using the wizard in case of utility settings type.
0xF202****	There was a communication failure. Following causes can be considered depending on the status. Communication time over (Break in cable, the specified port not supported, a mistake in specifying the COM port) Modem's power is switched OFF.	Check whether the cable is broken. Check whether the specified port is not supported. Check whether correct COM port is set. Check if the modem power is switched OFF. For detailed troubleshooting, please refer to the details of the Error Code got after replacing the first four digits with "0x0180". e.g. In case of "0xF202480B", please refer to the code "0x0180480B".

### 13.3 CPU, module and network board errors

In addition to the error codes in the previous sections, there are additional error codes relating to specific hardware modules. These error codes are separated into two parts; the upper two bytes of the error number give the module where the error was reported, and the lower two bytes give the error number within that module. These error numbers can then be found in the manual of the device that reported the error.

Error Code	Module which reported the error
0x01010000 to 0x0101FFFF	QCPU (A mode), ACPU, motion controller CPU [see also <a href="#">ACT error codes</a> <sup>[375]</sup> ]
0x01020000 to 0x0102FFFF	QnACPU
0x01030000 to 0x0103FFFF	C24
0x01040000 to 0x0104FFFF	QC24(N)
0x01050000 to 0x0105FFFF	E71
0x01060000 to 0x0106FFFF	QE71
0x01070000 to 0x0107FFFF	MELSECNET/H, MELSECNET/10, CC-Link or CPU board
0x01090000 to 0x0109FFFF	FXCPU
0x010A0000 to 0x010AFFFF	QCPU (Q mode)
0x010B0000 to 0x010BFFFF	Q series-compatible C24
0x010C0000 to 0x010CFFFF	Q series-compatible E71
0x010D0000 to 0x010DFFFF	PC CPU module
0x010F0000 to 0x010FFFFF	GOT

**Note:** The module number calculated from the above table may be incorrect if the device settings were incorrect, for example if the wrong CPU type is specified in the settings, the number for that (incorrect) CPU type may appear in the error number.

Another possible exception with these error codes is if the AJ71E71 or AJ71QE71 communication module is used. If the two lower bytes of the error number do not appear in the E71 or QE71 manual, this may mean that DIP switch SW2 on the front of the E71 or QE71 module is not correctly set. This switch controls whether the packet data is transmitted in ASCII or binary format, and when the wrong format is used it may not be possible to return an accurate error number.

When the driver is used to access another station (e.g. MELSECNET/H, MELSECNET/10, CC-Link, Ethernet etc.), it is also possible that the error code was returned by another module elsewhere in the network, which encountered an error while trying to relay the data. If it is not possible to find the two lower bytes of the error number in the expected manual, it may be that the error code can be found in the manual of another CPU, relayed network module or network board that is between the driver and the target device.

## 14 Appendix: DCOM Configuration

This appendix will explain the changes that need to be made to the configuration on the client and server PCs to use DCOM.

The server PC and the client PC must both be configured correctly, so that the connection works in both directions.

### 14.1 Introduction

When connecting to the MX OPC server from a client application (such as a SCADA package or a Visual Basic program), it is not normally necessary to use DCOM. Typically the server and client components will be on the same computer anyway, or there will be a better way to pass the data from the OPC server across the network, such as using an Ethernet connection to the PLC, or passing the data between computers using a SCADA package's own built-in networking. However, there are some situations where DCOM is a suitable way to pass data from the OPC server between PCs, such as reading data from a Visual Basic program on another computer when there is only a single serial link connection available to the PLC.

#### 14.1.1 Security considerations

Configuring DCOM almost always involves a compromise in the level of security on the client and server PCs, as security settings have to be 'opened up' to allow DCOM to communicate between the PCs in both directions – with the callback mechanisms, the OPC server also behaves as a client, and the OPC client also behaves as a server.

DCOM is not recommended for use to connect two computers over an internet connection; XML-DA is more suitable for this and does not have the same incompatibilities that DCOM has with the network address translation (NAT) used in most internet firewalls and routers.

Before setting up DCOM in a corporate environment, you will need to discuss the changes to the security configuration with your network administrator or IT department, to make sure that the settings are consistent with your company's security policy.

If the OPC server and client PCs are both on the same network and isolated from other PCs and the internet, you may be able to use a simpler security configuration than if the PCs are connected across a corporate network.

#### 14.1.2 Compatibility

Some of the changes that need to be made to the DCOM configuration will affect all applications using DCOM on the computer, not just the MX OPC server.

It is recommended that you make a note of the original settings before making changes, so that if another application using DCOM (such as another OPC server) stops working, the original settings can be restored.



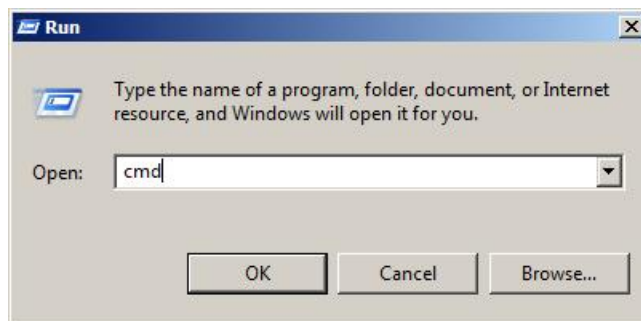
## 14.2 Before you begin

Before you start to make changes to the DCOM configuration, please ensure that:

- The client and server PCs are up to date with service packs and security fixes (e.g. service pack 4 for Windows 2000, and service pack 3 for Windows XP). There are some known problems with DCOM that are fixed by service packs, and applying the latest security fixes will reduce the risks introduced when the security configuration is changed.
- You are logged on to the PC(s) using a local administrator account
- There is a working TCP/IP network connection between the client and server PCs. One way to test this is by using the 'ping' command:
  1. Open a command prompt as follows:

### Windows XP, Windows 2000 or Windows Server 2008 R2

Select 'Start' -> 'Run', enter 'cmd' in the dialog that appears and click 'OK'.



### Windows 7, Windows Vista or Windows Server 2008 R2

Click the Windows icon or start button, then enter 'cmd' in the search box and press enter.



2. In the command window that appears, enter 'ping <otherpc>', where <otherpc> is the name of the other computer e.g. 'ping SCADAserver'. A computer's network name can normally be found by right-clicking on 'My computer', selecting 'properties' then clicking on the 'Computer name' tab. If in doubt, ask your network administrator for assistance.
  3. After pressing return, you should normally see a message such as 'Pinging <address> with 32 bytes of data' (where <address> is the IP address such of the other PC, for example 192.168.200.100) followed by four response messages from the other computer. If no address is shown, there may be a problem in finding the other computer by name – ask your network administrator to check the PCs settings. If the address is found but there are no responses, this may mean that the connection is not working, or it can sometimes indicate that a third-party firewall on one of the computers is preventing the ping request from being sent or the ping response from being received.
- If you are developing the client application yourself, make sure that you have applied the most recent service packs to your development environment. There are known issues with DCOM in un-patched versions of (for example) Visual Basic 6.

### 14.2.1 User accounts

For the highest level of security, it is safest to assign DCOM access rights to specific user accounts instead of assigning full access rights to all users.

The best way to set up security depends on your network configuration and the client software you will be using, but some typical ways in which the account security can be configured are:

Creating a new account for the OPC server connection – a single account is created and always used for the OPC connection. This is useful in situations where the OPC server is always used unattended or used by a client application that runs as a service on another computer. The MX OPC server service can be configured to run 'as' the new user.

Adding access rights for each remote user individually.

Creating a group for OPC users – this is useful when many different named users need access to the server, and has the advantage that when another user needs to be given access, they can be simply added to the group without having to go through all of the configuration steps again.

Allowing all users to access the server, by adding permissions for the group 'Everyone'. This is only advisable if you are using a separate, isolated network and if there have been problems connecting to the OPC server with the other options.

There are some other built-in accounts that may need to be given access rights depending on your application:

SYSTEM – most system services run under this account by default, unless they have been manually configured to run under a specific user account. If a system service is acting as an OPC client, you may need to give access rights to the 'SYSTEM' user.

INTERACTIVE – this account represents the currently logged-on user. Some applications may operate in a way that requires access rights to be assigned to this account.

ANONYMOUS LOGON (or 'NT AUTHORITY\ANONYMOUS LOGON' on Windows 2000) – this represents a connected user whose details are not available, which may be a result of connecting with authentication set to 'None' (described later).

Distributed COM users - this security group may be available on later operating systems to simplify assigning DCOM rights to users.

**Domain issues**

Depending on whether the client and server PCs are members of a domain, you may need to make some additional configuration changes.

Domain status	Notes
Client and server PCs are in the same Domain	When adding a new account or group for the OPC server connection, it can be added to the domain and will then be available on both PCs.
Client and server PCs are in different Domains	Each domain must be configured to trust user accounts from the other domain. You will need to contact the administrator of each domain to confirm that this change is permitted by your corporate security policy.
Neither of the computers is in a domain	Whether adding a new account for the OPC server connection, or assigning access rights to named users, there must always be an account with the same username and password set up in both PCs independently. If the account details do not match, the connection will not work, so if you change the password on one of the PCs, it must be changed on the other PC as well.
Only one of the PCs is in a domain	For this configuration to work, it may be necessary to configure DCOM on the server to allow all users full access rights. In general, this configuration should be avoided.

It is also recommended that the same operating system (e.g. Windows 2000, Windows XP or later) is used on both computers.

### 14.3 'Express' DCOM configuration on Windows XP Service Pack 2

This topic explains the simplest possible configuration of DCOM on Windows XP Service Pack 2 (or later). This assumes that you have a standard installation of Windows XP and will be assigning access rights to a single named user account or group, without any requirement for tight security. If you require more thorough security, your PC(s) have a more complex configuration, or if DCOM does not work first time after configuration, you will need to follow the full instructions in the section ['Configuring DCOM on Windows XP Service Pack 2'](#)<sup>[454]</sup>.

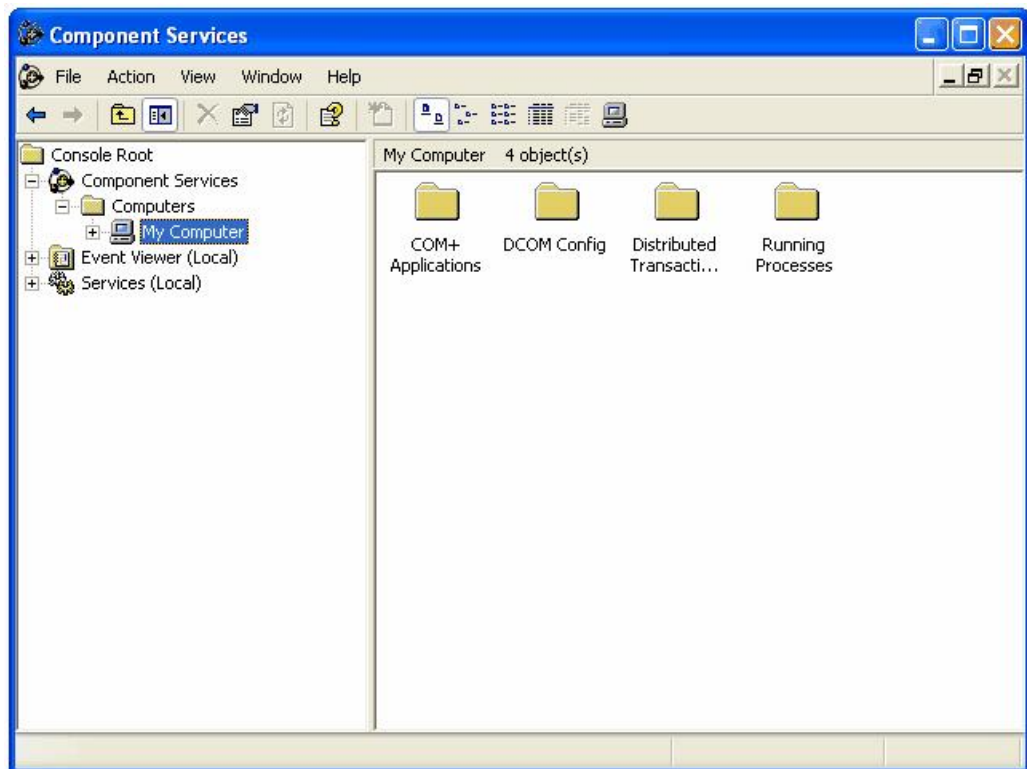
First, configure the Windows firewall for DCOM by following the instructions in ['Configuring the Windows Firewall'](#)<sup>[455]</sup>. This will allow the applications using DCOM to access the network.

Next, the DCOM settings must be configured. From the 'Start' menu, select 'Run' to show the window below.

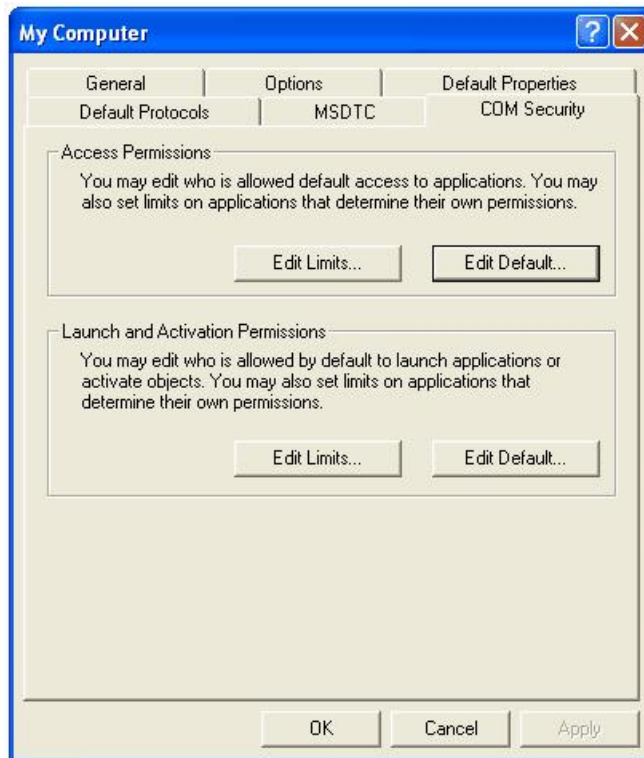


Enter 'dcomcnfg' and click 'OK' to start the DCOM configuration utility.

In the first window that appears (see example below), click on the 'Component services' and 'Computers' items in the left hand pane to expand them, so that 'My Computer' is shown.

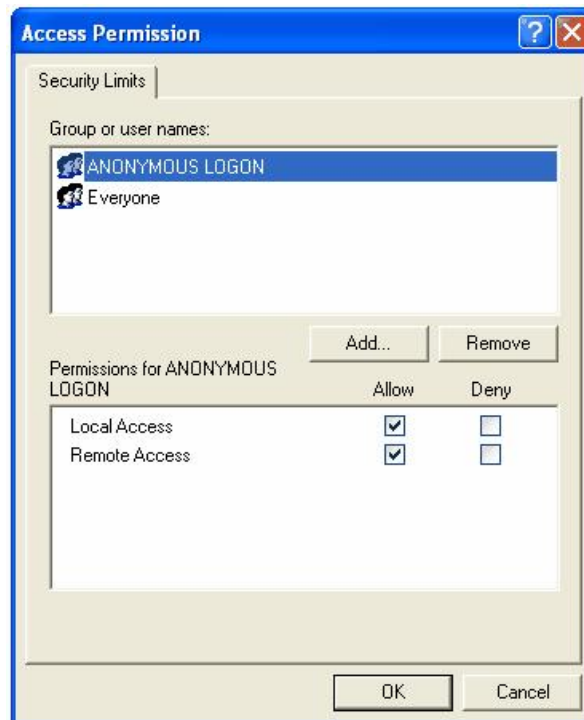


Right click on 'My computer' and select 'Properties' from the pop-up menu. In the next window, click on the 'COM security' tab to show the dialog below:



The 'Limits' on this page will override the COM security settings for individual components, and from XP service pack 2 onwards the default settings are too restrictive to allow OPC communications to work. The restrictions must be relaxed before using DCOM to access an OPC server remotely.

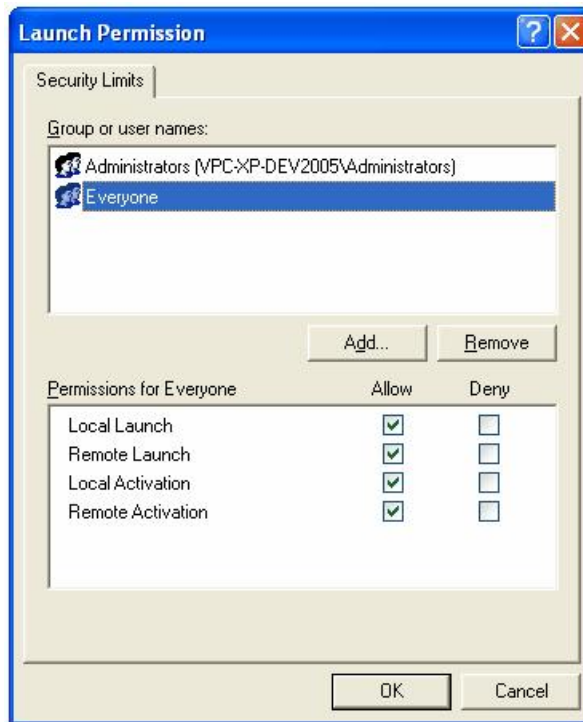
In the 'Access permissions' section, click 'Edit limits' to display the window below.



If 'ANONYMOUS LOGON' does not already appear in the list, add it. Make sure that the 'Remote Access' / 'Allow' checkbox is selected.

Use the 'Add' button to add your named user or group, and make sure the 'Allow' checkbox is set for both 'Local access' and 'Remote access'.

Click 'OK' to save the changes, then on the 'COM security' screen click the 'Edit limits' button in the 'Launch and Activation Permissions' section to show the window below:



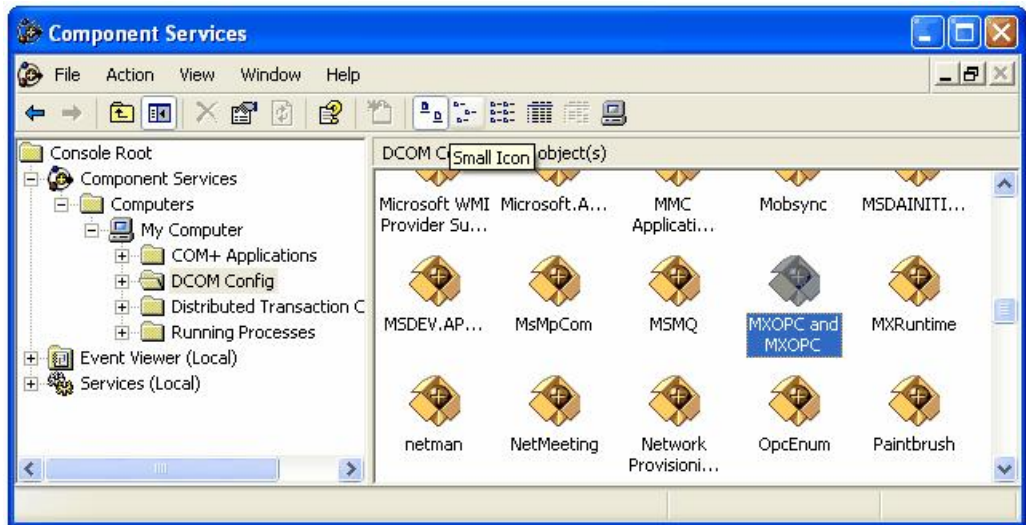
Use the 'Add user' button to add your named user or group, and ensure that all the 'Allow' checkboxes are set.

Click 'OK' to return to the 'COM security' window. The 'Edit defaults' settings for 'Access permissions' and 'Launch permissions' on this page are used to set the default permissions for COM applications. An application can either use the default settings, or can set customized security permissions (described later). These dialogs have the same format as the 'Edit limits' screens above.

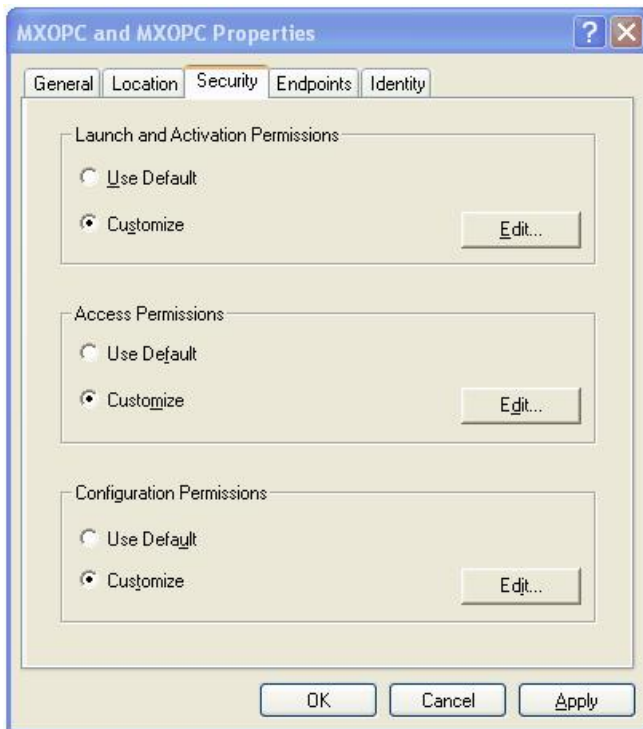
In the 'Edit defaults' setting for 'Access permissions', add your named users or group, and ensure that the 'Allow' box is checked for both local and remote access.

Repeat this step for the 'Edit defaults' setting for 'Launch and activation permissions', ensuring that all of the 'allow' check boxes are set.

When you have finished, click 'OK' to return to the 'Component Services' dialog. For the server PC configuration (only), you will need to configure the security on the MX OPC server, MX Runtime and OPC enumeration services. To do this, double-click on the 'DCOM config' entry, and find 'MXOPC', 'MXRuntime' and 'OpcEnum' in the list, as shown in the picture below.



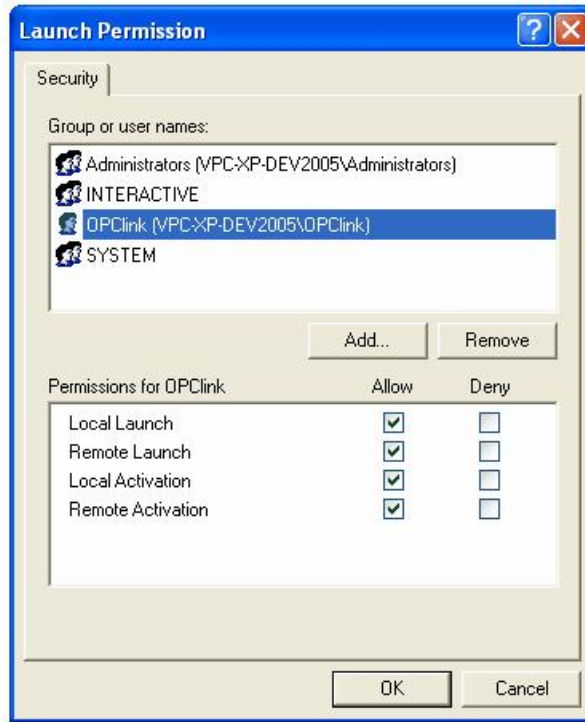
This step will need to be repeated for 'MXOPC', 'MXRuntime' and 'OpcEnum' in turn – 'MXOPC' will be used for this example. Right click on the 'MXOPC' icon and select 'properties', then switch to the 'Security tab'.



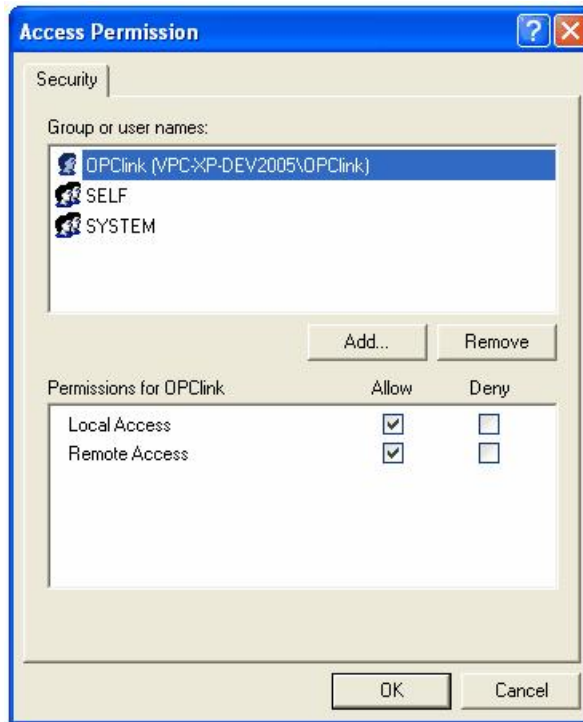
The 'Launch and activation permissions' and 'Access permissions' can both be left as 'Default' to use the default security settings configured earlier. Alternatively, they can be customized to allow more control over the users that can start and use each server. To do this, select 'customise' for each option and select 'Edit'.



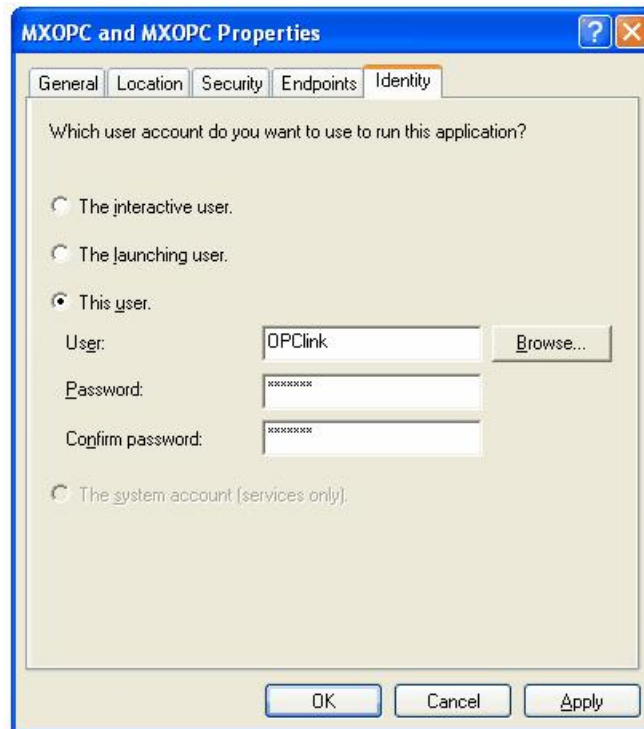
A sample 'Launch and activation permissions' edit window is shown below, in which a named user 'OPClink' has been added. As with the 'edit defaults' settings, all four 'allow' check boxes must be set for users requiring access through DCOM.



The 'Access permissions' configuration screen is shown below, with a named user 'OPClink' added.



After changing the settings on the security tab (if required), select the 'Identity' tab. In the sample screen below, the server has been configured to run as a named user.



The options on this screen are:

- **The interactive user** – the user who is currently logged on to the machine. This can cause problems with DCOM, as there is no guarantee of which user (if any) will be logged on when the server is accessed from another computer.
- **The launching user** – the user who accessed the server, causing it to be started.
- **A named user** – the server runs as a specified user, whose name and password are provided. This option allows the server to be configured independently of the user that started the server and the user that is logged on, and works well for unattended computers.
- **The system account**, when the server is configured to run as a service.

For a simple configuration, 'named user' is recommended. After setting the option, press 'OK' to return to the 'Component settings' page, which can then be closed.

If the OPC server is installed as a service, you may also wish to set up the service to run as a named user - see section '[Configuring services to use a specific user account](#)'<sup>[471]</sup> for instructions on how to do this.

## 14.4 Configuring DCOM on Windows Vista or later

This section contains instructions specific to Windows 7, Windows Server 2008 R2 and Windows Vista. The configuration for these operating systems is very similar, although there may be slight differences in the appearance of the dialogs.

Most of the changes in this section must be made on both the server PC and the client PC, so that the connection works in both directions.

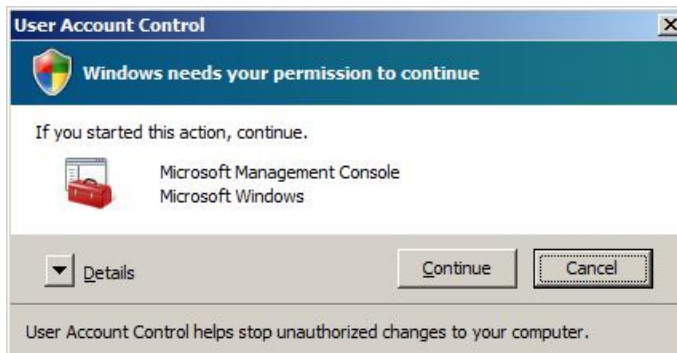
### 14.4.1 Configuring the Windows firewall

If you are using the Windows firewall, you will need to configure it to allow DCOM network traffic to pass through. If you are using a third party software firewall, please refer to the manual to find out how to make the equivalent configuration changes. If DCOM does not work when your third-party firewall is enabled, but works when it is switched off, the problem is likely to be with the firewall configuration.

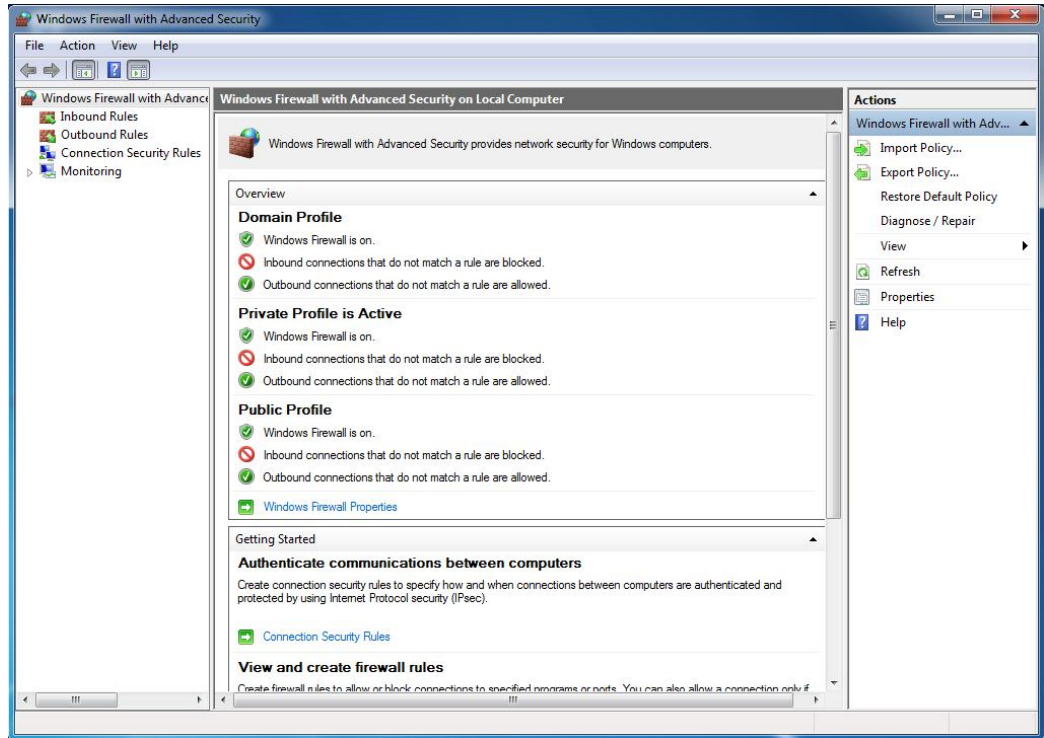
To edit the windows firewall settings, click on the Windows icon (or start button), and enter 'wf.msc' in the search box before pressing enter.



On Windows Vista, you may see a UAC prompt like the one below, in which case click 'Yes' or 'Allow'.

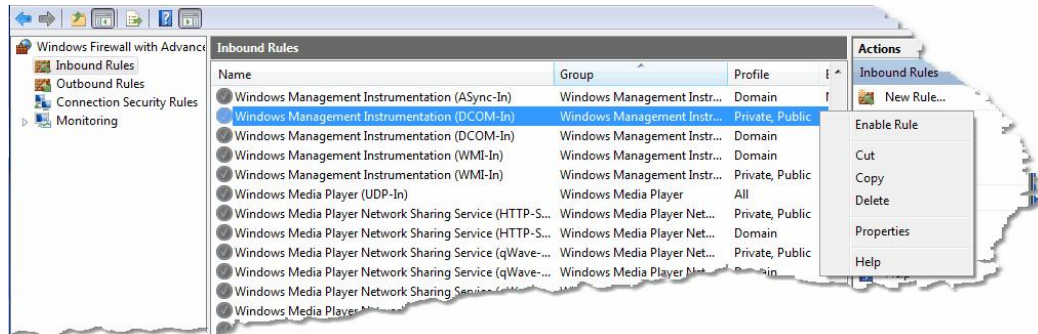


This will show the firewall advanced settings page, as in the sample below.



**Advanced firewall settings**

Click on 'Inbound Rules' and find 'Windows Management Instrumentation (DCOM-In)'. If the 'enabled' column for the rule does not already show 'Yes', right click on the rule and select 'Enable rule' for the menu. The rule may appear more than once for different profiles, so it will need to be enabled in any profile where it will be used.

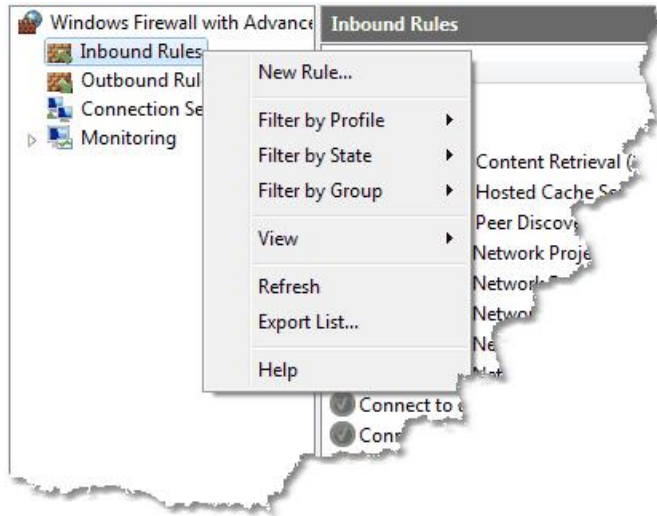


**Enabling DCOM inbound rule**

Every program that will use DCOM must now be added to the exceptions list. These programs include:

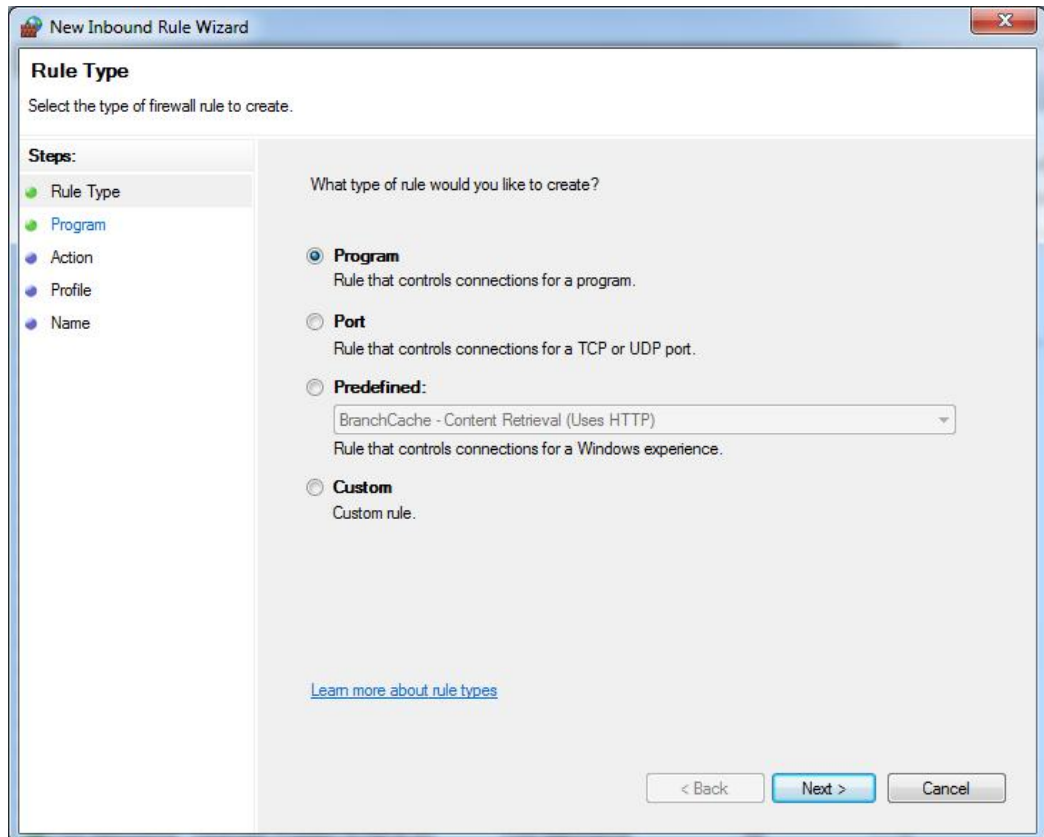
- The OPC server itself (on the server PC)
- The 'OPCenum' service (on the server PC)
- Each client application (on the client PC where it will run)

The example below will show how to add the 'OPCenum' service. Start by right clicking on 'Inbound Rules' and then selecting 'New Rule...'



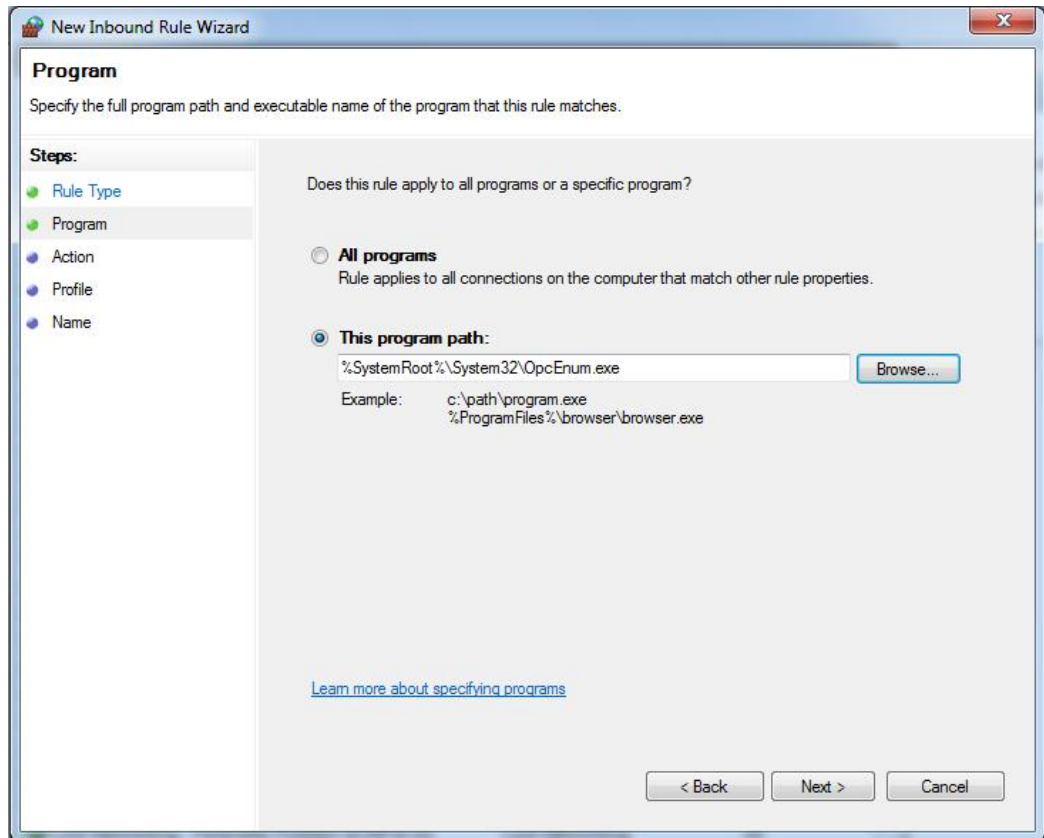
New inbound rule

When the 'New Inbound Rule Wizard' appears, select 'Program' (as shown below) before clicking 'Next'.



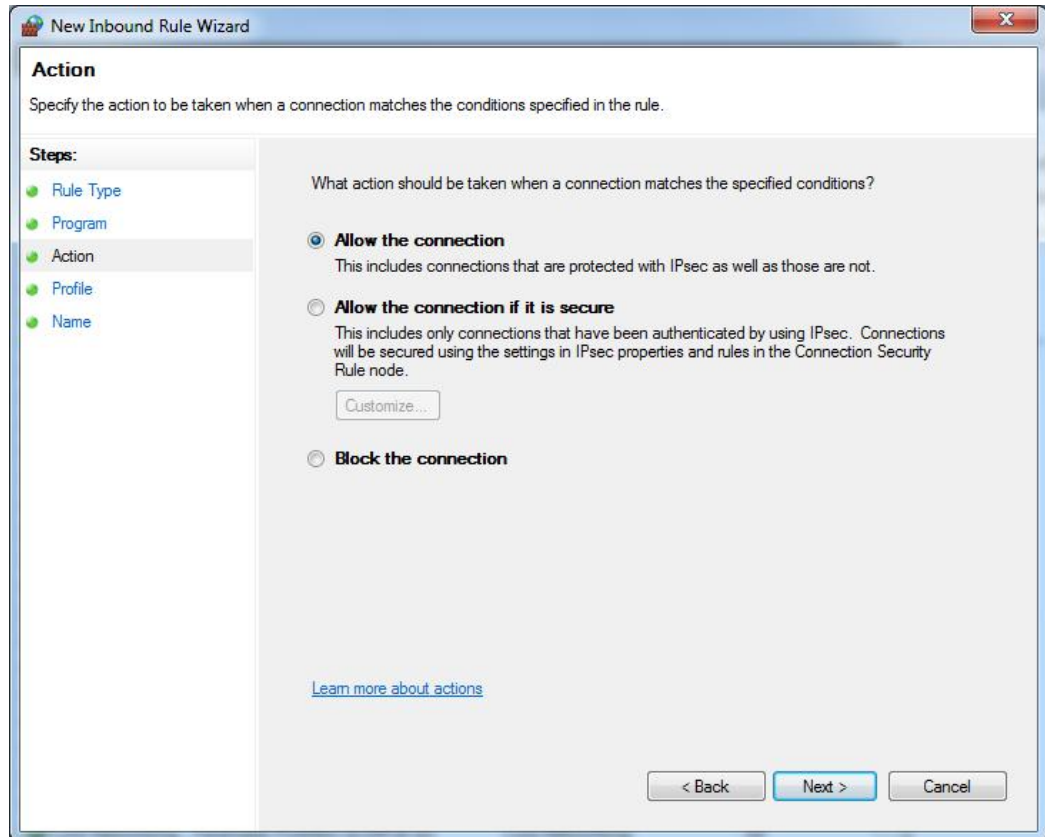
New inbound rule wizard - rule type

On the next page, select 'This program path' and either enter the path to the program to add, or use the 'Browse' button to search for it. The example picture below shows the usual path to OpcEnum.exe. Depending on your operating system, OpcEnum.exe could also be at '%SystemRoot%\SysWOW64\OpcEnum.exe' (expanded to 'C:\Windows\SysWOW64\OpcEnum.exe').



Select program path

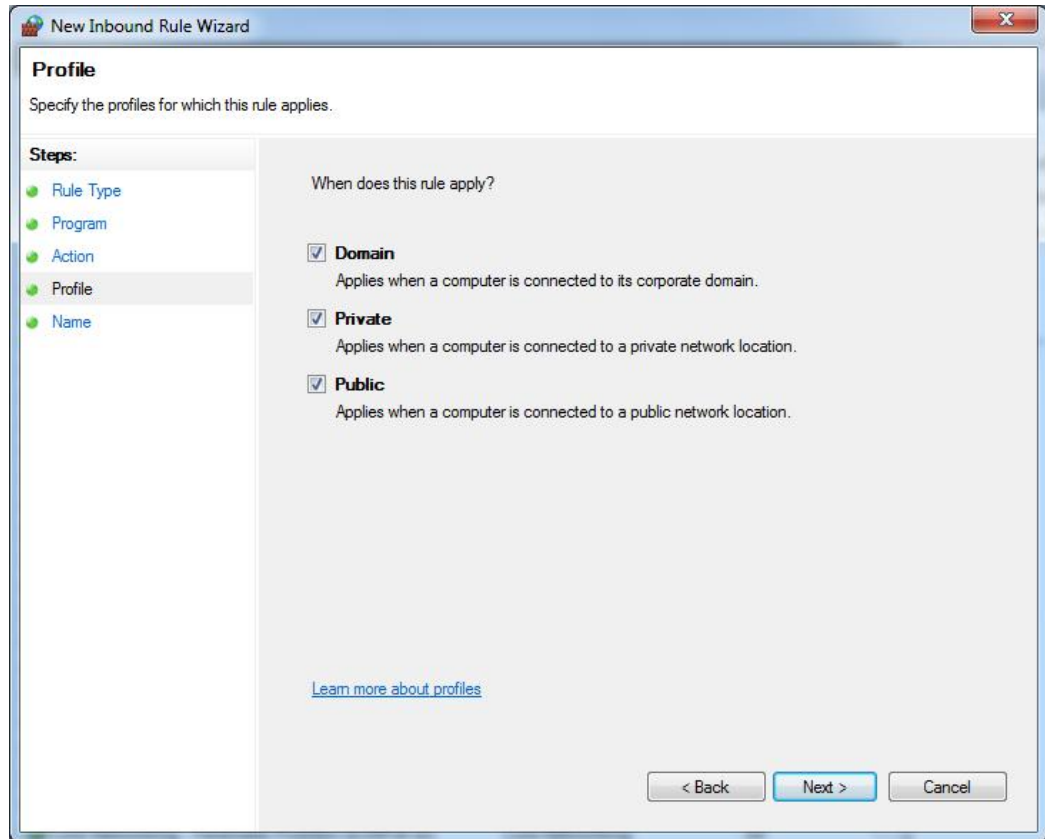
Click 'Next' to show the 'Action' page. Select 'Allow the connection' (or alternatively if you have configured IPsec, select 'Allow the connection if it is secure').



Select action

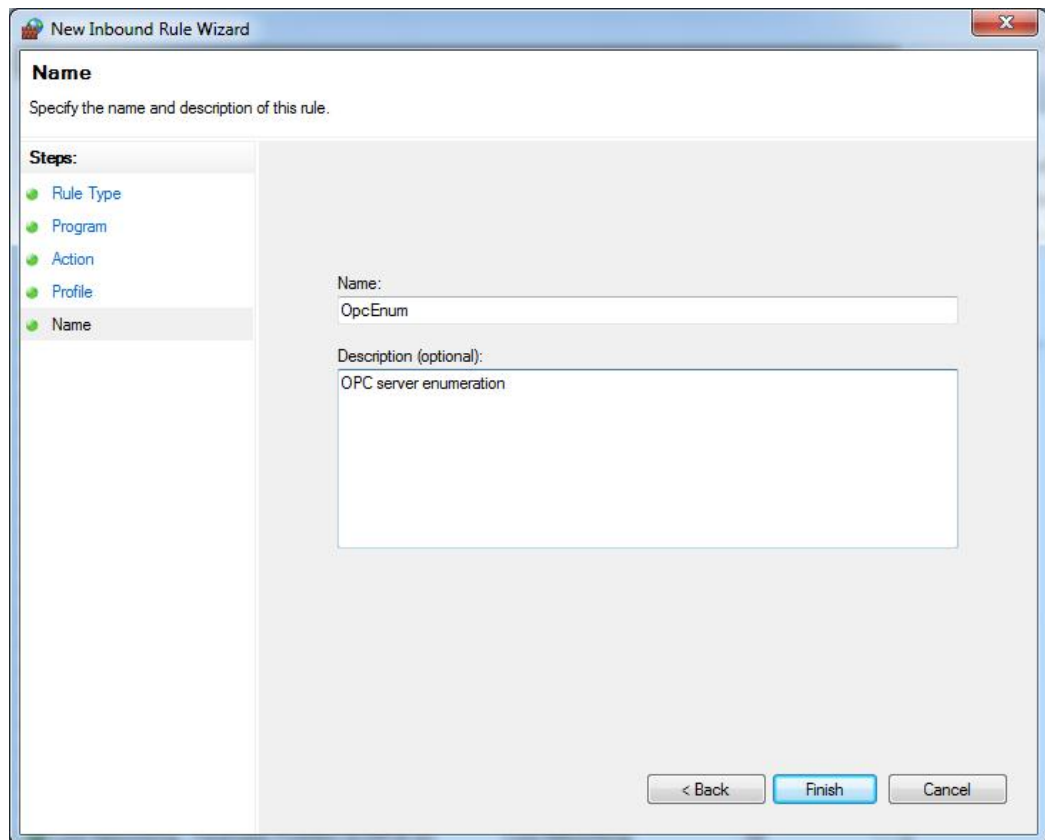


Click 'Next' to show the 'Profiles' page. Make sure there is a check mark next to all the profiles that should use this rule.



Select profiles

Click 'Next' to show the screen below, where a name and description can be entered for the new rule.



#### Enter description

When you click 'Finish', the new rule will appear in the incoming rules list.

Repeat the above steps to add the OPC server itself (normally located in '%SystemDrive%\MELSEC\MX OPC server 6.05\MXOPC.exe', but it will depend on the path selected during installation).

On the client PC, it may be necessary to perform this step for each client application.

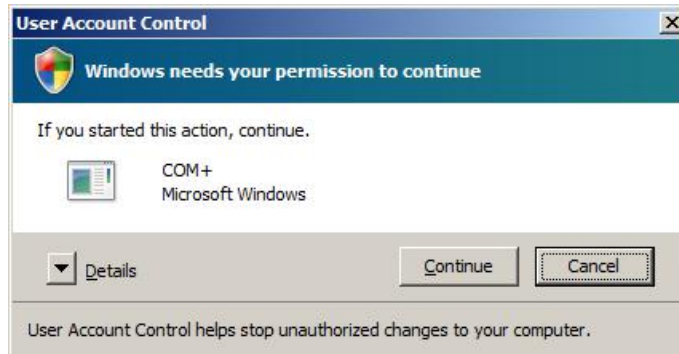
When all applications that will use DCOM have been added, select 'File' -> 'Exit' to close the firewall settings screen.

### 14.4.2 Configuring DCOM security

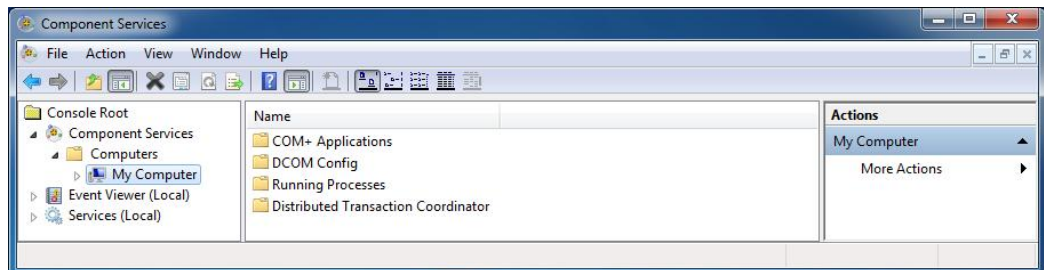
Click on the Windows icon or 'Start' button, then in the search box type 'dcomcnfg' and press enter.



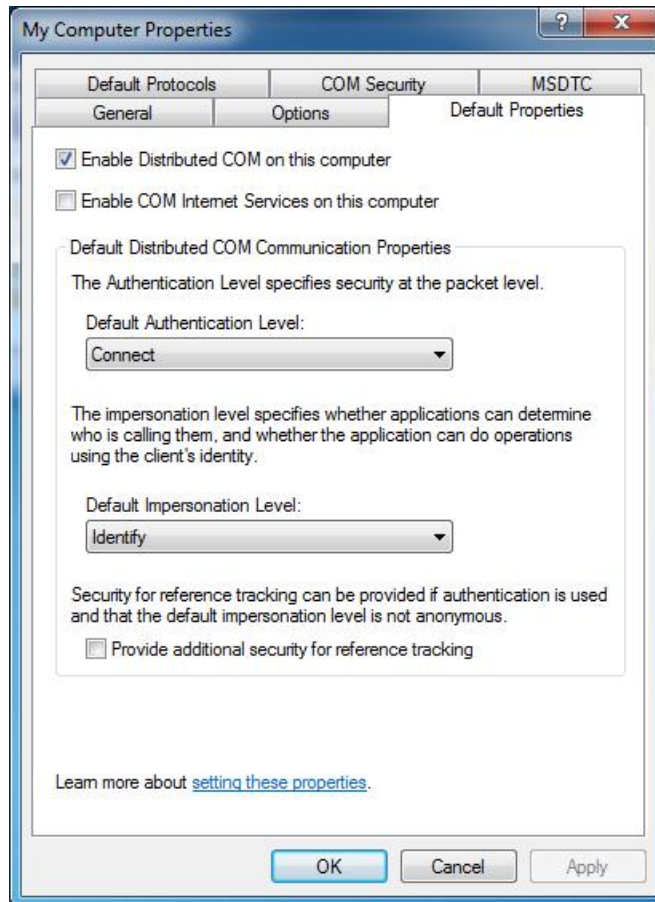
On Windows Vista, you may see the user account control prompt shown below. Click 'Continue' or 'Yes'.



In the first window that appears (see example below), click on the 'Component services' and 'Computers' items in the left hand pane to expand them, so that 'My Computer' is shown.



Right click on 'My computer' and select 'Properties' from the pop-up menu. In the next window, click on the 'Default properties' tab and a window similar to the one below will be shown:

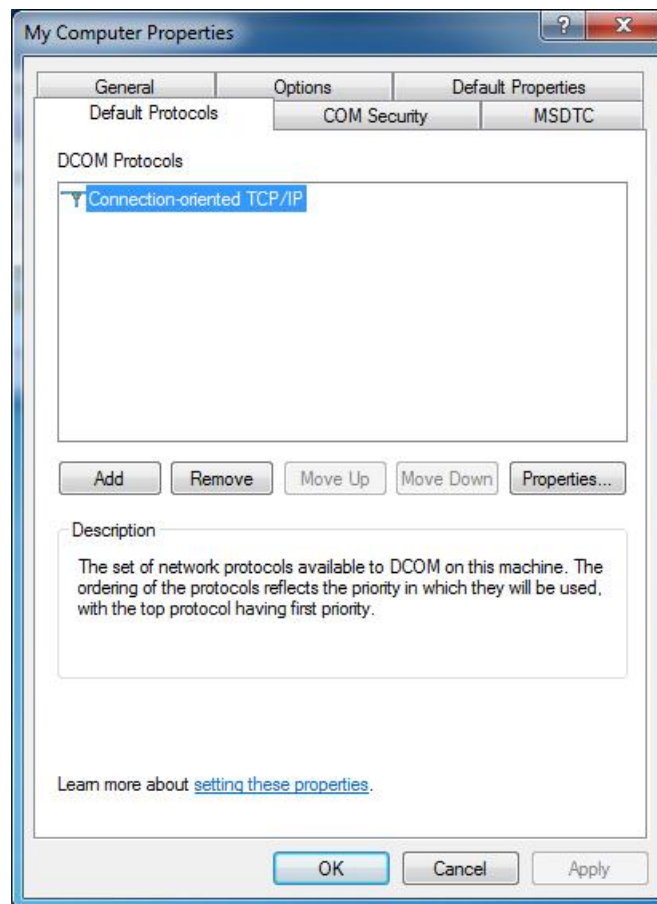


Make sure that 'Enable distributed COM on this computer' is checked.

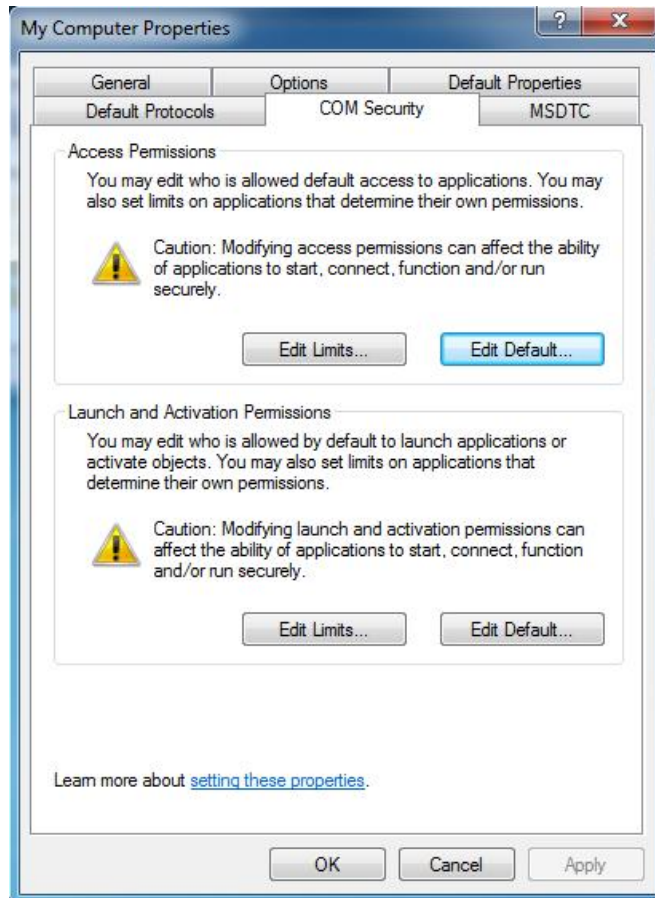
The 'Default authentication level' should normally be set to 'Connect', which checks the user's details once when a connection is first made. If there are problems with the connection, the minimal security option 'None' can be used (Note: this requires that the 'Anonymous login' user is given access rights, as described later).

The 'Default impersonation level' should normally be set to 'Identify'. If you are having problems connecting, this can be changed to 'Impersonate'.

Switch to the 'Default protocols' tab. In this dialog (see example picture below), make sure that the 'Connection oriented TCP/IP' protocol appears at the top of the list. If there are any other protocols in the list which you are not using, remove them.

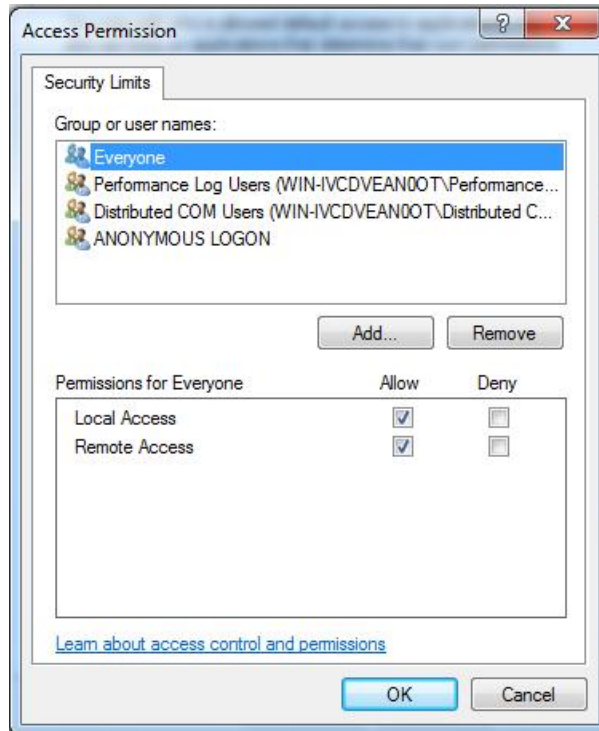


Click on the 'COM security' tab to show the dialog below:



The 'Limits' on this page will override the COM security settings for individual components, and if the settings are too restrictive DCOM will not be able to access an OPC server remotely.

In the 'Access permissions' section, click 'Edit limits' to display the window below.

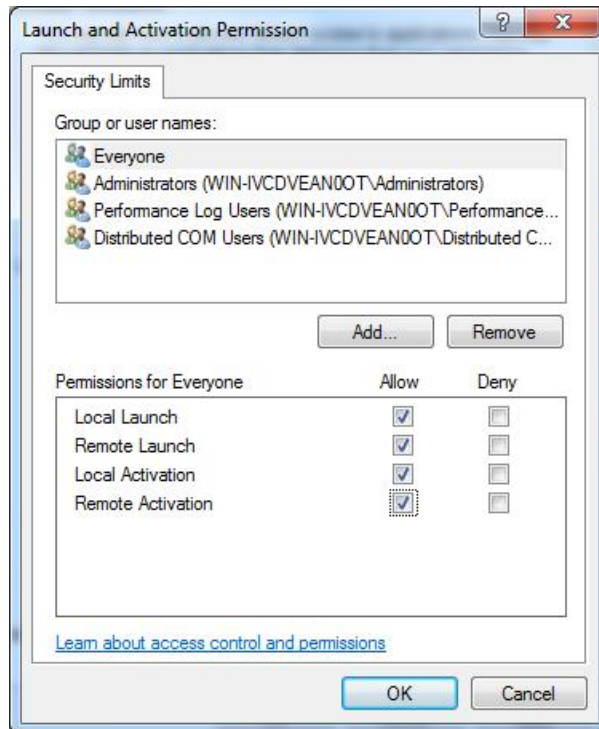


If 'ANONYMOUS LOGON' does not already appear in the list, add it – this is required to allow enumeration of OPC servers by remote PCs, and for applications where minimal security is being used due to other connection issues. If you do not need to fetch a list of available OPC servers from other PCs, you may be able to increase the server's security by leaving remote access for 'Anonymous logon' switched off.

In the example above, DCOM can be used by all users (the 'Everyone' group). Alternatively, if you are enabling DCOM for specific users or a group of users, the accounts can be added to this list individually.

Any user accounts in this list that need to use DCOM must have the 'Allow' checkbox set for both 'Local access' and 'Remote access'.

Click 'OK' to save the changes, then on the 'COM security' screen click the 'Edit limits' button in the 'Launch and Activation Permissions' section to show the window below:



Ensure that all the 'Allow' checkboxes are set for each user or group that will access the server remotely. In the above picture, the 'Everyone' group has been enabled, but for a more secure configuration, you can add only the user(s) or group(s) that will actually use DCOM.

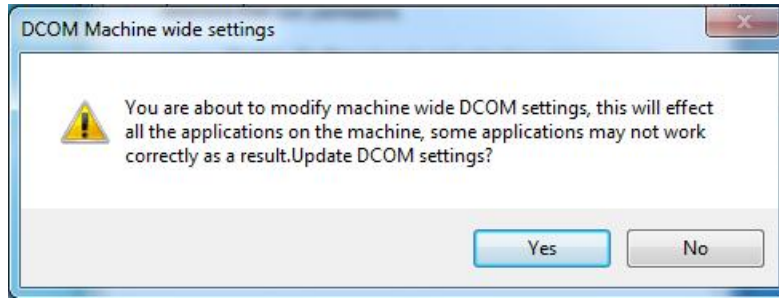
Click 'OK' to return to the 'COM security' window. The 'Edit defaults' settings for 'Access permissions' and 'Launch permissions' on this page are used to set the default permissions for COM applications. An application can either use the default settings, or can set customized security permissions (described later). These dialogs have the same format as the 'Edit limits' screens above.

In the 'Edit defaults' setting for 'Access permissions', add the users or groups that will need OPC access, and ensure that the 'Allow' box is checked for both local and remote access. For a minimal security configuration, the 'Everyone' user can be given these rights.

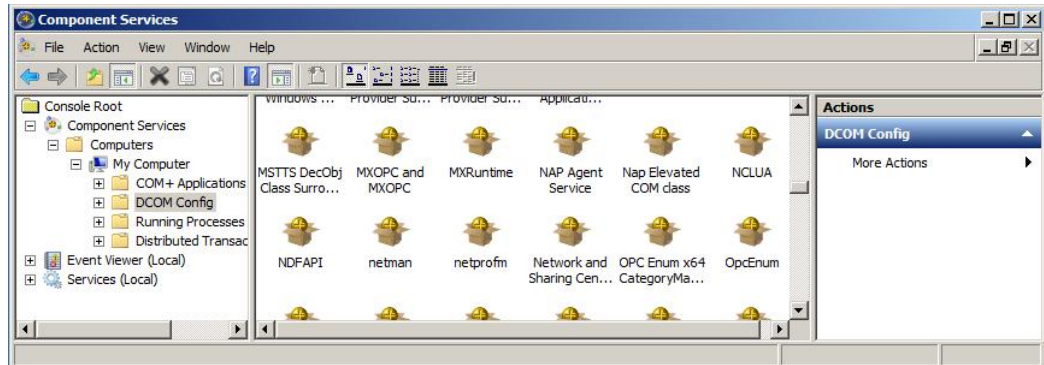
Repeat this step for the 'Edit defaults' setting for 'Launch and activation permissions', ensuring that all of the 'allow' check boxes are set.



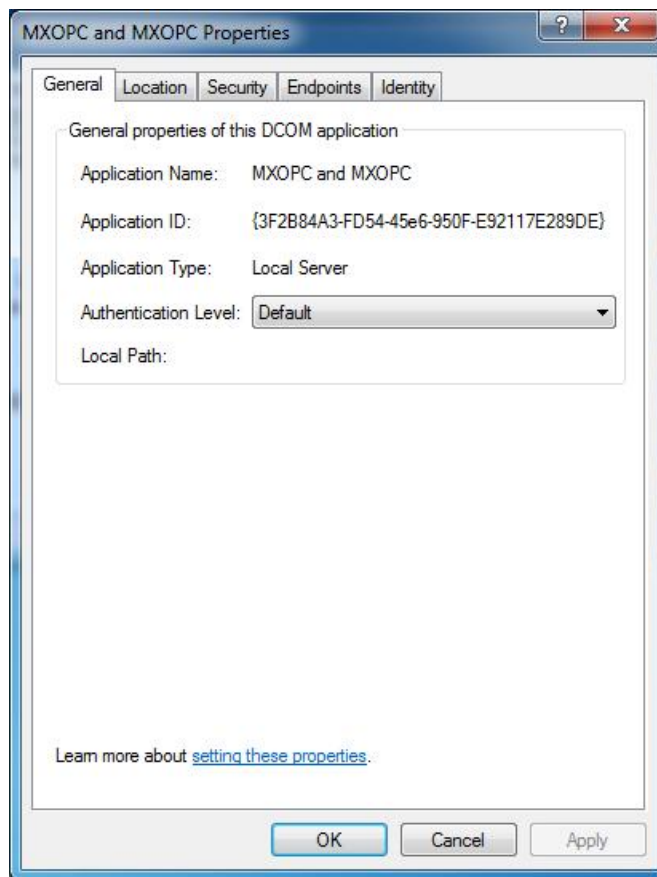
When you have finished, click 'OK' to return to the 'Component Services' dialog. You may see a warning message like the one below, in which case answer 'Yes'.



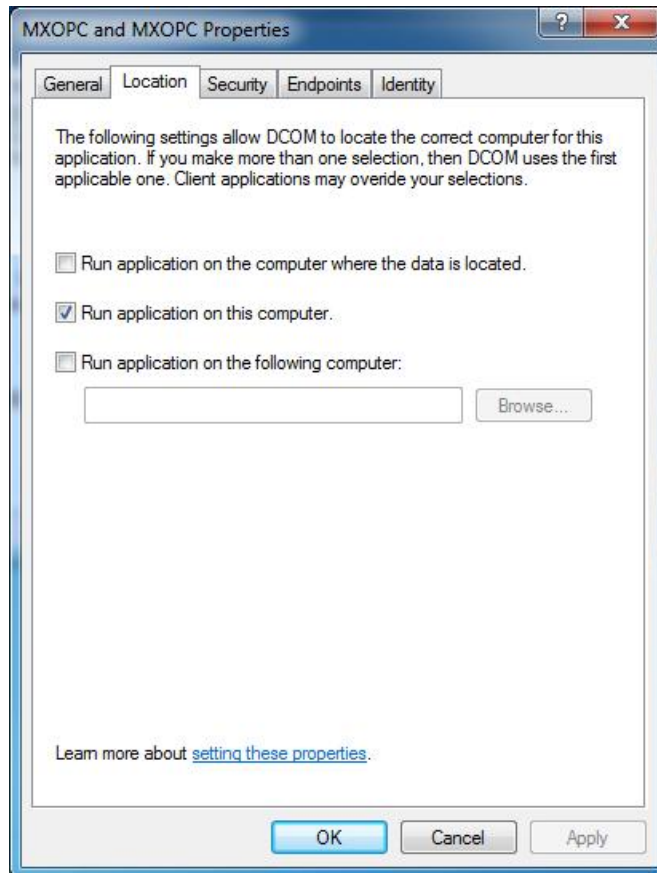
For the server PC configuration (only), you will need to configure the security on the MX OPC server, MX Runtime and OPC enumeration services. To do this, double-click on the 'DCOM config' entry, and find 'MXOPC', 'MXRuntime' and 'OpcEnum' in the list, as shown in the picture below. Depending on your operating system and installation, these items may not be visible in the window at the same time, and there may also be an 'OPC Enum x64 CategoryManager' item.



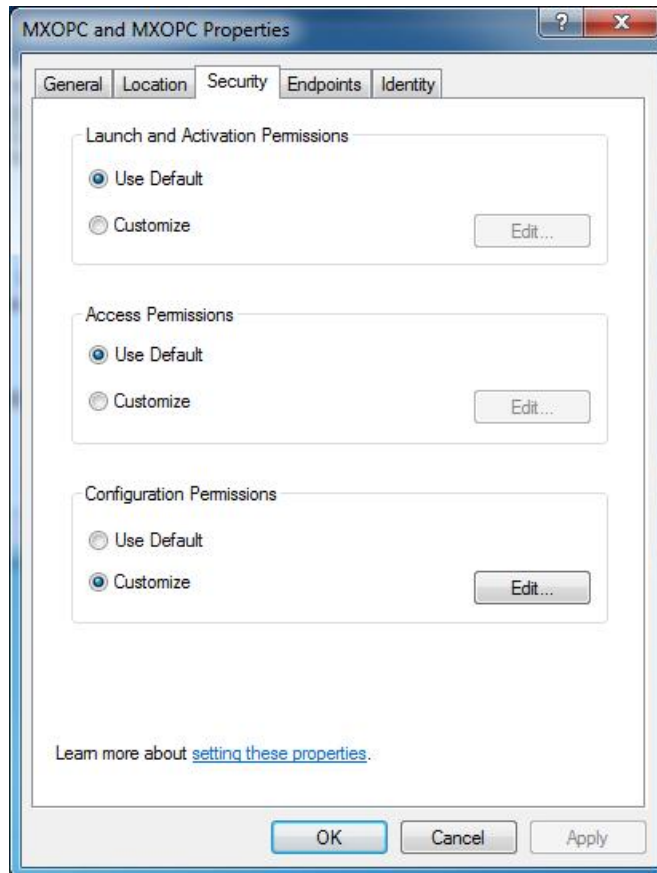
This step will need to be repeated for each item in turn – 'MXOPC' will be used for this example. Right click on the 'MXOPC' icon and select 'properties'. The window below should appear; confirm that the authentication level setting is 'Default'.



Switch to the 'Location' tab as shown below. Confirm that the 'Run application on this computer' check box is selected.

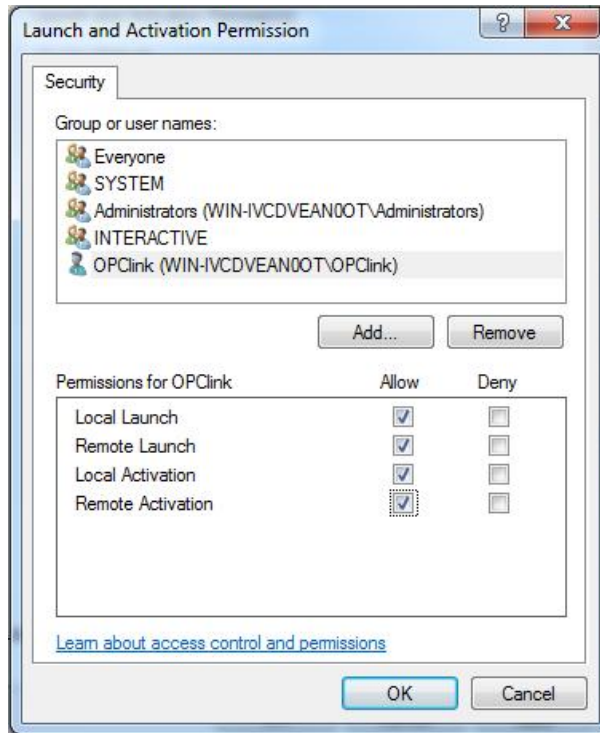


Now switch to the 'Security tab'.

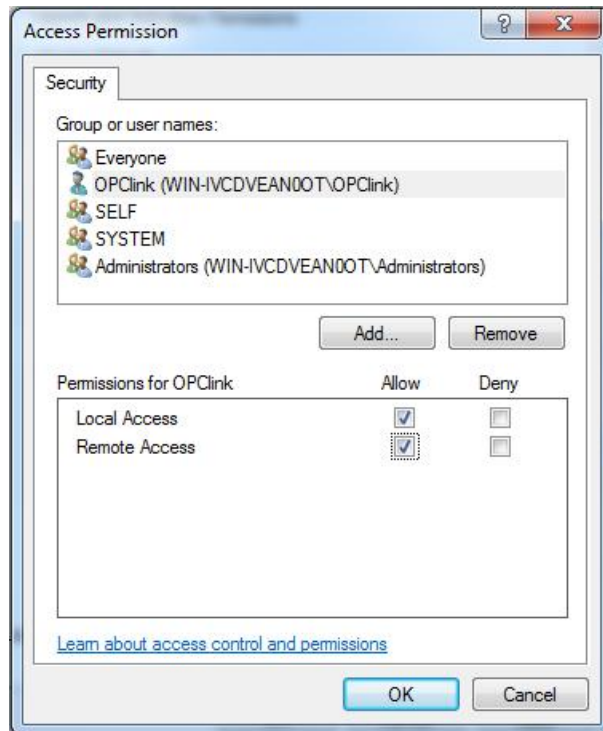


The 'Launch and activation permissions' and 'Access permissions' can both be left as 'Default' to use the default security settings configured earlier. Alternatively, they can be customized to allow more control over the users that can start and use each server. To do this, select 'customize' for each option and select 'Edit'.

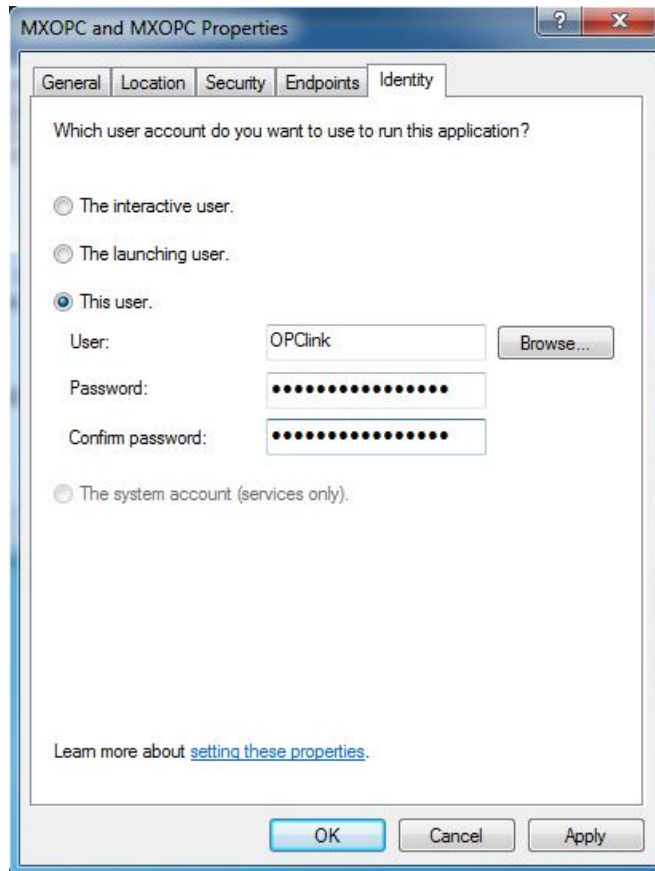
A sample 'Launch and activation permissions' edit window is shown below, in which a named user 'OPClink' has been added. As with the 'edit defaults' settings, all four 'allow' check boxes must be set for users requiring access through DCOM.



The 'Access permissions' configuration screen is shown below, with a named user 'OPClink' added.



After changing the settings on the security tab (if required), select the 'Identity' tab. In the sample screen below, the server has been configured to run as a named user.



The options on this screen are:

- **The interactive user** – the user who is currently logged on to the machine. This can cause problems with DCOM, as there is no guarantee of which user (if any) will be logged on when the server is accessed from another computer.
- **The launching user** – the user who accessed the server, causing it to be started.
- **A named user** – the server runs as a specified user, whose name and password are provided. This option allows the server to be configured independently of the user that started the server and the user that is logged on, and works well for unattended computers.
- **The system account**, when the server is configured to run as a service.

Press 'OK' to return to the 'Component settings' page, which can then be closed.

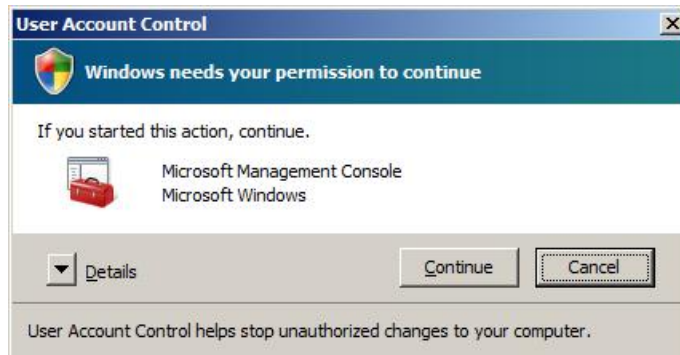
### 14.4.3 Configuring services to use a specific user account

When the OPC server is installed as a service, it can be useful to configure it to run using a specific user account. The 'OPC enumeration' EXE is also installed as a service. In this section, the 'OPC enumeration' EXE will be used as an example, but the same method applies when the OPC server settings are changed.

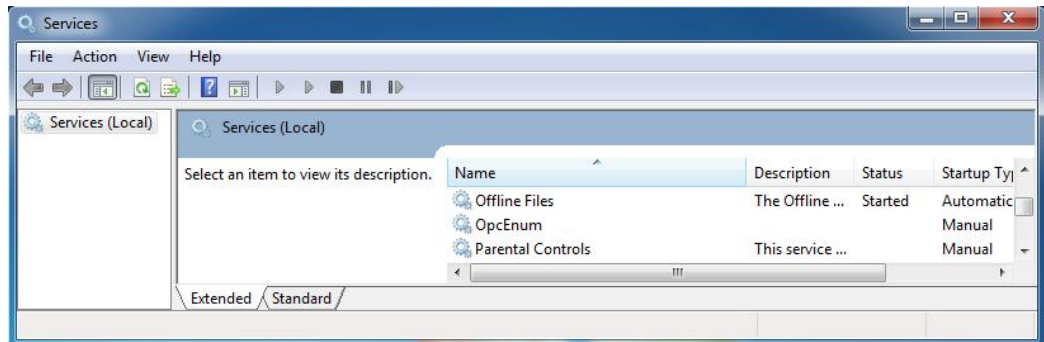
To open the 'services' page, click on the Windows icon (or 'Start' button) then type 'Services.msc' in the search box before pressing enter.



If the warning below appears, select 'Continue' or 'Yes'.

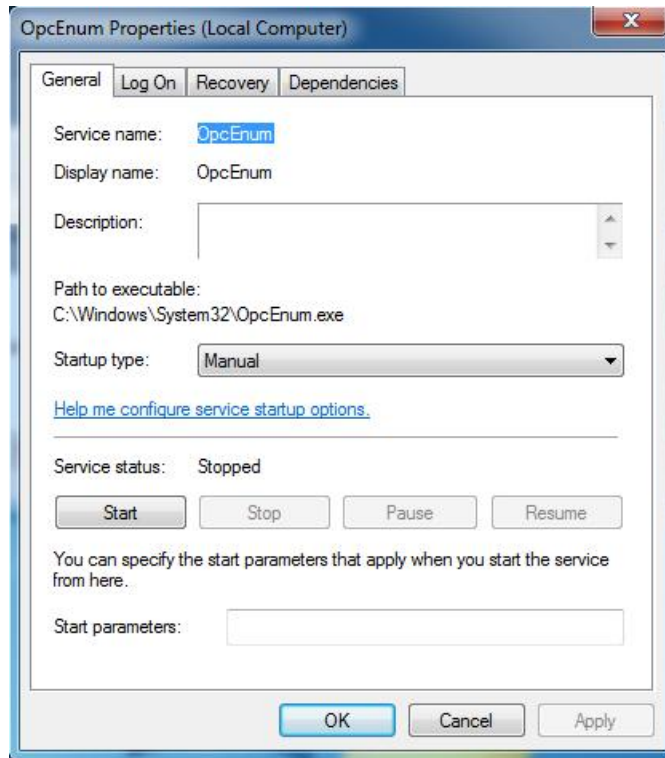


Find the service to be changed in the list. The sample screen below shows the OPC enumeration service. You may also need to change the settings for the 'Mitsubishi Electric MX OPC server 6.05' service.

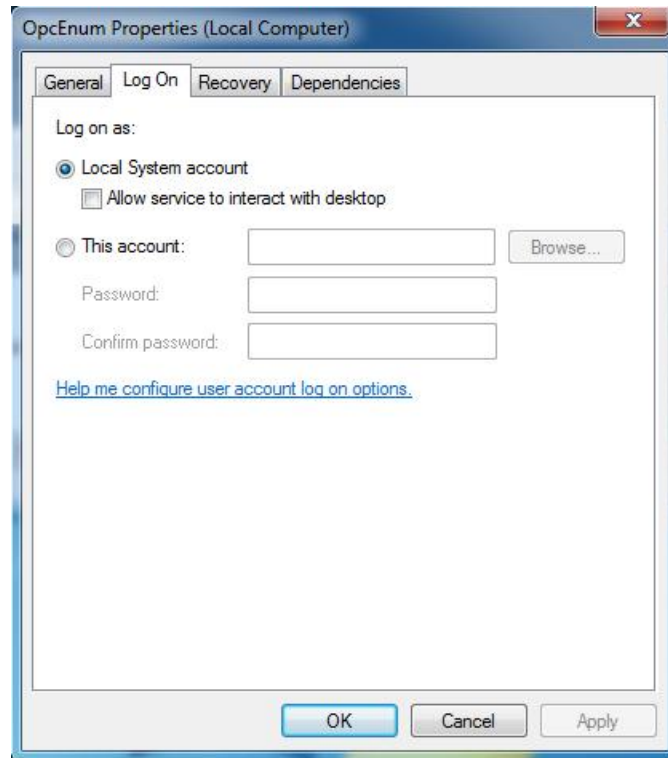




Right click on the service name and select 'Properties'. The service configuration screen (below) will be shown.



Switch to the 'Log On' tab, select 'This account' and enter the username and password of the user to run as.



Now select 'OK' and close the services window.

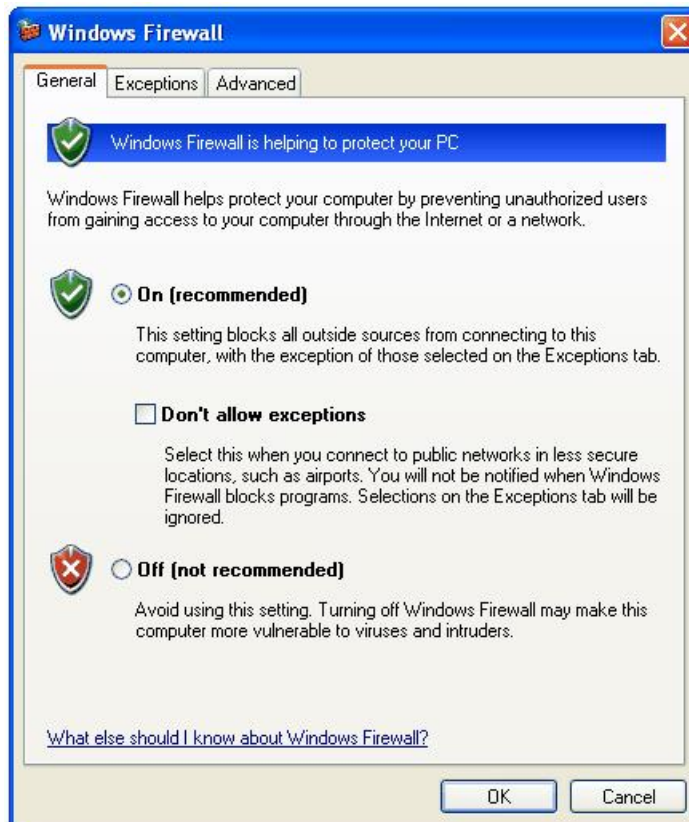
## 14.5 Configuring DCOM on Windows XP service pack 2

This section contains instructions specific to Windows XP. Most of the changes in this section must be made on the server PC and also on the client PC, so that the connection works in both directions.

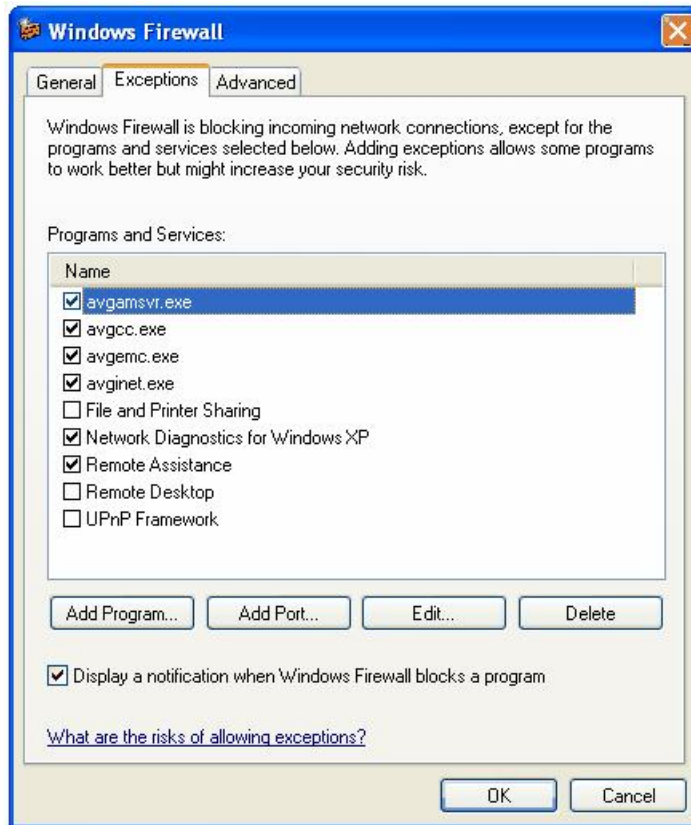
### 14.5.1 Configuring the Windows firewall

If you are using the Windows firewall provided with Windows XP service pack 2 (or later), you will need to configure it to allow DCOM network traffic to pass through. If you are using a third party software firewall, please refer to the manual to find out how to make the equivalent configuration changes. If DCOM does not work when your third-party firewall is enabled, but works when it is switched off, the problem is likely to be with the firewall configuration.

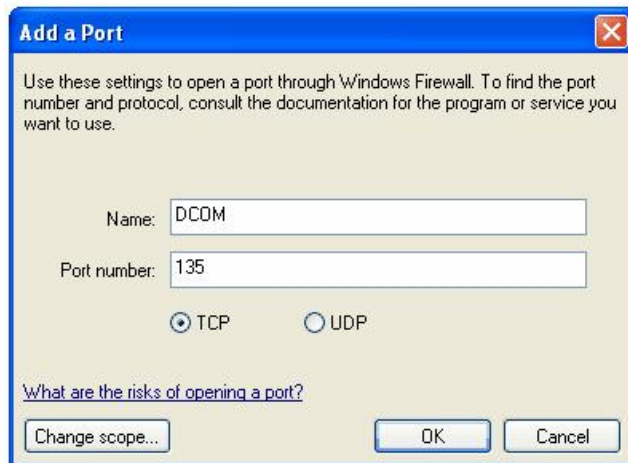
From the 'Start' menu, open 'Control panel'. If you are using 'Classic view', the 'Windows firewall' icon will already appear in the list, otherwise in 'Category view' it will be under the 'Network and internet connections' category. Double-click the icon to view the firewall settings screen (shown below).



Click on the 'Exceptions' tab, and a screen similar to the one below will be shown.



Click the 'Add port...' button, and enter the details below to allow other applications to connect to the DCOM port. In third party firewalls, this may be referred to as 'RPC'.

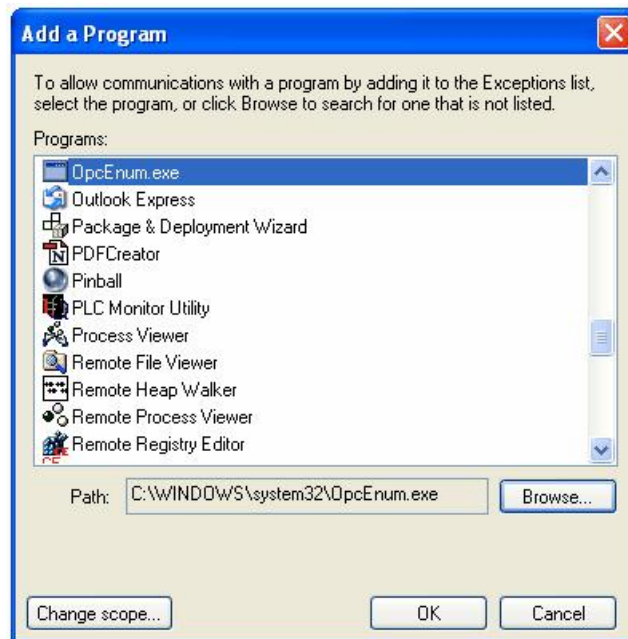


Click 'OK' to return to the previous screen, where 'DCOM' will have been added to the exceptions list.

Every program that will use DCOM must now be added to the exceptions list. These programs include:

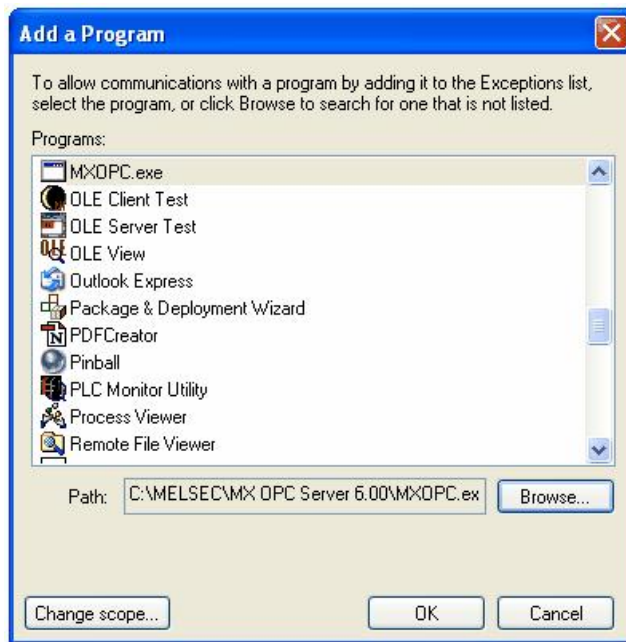
- The OPC server itself (on the server PC)
- The 'OPCenum' service (on the server PC)
- Each client application (on the client PC where it will run)

The example below will show how to add the 'OPCenum' service. Start by clicking on the 'Add program...' button on the exceptions screen. In the window that appears, use the 'Browse' button and find the file 'OPCenum.exe', which is normally located in the Windows\System32 directory. The screen should now look like the one below:



Click 'OK' to return to the exceptions screen, where the program will have been added.

Repeat the 'Add program...' steps for the OPC server itself (normally located in 'C:\MELSEC\MX OPC server 6.05\MXOPC.exe', but it will depend on the path selected during installation), as in the example screen below.



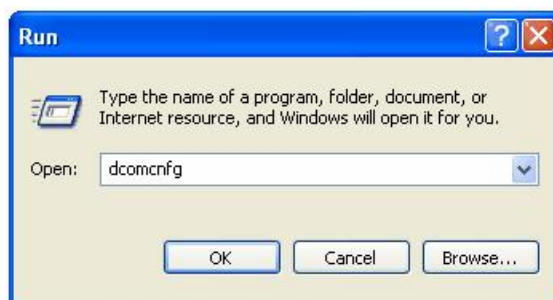
Repeat the 'Add program...' steps for the 'Microsoft Management Console' (this can usually be found in 'C:\Windows\System32\MMC.exe' but this will depend on your Windows installation).

On the client PC, you will need to perform this step for each client application.

When all applications that will use DCOM have been added, click 'OK' on the exceptions screen to finish.

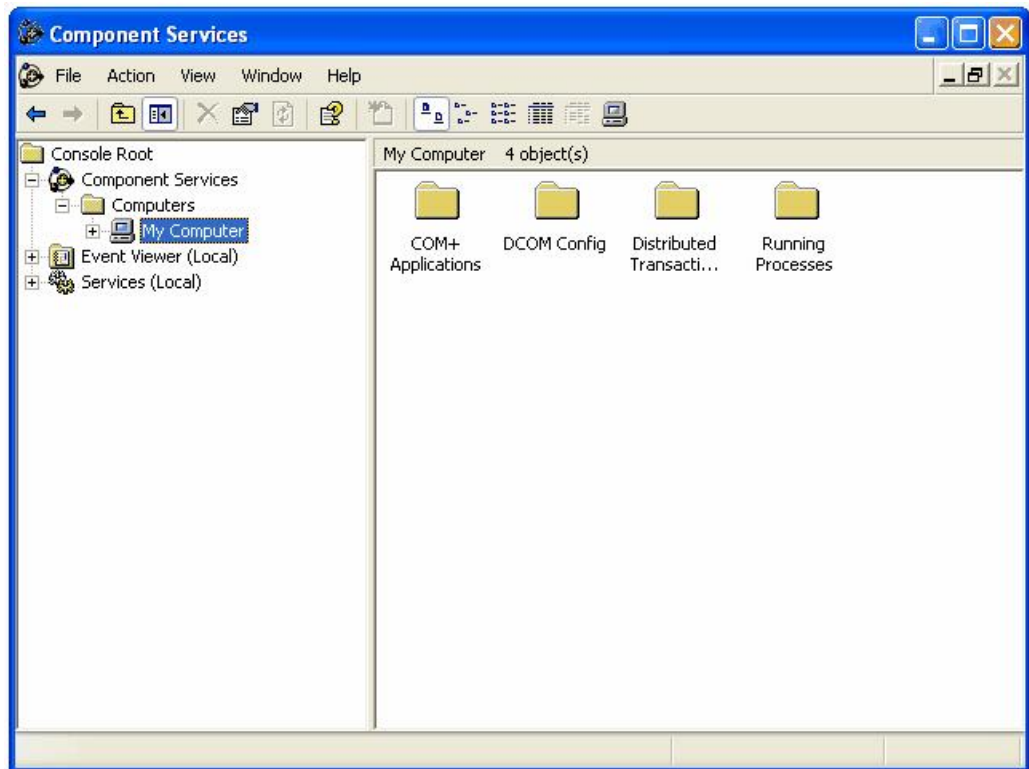
### 14.5.2 Configuring DCOM security

From the 'Start' menu, select 'Run' to show the window below.

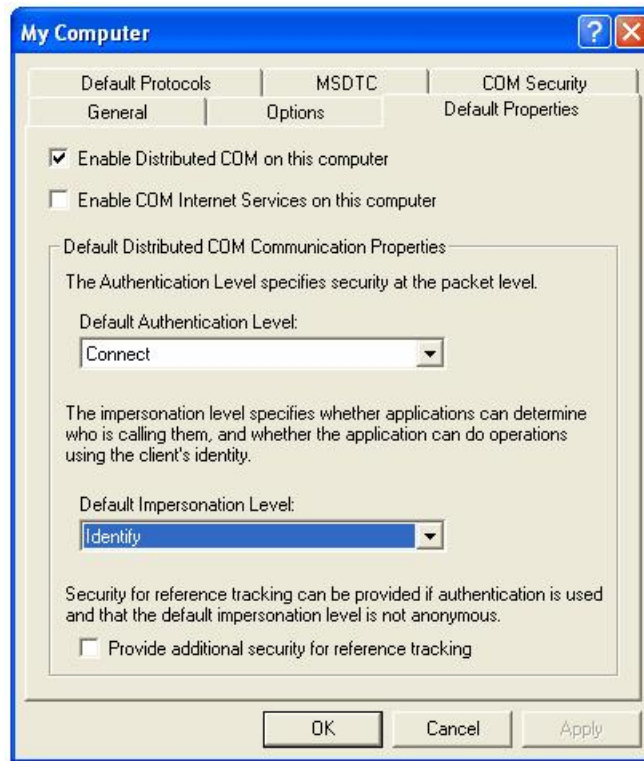


Enter 'dcomcnfg' and click 'OK' to start the DCOM configuration utility.

In the first window that appears (see example below), click on the 'Component services' and 'Computers' items in the left hand pane to expand them, so that 'My Computer' is shown.



Right click on 'My computer' and select 'Properties' from the pop-up menu. In the next window, click on the 'Default properties' tab and a window similar to the one below will be shown:



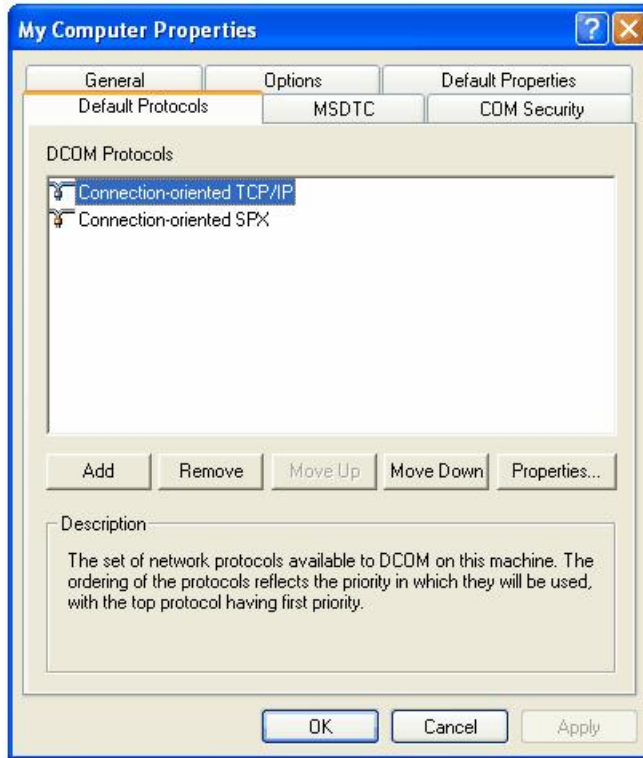
Make sure that 'Enable distributed COM on this computer' is checked.

The 'Default authentication level' should normally be set to 'Connect', which checks the user's details once when a connection is first made. If there are problems with the connection, the minimal security option 'None' can be used (Note: this requires that the 'Anonymous login' user is given access rights, as described later).

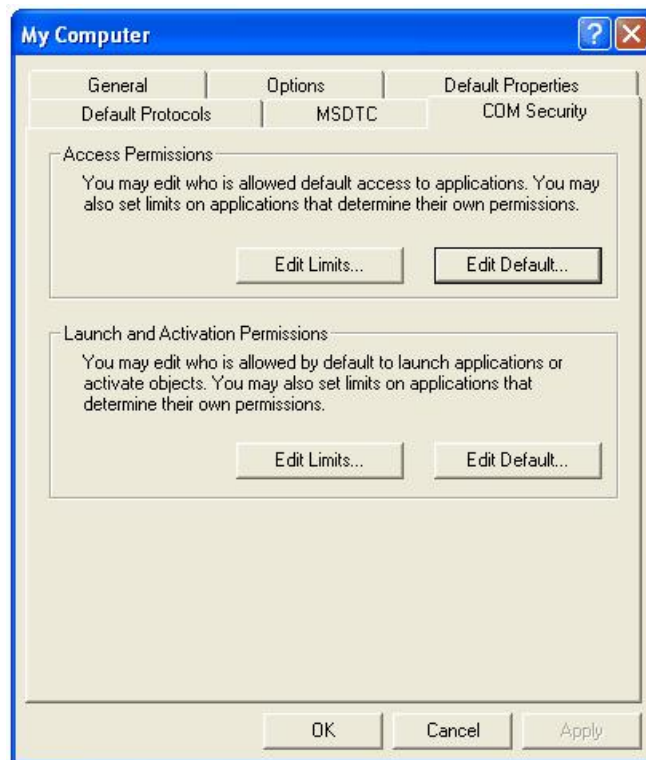
The 'Default impersonation level' should normally be set to 'Identify'. If you are having problems connecting, this can be changed to 'Impersonate'.



Switch to the 'Default protocols' tab. In this dialog (see example picture below), make sure that the 'Connection oriented TCP/IP' protocol appears at the top of the list. If you are not using the other protocols, remove them.

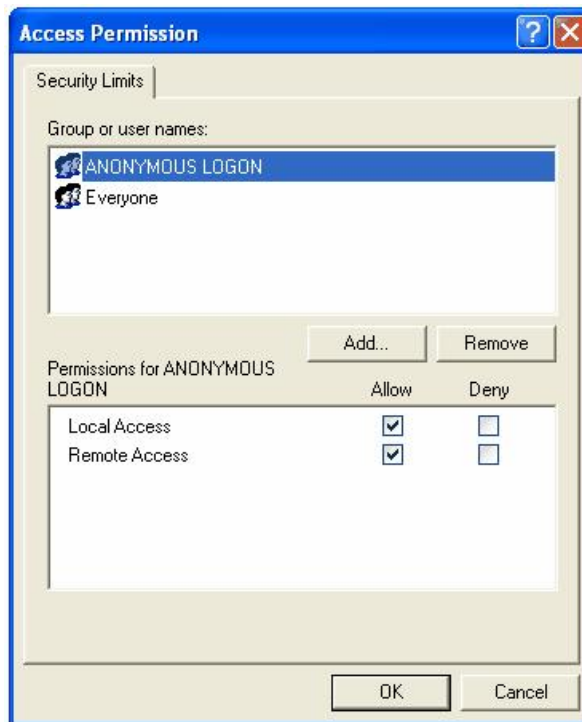


Click on the 'COM security' tab to show the dialog below:



The 'Limits' on this page will override the COM security settings for individual components, and from XP service pack 2 onwards the default settings are too restrictive to allow OPC communications to work. The restrictions must be relaxed before using DCOM to access an OPC server remotely.

In the 'Access permissions' section, click 'Edit limits' to display the window below.

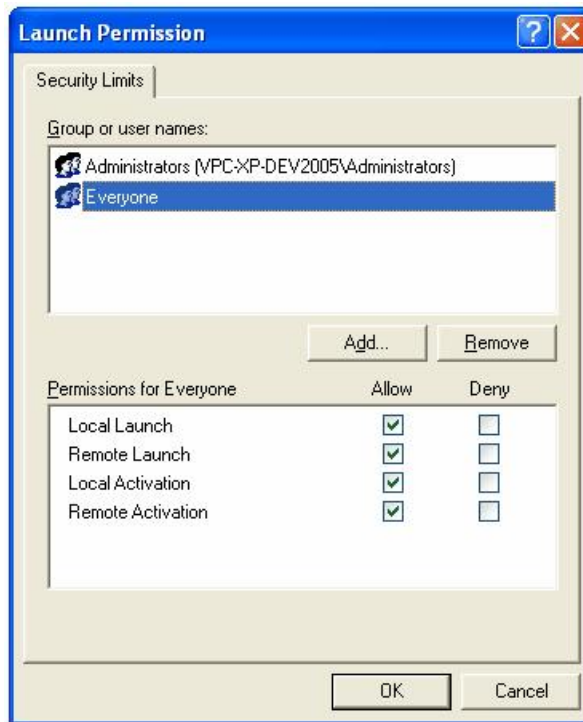


If 'ANONYMOUS LOGON' does not already appear in the list, add it – this is required to allow enumeration of OPC servers by remote PCs, and for applications where minimal security is being used due to other connection issues. If you do not need to fetch a list of available OPC servers from other PCs, you may be able to increase the server's security by leaving remote access for 'Anonymous logon' switched off.

In the example above, DCOM can be used by all users (the 'Everyone' group). Alternatively, if you are enabling DCOM for specific users or a group of users, the accounts can be added to this list individually.

Any user accounts in this list that need to use DCOM must have the 'Allow' checkbox set for both 'Local access' and 'Remote access'.

Click 'OK' to save the changes, then on the 'COM security' screen click the 'Edit limits' button in the 'Launch and Activation Permissions' section to show the window below:



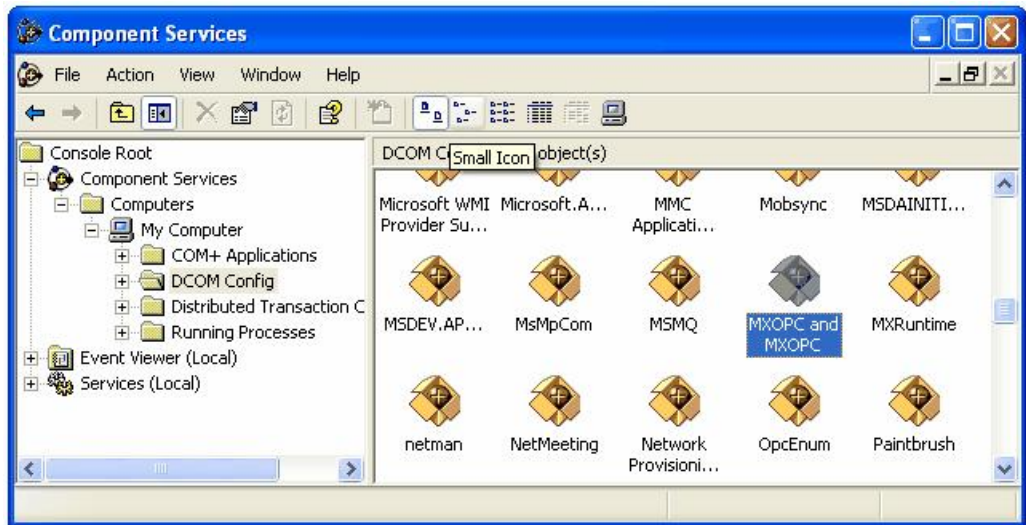
Ensure that all the 'Allow' checkboxes are set for each user or group that will access the server remotely. In the above picture, the 'Everyone' group has been enabled, but for a more secure configuration, you can add only the user(s) or group(s) that will actually use DCOM.

Click 'OK' to return to the 'COM security' window. The 'Edit defaults' settings for 'Access permissions' and 'Launch permissions' on this page are used to set the default permissions for COM applications. An application can either use the default settings, or can set customized security permissions (described later). These dialogs have the same format as the 'Edit limits' screens above.

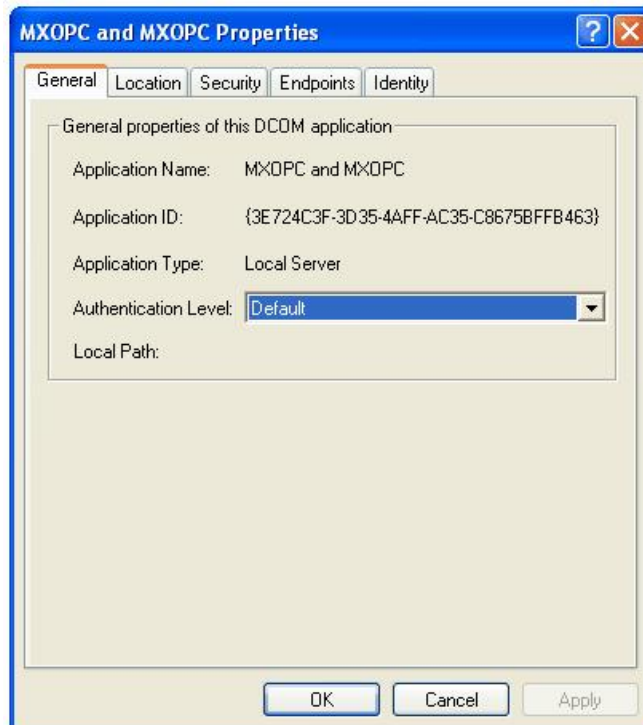
In the 'Edit defaults' setting for 'Access permissions', add the users or groups that will need OPC access, and ensure that the 'Allow' box is checked for both local and remote access. For a minimal security configuration, the 'Everyone' user can be given these rights.

Repeat this step for the 'Edit defaults' setting for 'Launch and activation permissions', ensuring that all of the 'allow' check boxes are set.

When you have finished, click 'OK' to return to the 'Component Services' dialog. For the server PC configuration (only), you will need to configure the security on the MX OPC server, MX Runtime and OPC enumeration services. To do this, double-click on the 'DCOM config' entry, and find 'MXOPC', 'MXRuntime' and 'OpcEnum' in the list, as shown in the picture below.



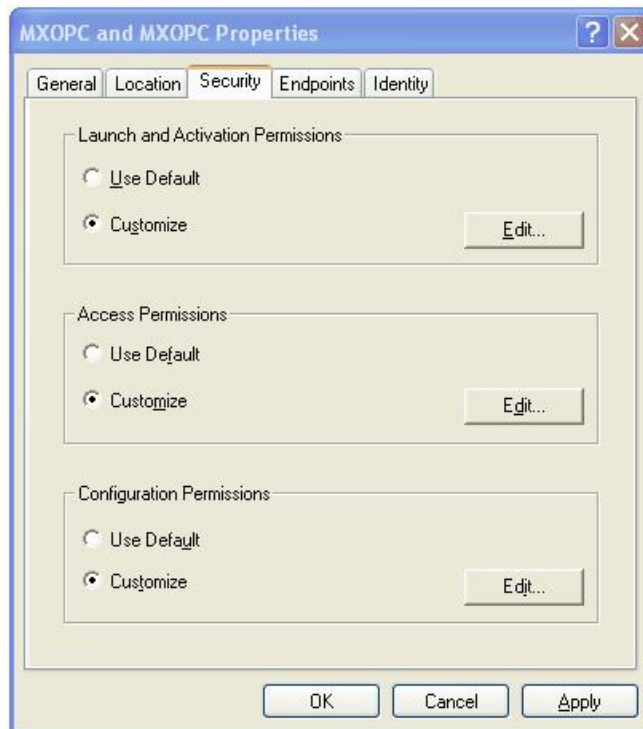
This step will need to be repeated for 'MXOPC', 'MXRuntime' and 'OpcEnum' in turn – 'MXOPC' will be used for this example. Right click on the 'MXOPC' icon and select 'properties'. The window below should appear; confirm that the authentication level setting is 'Default'.



Switch to the 'Location' tab as shown below. Confirm that the 'Run application on this computer' check box is selected.

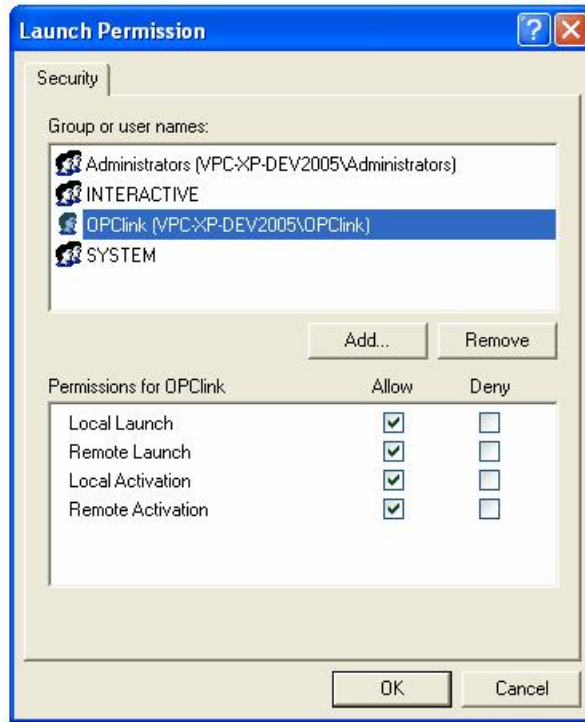


Now switch to the 'Security tab'.



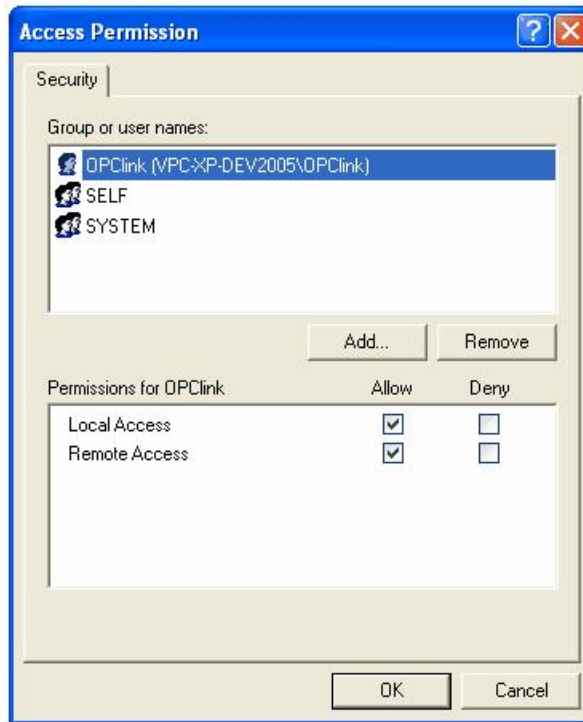
The 'Launch and activation permissions' and 'Access permissions' can both be left as 'Default' to use the default security settings configured earlier. Alternatively, they can be customized to allow more control over the users that can start and use each server. To do this, select 'customise' for each option and select 'Edit'.

A sample 'Launch and activation permissions' edit window is shown below, in which a named user 'OPClink' has been added. As with the 'edit defaults' settings, all four 'allow' check boxes must be set for users requiring access through DCOM.





The 'Access permissions' configuration screen is shown below, with a named user 'OPClink' added.



After changing the settings on the security tab (if required), select the 'Identity' tab. In the sample screen below, the server has been configured to run as a named user.



The options on this screen are:

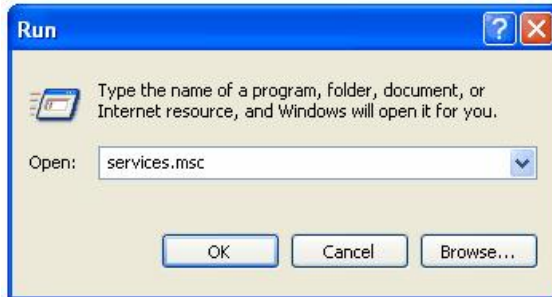
- **The interactive user** – the user who is currently logged on to the machine. This can cause problems with DCOM, as there is no guarantee of which user (if any) will be logged on when the server is accessed from another computer.
- **The launching user** – the user who accessed the server, causing it to be started.
- **A named user** – the server runs as a specified user, whose name and password are provided. This option allows the server to be configured independently of the user that started the server and the user that is logged on, and works well for unattended computers.
- **The system account**, when the server is configured to run as a service.

Press 'OK' to return to the 'Component settings' page, which can then be closed.

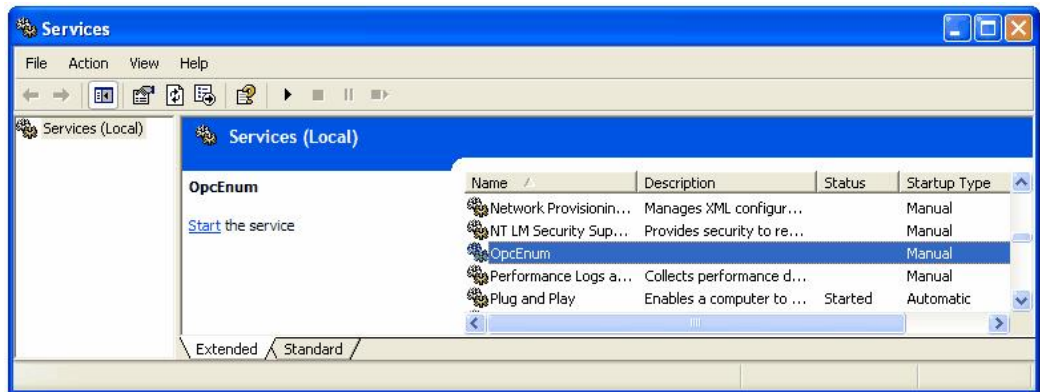
### 14.5.3 Configuring services to use a specific user account

When the OPC server is installed as a service, it can be useful to configure it to run using a specific user account. The 'OPC enumeration' EXE is also installed as a service. In this section, the 'OPC enumeration' EXE will be used as an example, but the same method applies when the OPC server settings are changed.

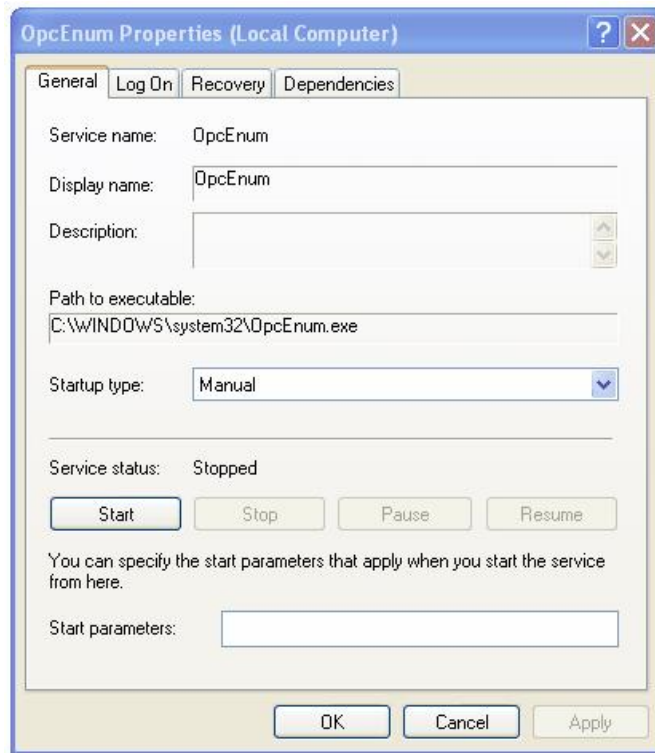
To open the 'services' page, select 'Start' -> 'Run', enter 'Services.msc' in the dialog that appears and click 'OK'.



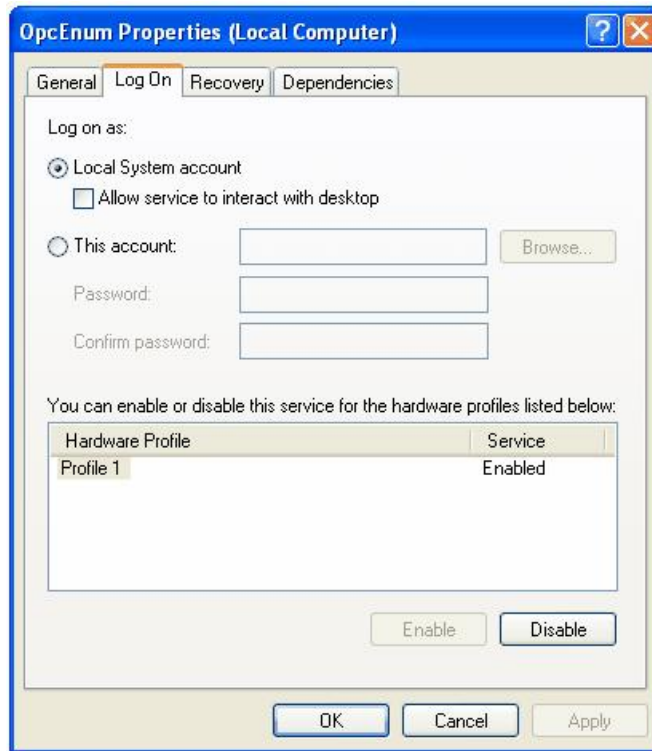
Find the service to be changed in the list. The sample screen below shows the OPC enumeration service. You may also need to change the settings for the 'Mitsubishi Electric MX OPC server 6.05' service.



Right click on the service name and select 'Properties'. The service configuration screen (below) will be shown.



Switch to the 'Log On' tab, select 'This account' and enter the username and password of the user to run as.



Now select 'OK' and close the services window.

## 15 Appendix: Using XML-DA

The XML-DA web service wrapper for OPC servers provides an interface to OPC servers using modern web technologies such as SOAP and XML. These technologies make it easier to access OPC data on other operating systems which have limited support for Windows or COM. The implementation as a 'wrapper' means that existing OPC servers can be enabled for web service access without any modifications to the software.

When connecting to the MX OPC server from a client application (such as a SCADA package or a Visual Basic program), it is not normally necessary to use XML-DA. Often the server and client components will be on the same computer anyway, or there will be a better way to pass the data from the OPC server across the network, such as using an Ethernet connection to the PLC, or passing the data between computers using a SCADA package's own built-in networking. However, there are some situations where XML-DA (like DCOM) is a suitable way to pass data from the OPC server between PCs, such as reading data from a Visual Basic or Visual C# program on another computer when there is only a single serial link connection available to the PLC. XML-DA is easier to configure than DCOM, and unlike DCOM it can be used to pass data across firewalls which use network address translation (NAT). However, it is more difficult to restrict user access to an XML-DA server.

This section will explain how to configure XML-DA with MX OPC Server. It is assumed that you are familiar with some COM and OPC terminology, and with the use of Internet Explorer and Microsoft Windows.

## 15.1 Security considerations

Unlike DCOM, XML-DA relies on the security settings on the server PC only. This means that it does not have the same issues when connecting between domains, or between computers running different operating systems. The disadvantage is that it is harder to check user credentials before allowing access to the OPC server.

XML-DA can be used to read data over an internet connection or across a network. Any computer which can reach the server PC using TCP/IP can connect to XML-DA if the firewall is set up to permit this. However, it is still not advisable to allow external internet traffic to access the XML-DA port on the server PC, as any newly-discovered vulnerability in either Internet Information Server (IIS) or XML-DA itself could allow an attacker to remotely control the server computer. Instead, if you need to connect to OPC-DA from another site using an internet connection, the safest way is to set up a virtual private network (VPN) to pass encrypted TCP/IP traffic between the two sites, and use this protected connection to connect to the XML-DA server.

Before setting up XML-DA in a corporate environment, you will need to discuss the changes to the security configuration with your network administrator or IT department, to make sure that the settings are consistent with your company's security policy.

In these instructions it is assumed that XML-DA and the web server will be installed on the same computer as the OPC server. If this is not the case and DCOM is used to connect from the OPC XML-DA wrapper to the server, you will need to set up DCOM security to allow a connection to the OPC server – see "[DCOM configuration](#)"<sup>[420]</sup> for instructions on how to do this. This will also affect your choice of the user account which is used to access the OPC server.

Some typical security options for XML-DA are shown in the following table.

Option	Notes
1.	<p><i>Fixed user name and password in the Web.config file within the virtual directory</i></p> <p>This method accesses the OPC server under an account with a fixed username and password, which are entered in plain text in the 'Web.config' file in the wrapper directory (described in more detail later in <a href="#">"Configuring the XML-DA wrapper"</a><sup>[493]</sup>). This is easy to set up, but means that the plain text password can be read by anyone with local access to the computer. With some extra configuration, the password can be hidden in a partially encrypted form in the registry, but it is still recoverable in a readable form by a determined attacker.</p> <p>The user account selected for this must have sufficient COM or DCOM access rights to use the OPC server. You may also find that the policy settings on the computer prevent you from using an account with a blank password.</p> <p>If you are using this option, it may be sensible to create a new user account especially for this purpose and assign it COM access rights to the MX OPC server (described later in section <a href="#">"COM configuration"</a><sup>[505]</sup>). You may also wish to prevent the account being used to log on locally (i.e. when the computer is started). To do this on a PC which is not in a domain, select 'Start -&gt; Run', type 'gpedit.msc', then click 'OK' to start the group policy editor. In the tree view, browse to 'Local computer policy -&gt; Computer configuration -&gt; Windows settings -&gt; Security settings -&gt; Local policies -&gt; User rights assignment', and in the 'Policy' list in the right-hand pane, double click on 'Deny logon locally'. Use the dialog that appears to add the new user account name to the list.</p>
2.	<p><i>Give OPC server access rights to the ASPNET user</i></p> <p>By default, the web service will run under an account created by Internet Information server called 'ASPNET'. This account can be given COM access rights to the MX OPC server, which removes the need to use a separate account and plain text password.</p> <p>If you have already configured DCOM (described in <a href="#">"DCOM configuration"</a><sup>[420]</sup>) and have assigned OPC access rights to a new group, you may be able to add the ASPNET account to this group. Alternatively, you can assign MX OPC server access rights to the ASPNET user independently (described later).</p> <p>The main risk with this option is that a badly-secured ASP.NET application running as the ASPNET user in the same instance of Internet Information Server may be easier for an attacker to exploit if it has the additional access rights needed by COM or DCOM.</p>

## 15.2 Before you begin

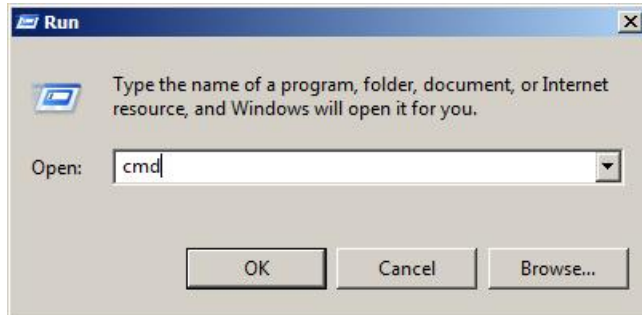
Before you start to modify the PC configuration, please ensure that:

- The server (and optionally, client) PCs are up to date with service packs and security fixes (e.g. service pack 4 for Windows 2000, and service pack 3 for Windows XP). Applying the latest security fixes will reduce the risks introduced when the security configuration is changed.
- You are logged on to the PC(s) using a local administrator account
- If you have not already installed Internet Information Server (IIS), you have your original operating system installation media available as you may be prompted for additional files.
- If you will be testing the XML-DA connection from another PC or using DCOM, that there is a working TCP/IP network connection between the client and server PCs. One way to test this is by using the 'ping' command:

1) Open a command prompt as follows:

### Windows XP or Windows 2000

Select 'Start' -> 'Run', enter 'cmd' in the dialog that appears and click 'OK'.



### Windows 7, Windows Vista or Windows Server 2008 R2

Click the Windows icon or start button, then enter 'cmd' in the search box and press enter.





- 2) In the command window that appears, enter 'ping <otherpc>', where <otherpc> is the name of the other computer e.g. 'ping SCADAserver'. A computer's network name can normally be found by right-clicking on 'My computer', selecting 'properties' then clicking on the 'Computer name' tab. If in doubt, ask your network administrator for assistance.
  - 3) After pressing return, you should normally see a message such as 'Pinging <address> with 32 bytes of data' (where <address> is the IP address such of the other PC, for example 192.168.200.100) followed by four response messages from the other computer. If no address is shown, there may be a problem in finding the other computer by name – ask your network administrator to check the PCs settings. If the address is found but there are no responses, this may mean that the connection is not working, or it can sometimes indicate that a third-party firewall on one of the computers is preventing the ping request from being sent or the ping response from being received.
- If you are developing the client application yourself, make sure that you have applied the most recent service packs to your development environment.
  - If you need to modify the DCOM or COM settings, make a note of the previous settings in case the changes have to be reversed. Some of the changes will affect all applications using DCOM on the computer, not just the MX OPC server, and this could stop other applications (such as other OPC servers) from working.

## 15.3 Configuring XML-DA

The configuration of XML-DA depends on the operating system used. Where there are significant differences between operating systems, this will be made clear in the directions.

NOTE: If you install Internet Information Server (IIS) after the .NET framework, ASP.NET pages will not work correctly. Follow the instructions in ["Adding ASP.NET support to a new installation of Internet Information Server"](#) [489] to fix this. If you have not yet installed either package, install IIS first to prevent the issue from occurring.

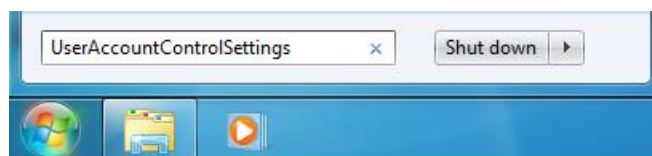
### 15.3.1 Turning off user account control

If you are installing on a later operating system which has user account control (UAC) such as Windows Vista, Windows 7 or Windows Server 2008, you may experience problems installing the XML-DA wrapper, even if you are logged in as a user with Administrator rights.

If any of the installers do not run correctly, it may be necessary to temporarily turn off user account control during the installation, and restore it afterwards. The instructions below explain how to turn UAC on or off, for each operating system. Note that it is usually necessary to reboot after changing the UAC settings.

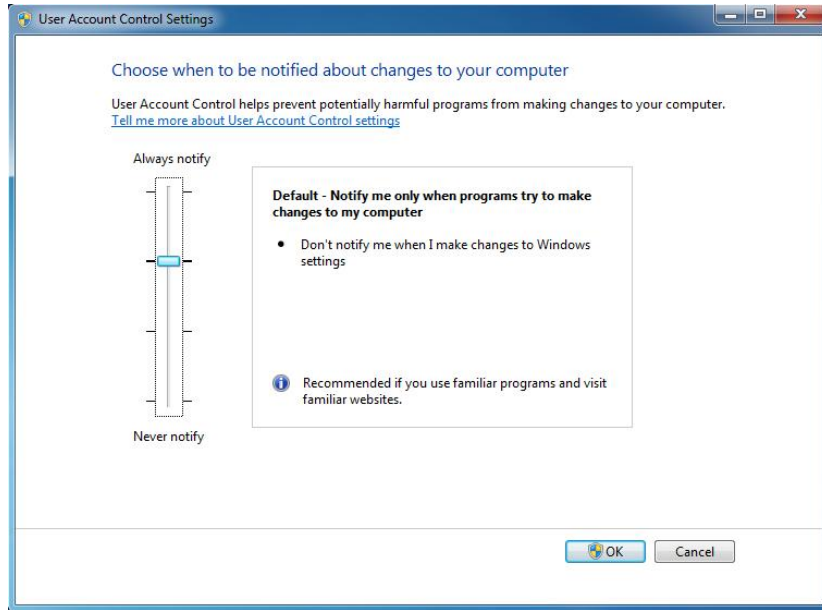
#### Windows 7 and Windows Server 2008

Click on the windows logo (or start button) and enter 'UserAccountControlSettings' in the search box that appears, before pressing enter.



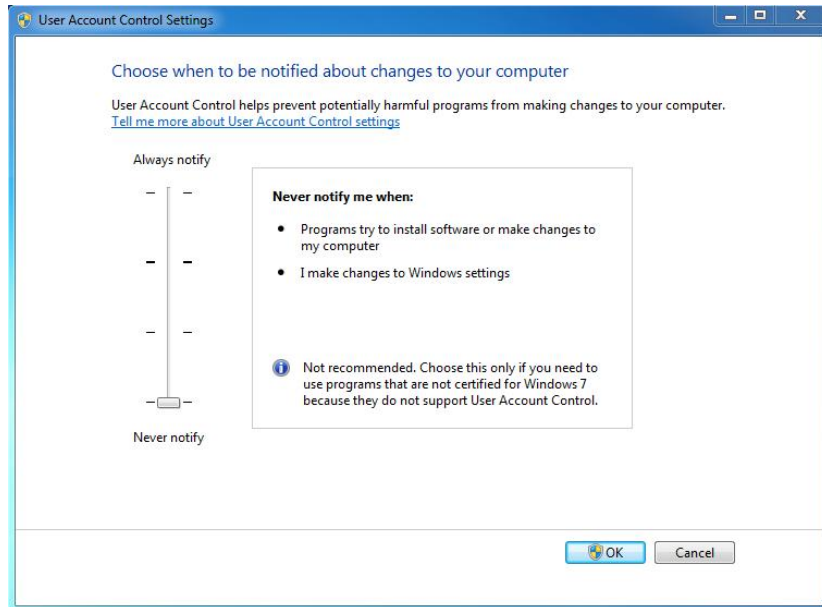
Open user account control

The slider will normally start on the 'Default' setting (as in the picture below). With this setting, UAC prompts will be shown as normal.



Default UAC setting

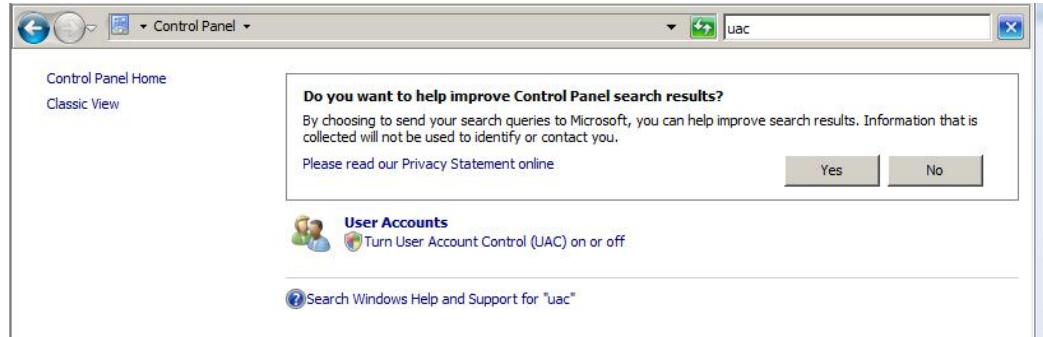
To disable UAC, drag the slider to 'Never' as shown below. This will prevent UAC prompts from appearing. After clicking 'OK' to confirm the setting, you will be prompted to restart the computer.



UAC switched off

### Windows Vista

Select 'Start' -> 'Control panel'. In the search box at the top right of the control panel window, type 'UAC'. The search results should be updated to show the 'Turn User Account Control (UAC) on or off' link as in the image below:



Control panel search for UAC

Click on 'Turn User Account Control (UAC) on or off'. In the window that appears, uncheck the box to disable UAC, or re-check the box to enable UAC. After pressing 'OK' to confirm the setting, you will be prompted to restart the computer.



Turn UAC on or off

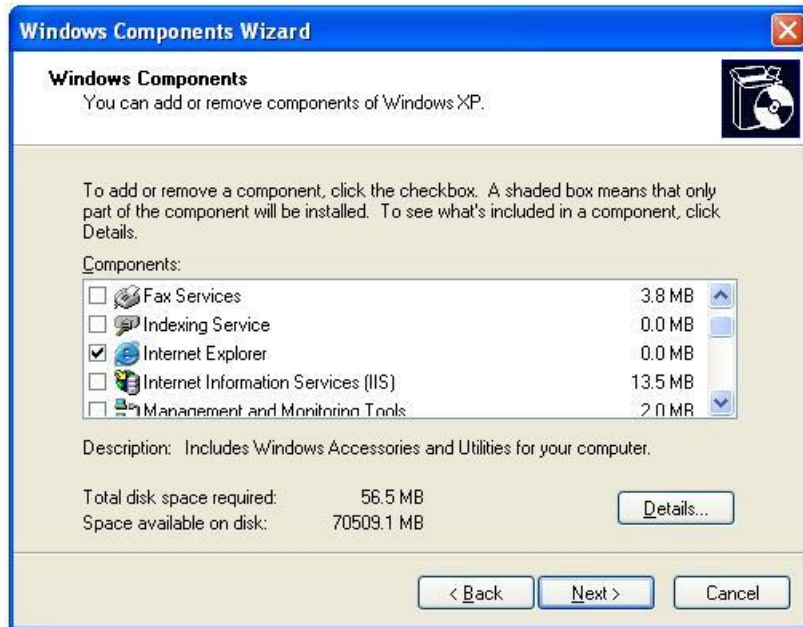
### 15.3.2 Installing Internet Information Server (IIS)

XML-DA requires Microsoft Internet Information Server (IIS) to operate. Although this is supplied with Windows 2000 and most editions of later operating systems, it is not always installed by default. IIS is not supported for use with every edition of Windows - for example, it is not available on Windows XP Home edition.

The method to install Internet Information server depends on whether your operating system is [Windows XP or 2000](#)<sup>[480]</sup>, [Windows Server 2008 R2](#)<sup>[482]</sup> or other [later operating systems](#)<sup>[481]</sup>.

### 15.3.2.1 Installing Internet Information Server on Windows XP

To find out whether or not IIS is installed, select 'Start -> [Settings] > Control panel -> Add or remove programs'. In the left side of the window that appears, select the third item 'Add / remove windows components'. The window should look similar to the picture below:



**Add/remove windows components**

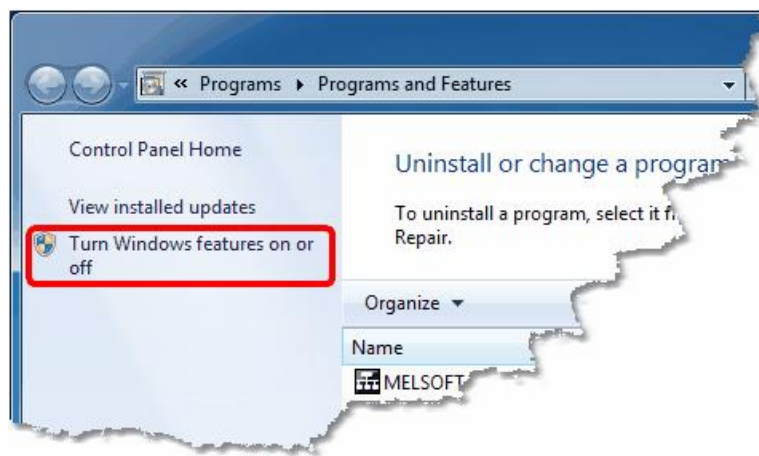
Find 'Internet Information Services (IIS)' in the list. If the box to the left is ticked, it has already been installed, in which case you can click 'Cancel' and go on to the next section. Otherwise, click on the empty box next to 'Internet Information Services (IIS)' to select it for installation. Select 'Details' if you want to modify the parts of Internet Information Server that are installed (although the defaults are acceptable for XML-DA). When you are ready, click 'Next' and Internet Information Server will be installed. You may be prompted for the original operating system CD or DVD media if there are any files required that are not already on the computer.

### 15.3.2.2 Installing Internet Information Server (Vista or Windows 7)

To install Internet Information Service on Windows 7, Windows Vista and later operating systems (other than server operating systems), start by clicking on the Windows icon (or 'Start' button) then type 'appwiz.cpl' in the search box before pressing enter.

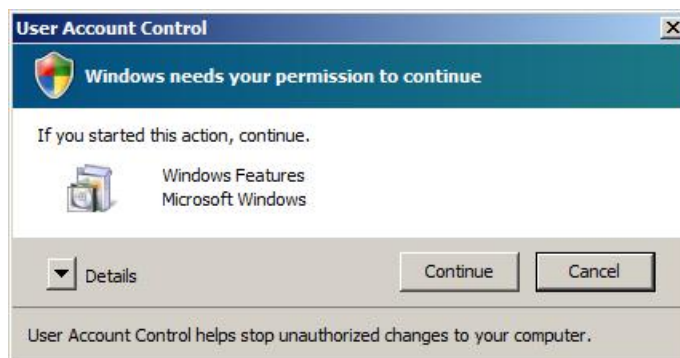


In the window that appears, click 'Turn Windows features on or off' (as shown below).



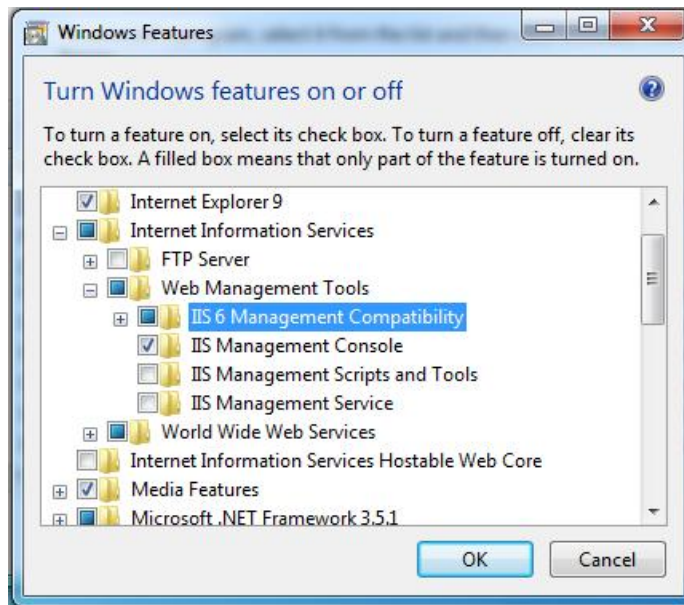
**Turn Windows features on or off**

Depending on your operating system, you may see a user account control prompt like the one below. Click 'Continue' or 'Yes'.



**Windows features UAC prompt**

The windows features page is shown below. Find 'Internet Information Services' in the list. Ensure that the box to the left is ticked, then expand the item to find 'Web Management Tools' / 'IIS 6 Management Compatibility' and check the box next to it as well. When you are ready, click 'OK' and Internet Information Server will be installed.



Enable internet information server

**15.3.2.3 Installing Internet Information Server (Windows Server 2008)**

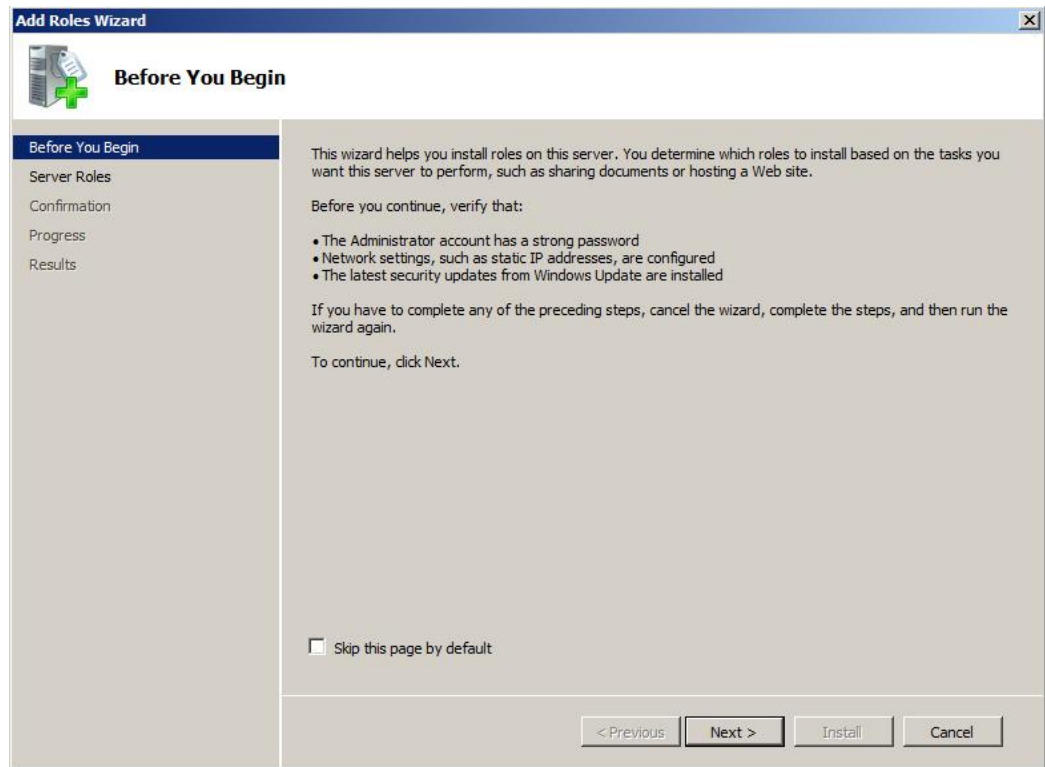
To install Internet Information Server on Windows Server 2008 R2, start by clicking on the Windows icon (or 'Start' button) then type 'ServerManager.msc' in the search box before pressing enter.



When the 'Server Manager' window appears, right click on 'Roles' then select 'Add Roles' (as shown below).

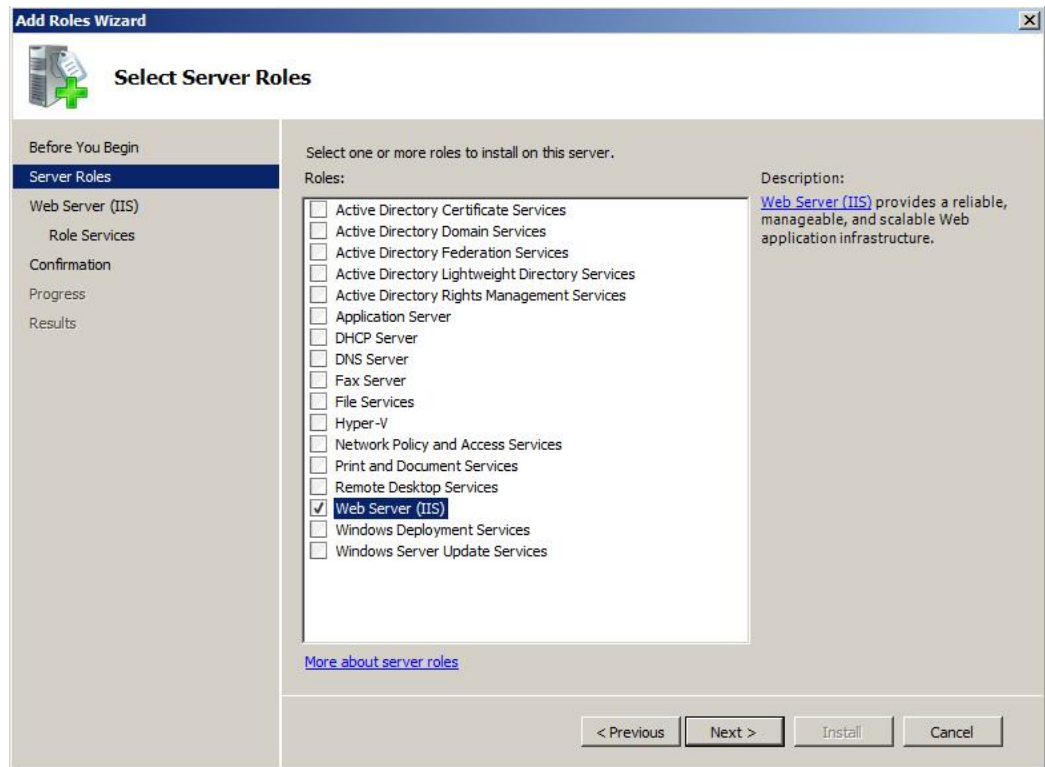


Click 'Next' to skip the introduction screen (pictured below).



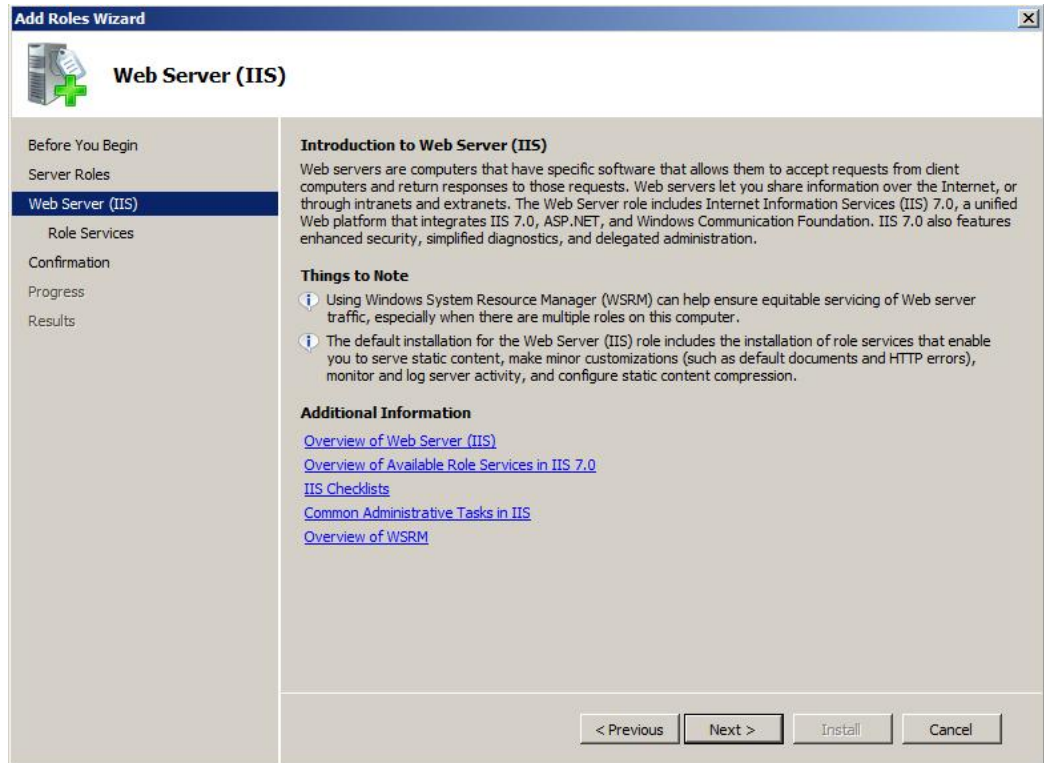


In the 'Server Roles' list, check 'Web Server (IIS)' then click 'Next'.

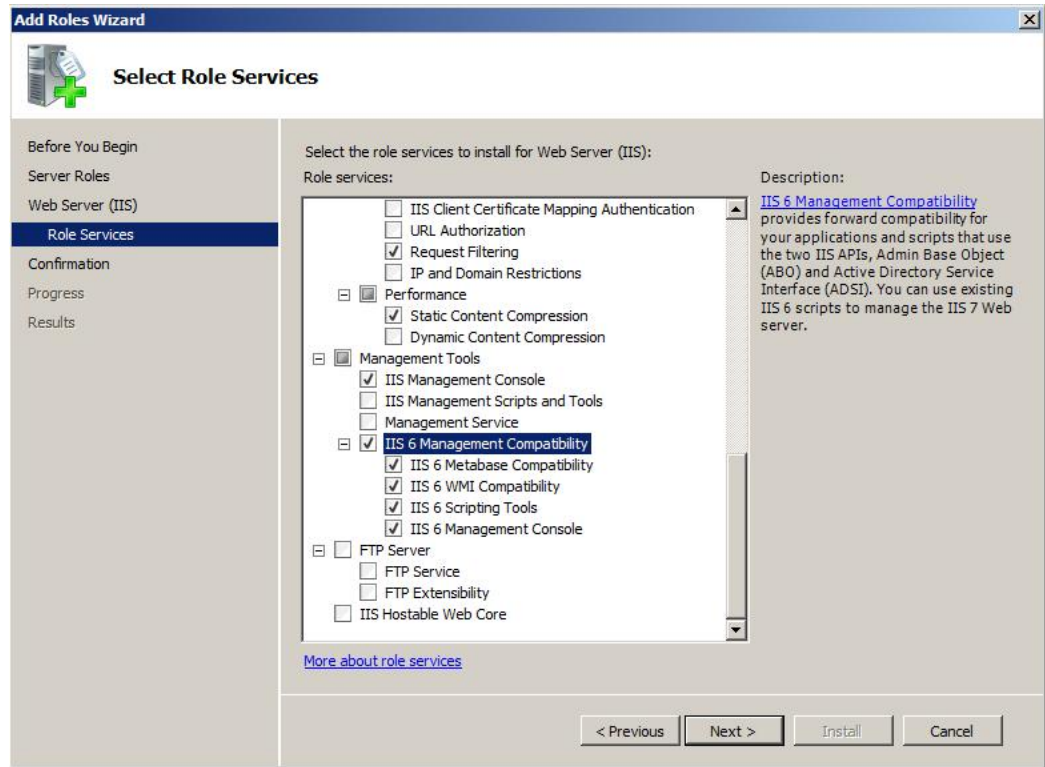




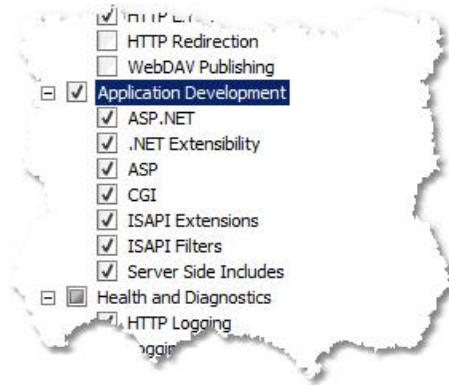
Read through the instructions on the following page (shown below) then click 'Next'.



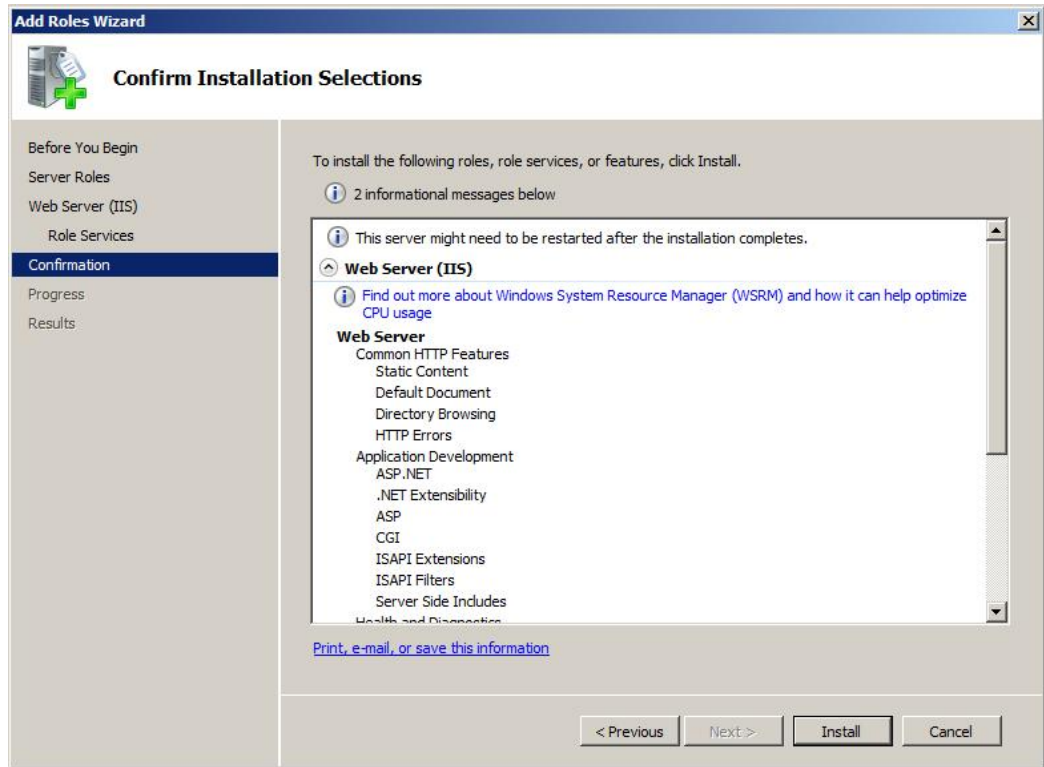
On the 'Role Services' page, scroll the list down to find 'IIS 6 Management Compatibility' under 'Management tools', and check the box next to it:



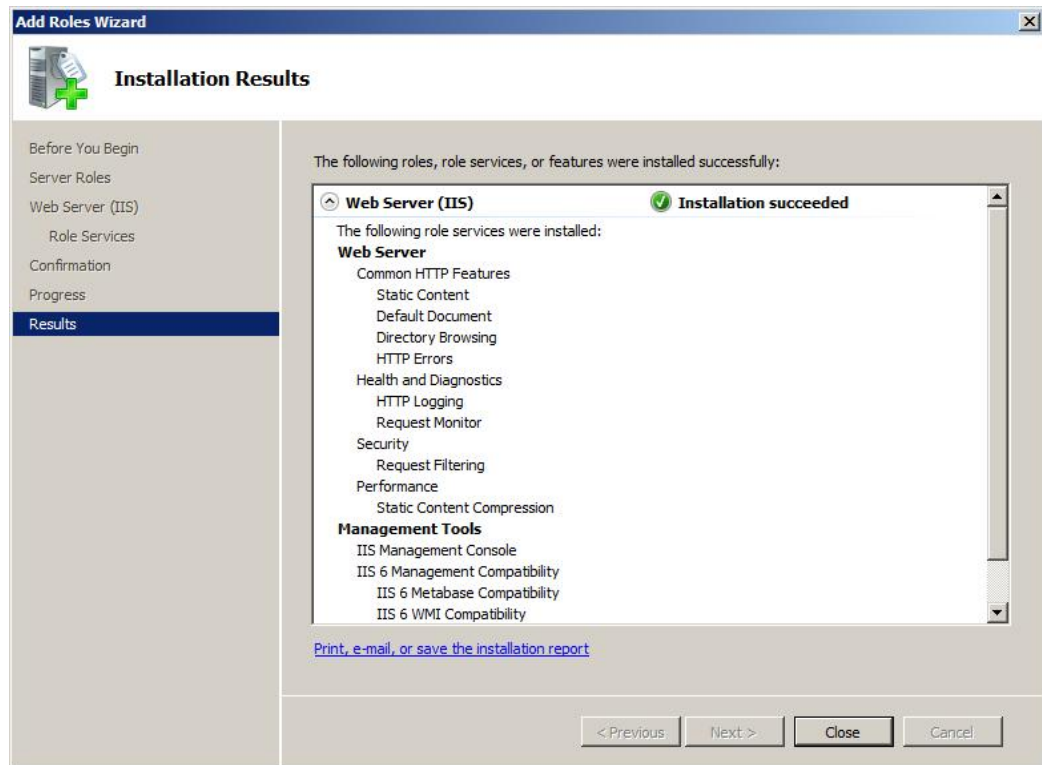
The 'Application development' items should also be enabled (as shown below), before clicking 'Next'.



The confirmation page below will be shown. Click 'Install'.



When the installation results page appears, click 'Close'.



### 15.3.3 Installing Microsoft .NET framework v1.1

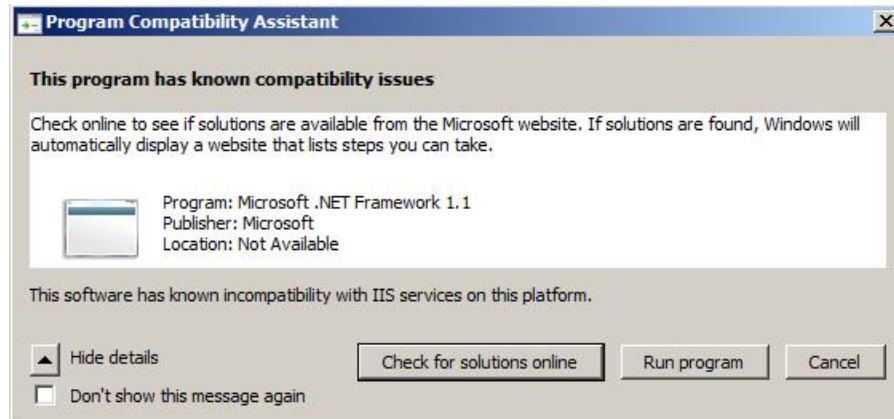
The XML-DA wrapper web service requires Microsoft .NET framework v1.1 to run. To determine whether or not this is installed on your computer, select 'Start -> Control panel -> Add/remove programs' (or 'Start -> Control panel -> Programs / Uninstall a program') and see if 'Microsoft .NET framework v1.1' appears in the list. If the .NET framework was already installed before you installed IIS, you may need to follow the instructions in the next section before ASP.NET pages will work correctly.

If the .NET framework v1.1 was not installed, download the installer from the Microsoft web site at [www.microsoft.com/downloads](http://www.microsoft.com/downloads). By searching for ".NET framework 1.1" you should be able to find the two items:

- .NET framework v1.1 redistributable package (filename 'dotnetfx.exe')
- .NET framework v1.1 service pack 1 (filename 'NDP1.1sp1-KB867460-X86.exe')

Download both of these, and install the redistributable package first, followed by the service pack. After installing both these items, there will be some remaining security fixes, which can be installed through Windows update (see the next section).

When the .NET framework v1.1 is installed on Windows Server 2008 R2, you will see a warning like the one below. Select 'Run program'.



**Compatibility warning on Windows Server 2008**

### 15.3.3.1 Adding ASP.NET support to a new installation of Internet Information Server

If Internet Information Server was installed after the .NET framework, you will need to register ASP.NET with Internet information server using the 'aspnet\_regiis.exe' file located in the .NET framework directory.

On Windows XP or earlier, select 'Start -> Run' and enter the following carefully before clicking 'OK':

```
%WINDIR%\Microsoft.NET\Framework\v1.1.4322\aspnet_regiis.exe -i
```

For Vista or later operating systems, click on the Windows icon or 'Start' button, then enter the same text in the search box before pressing Ctrl+Shift+Enter. You may need to answer 'Yes' or 'Allow' if a user access control prompt appears.

If you will also be using ASP.NET from other versions of the .NET framework later, you may wish to register those versions as well. For example, to register the current .NET 2.0 framework, the command to use is:

```
%WINDIR%\Microsoft.NET\Framework\v2.0.50727\aspnet_regiis.exe -i
```

### 15.3.4 Updating system components

If you installed Internet Information Server (IIS) or the Microsoft .NET framework in the previous steps, it is advisable to update your computer to ensure that it has the latest security patches, as there may have been security fixes developed since the installation media were created. If the computer has an internet connection, the easiest way to do this is to use Windows Update.

#### A) If your Computer is configured for automatic Windows updates

If you already have automatic updates enabled, you can trigger an update by selecting 'Start->Run' and entering 'wuauclt /DETECTNOW' (or on later operating systems, clicking the Windows icon or 'Start' button and typing the same text into the search box). Depending on how the automatic updates are configured, you will either be prompted to install the new updates and to restart the computer afterwards, or the installation may be performed the next time you restart the computer.

#### B) If your computer is not configured for automatic Windows updates

To update the computer manually, either (depending on your operating system):

- select Start -> [All] Programs -> Windows update (or Microsoft update)
- start Internet explorer and select 'Tools -> Windows Update'

then follow the prompts and select any patches related to Internet Information Server for installation. You may need to restart your computer after the patches are installed.

For the .NET 1.1 framework in particular, you may find that you have to repeat this step until no more updates are found – the updates are not included in a single package.

### 15.3.5 Installing the XML-DA wrapper

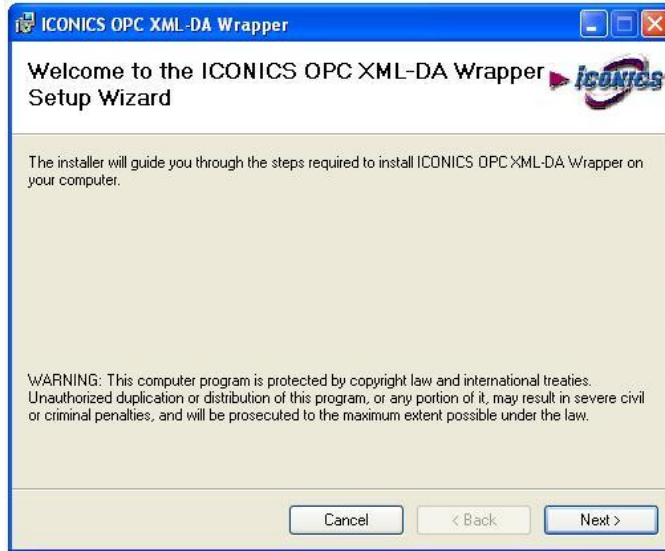
To install the wrapper web service, run the supplied installer 'XMLDAwrapperSetup.msi'. Note that on later operating systems such as Windows 7, Windows Vista and Windows Server 2008, you may need to [turn off user account control](#)<sup>[477]</sup> temporarily for the installation to complete successfully.

If you see the message below, the .NET framework v1.1 has not been correctly installed - see "[Installing Microsoft.NET framework v1.1](#)"<sup>[488]</sup> for instructions on how to do this.

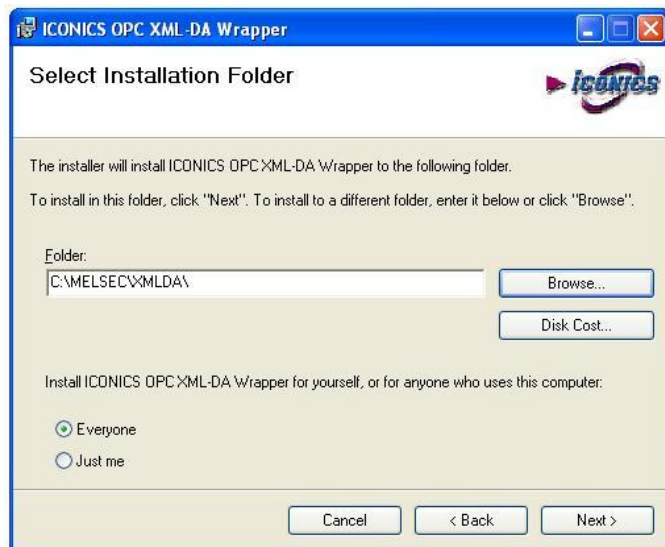


**.NET framework required prompt**

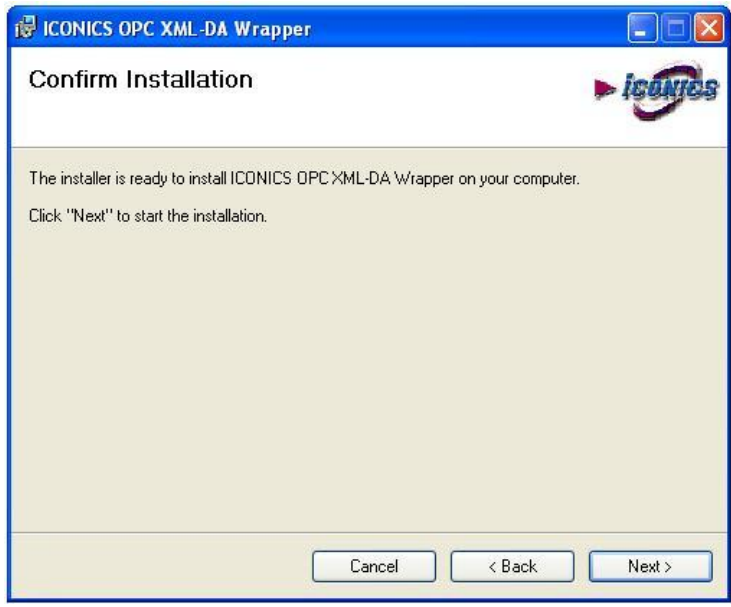
Once the .NET framework v1.1 is installed, the installer will go on to the 'Welcome' page (below). Click 'Next' to continue.



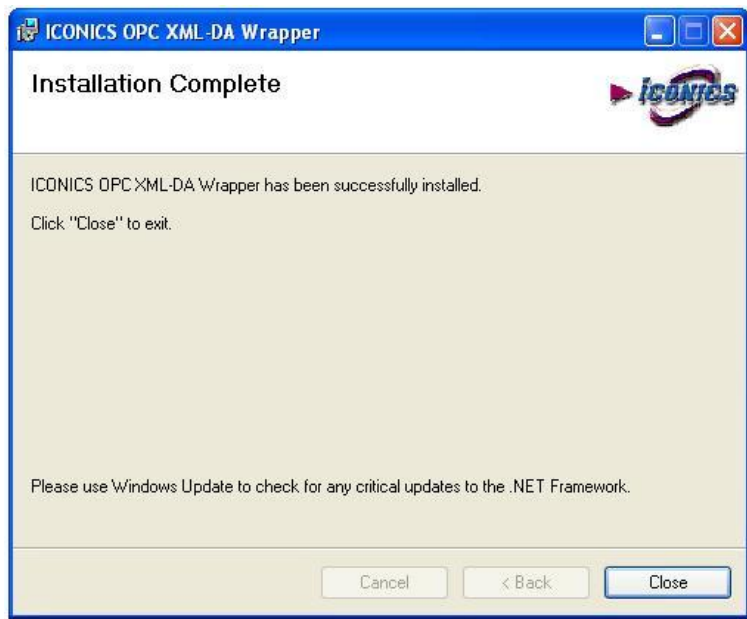
You will now be prompted for the location where the wrapper should be installed. For this example, the wrapper will be installed to 'C:\MELSEC\XMLDA' (see picture below). You may optionally select 'Everyone' to allow other users to access the files. Select 'Next' to continue.



At the 'Confirm installation' prompt, select 'Next'.



When installation is complete, you will see the dialog below and the readme file will be displayed. Click 'Close'.

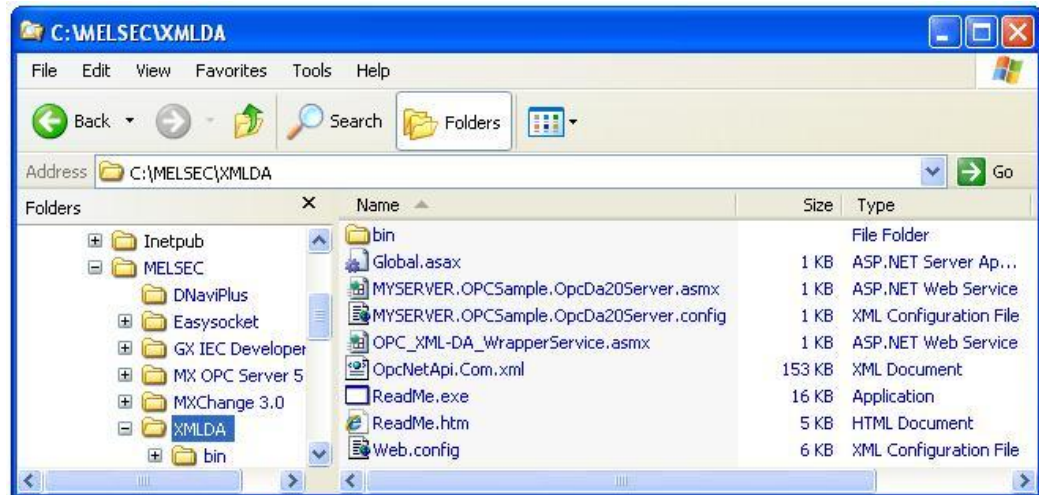




### 15.3.6 Configuring the XML-DA wrapper

The next step is to set up a web service for MX OPC server.

Browse to the directory where the wrapper was installed (in the above example, this was 'C:\Melsec\XMLDA'). The directory contents should be similar to the following example:



Take a copy of the file 'OPC\_XML-DA\_WrapperService.asmx', and rename this copy as 'Mitsubishi.MXOPC.6.asmx' – this is the 'ProgID' (human-readable name) of the current MX OPC server version. Other OPC servers can be added in the same way by using the 'ProgID' of the server with the '.asmx' extension. If no config file is supplied, the name of the file is used to find the OPC server to read from, which is assumed to be on the local PC.

An alternative method of configuration is demonstrated by the 'MYSERVER.OpcDa20Server.asmx' and 'MYSERVER.OpcDa20Server.config' files. In this case, a '.config' file is supplied containing the PC and OPC server names in the 'Node' and 'ProgID' entries respectively. This requires that DCOM is configured correctly to allow connections to the PC where the OPC server is running – see the ["DCOM configuration"](#)<sup>[420]</sup> for instructions on how to do this.

The 'Web.config' file is a text file can be edited with e.g. 'Notepad' (Select Start -> Run and enter 'notepad.exe' then click OK and browse to the file) to configure the account that is used to access the OPC server. Find the lines shown below in the file:

```
<!-- IDENTITY
This section allows the web service to impersonate a specific Windows user account that
determines what privileges the web service has. This is important for the XML-DA sample
web service since it must be able to launch as access local COM servers. The two sample
COM servers included with the distribution are configured to allow 'Everyone' access which
means that no special web service configuration is required. Changes to this section are
required before the XML-DA sample server can connect to other COM servers.

There are two possibilities for determining identity: the web service can use an account
that is explicitly specified in this file or it can use the account identified by IIS
integrated windows authentication, If the later approach is used then there must be no entry
in this file for 'userName/password' and the virtual directory for the webservice must
have anonymous access must be disabled (which forces the client to explicitly login).
-->

<!-- <identity impersonate="true" userName="[username]" password="[password]"/> -->
```

Whether this needs to be changed or not depends on your chosen security option from the table in ["Security considerations"](#)<sup>[474]</sup>. By default, the last line of the section above is commented out (with '<!--' and '-->' markers at the start and end of the line. With this line disabled, Internet information server will run the web service using its built-in 'ASPNET' account. This is the correct method if you are using security option 2 ('assign rights to the ASPNET user'), in which case you can go on to the next section.

The alternative is to select a fixed user name and password for the account to use – this is the correct method for security option 1 ('fixed user name and password in web.config'). To do this, copy the line and paste a copy of it on the line below. Remove the '<!--' and '-->' comment markers at the start and end of the new line, and set a username and password. The resulting lines for username 'xmluser' and password 'opensesame' would look like this:

```
<!-- <identity impersonate="true" userName="[username]" password="[password]"/> -->  
<identity impersonate="true" userName="xmluser" password="opensesame"/>
```

**WARNING:** Do not re-use a password that you use on other systems and wish to keep secret, as it is stored in readable text form and can be seen by any other user with access to the computer.

After modifying the file, save the changes and close notepad.

### 15.3.7 Configuring Internet Information Server (IIS)

The configuration of Internet Information Server depends on the operating system and IIS versions.

Depending on your operating system, either following the instructions for [Windows XP and Windows 2000](#)<sup>[495]</sup>, or [later operating systems](#)<sup>[499]</sup>.

### 15.3.7.1 Configuring Internet Information Server (IIS) on Windows XP

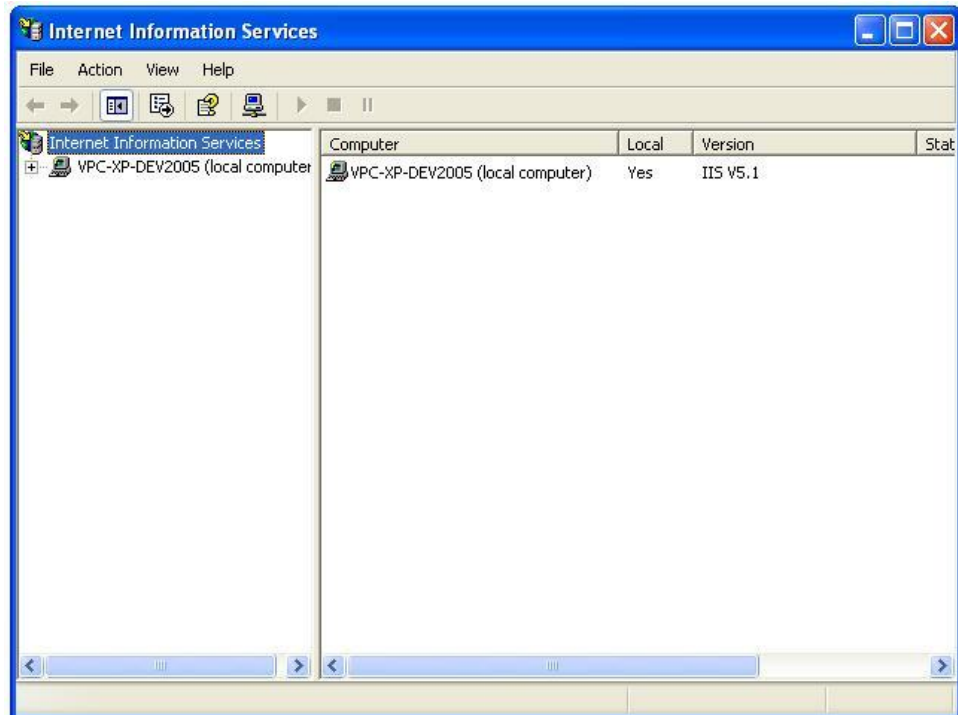
To use XML-DA, a new 'Virtual directory' must be configured in IIS. First, run the IIS configuration tool from:

#### Windows XP

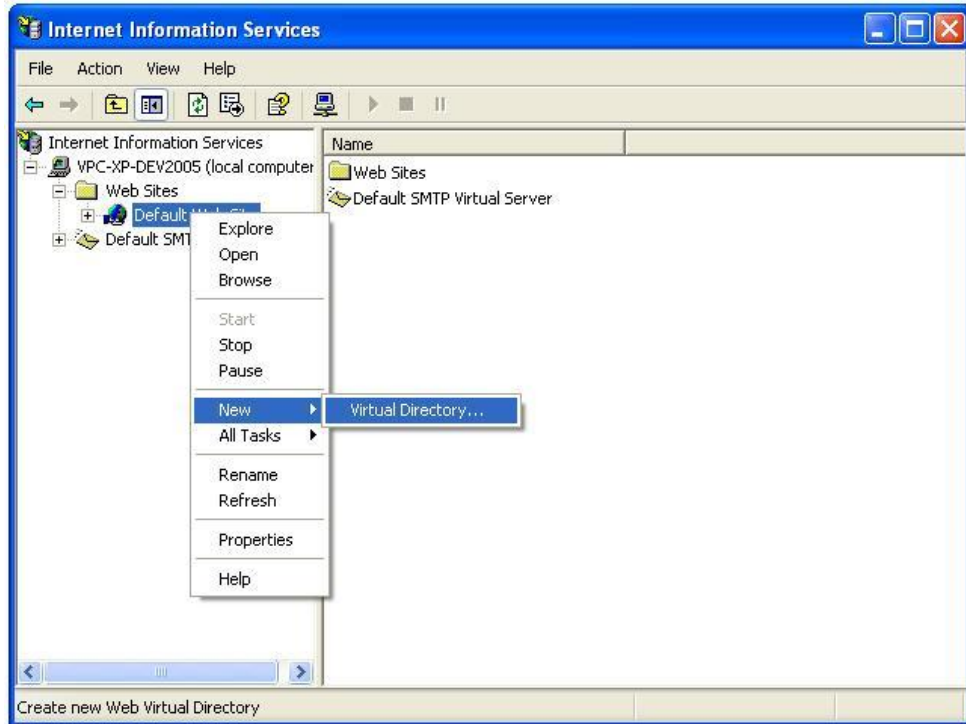
Start -> Control Panel -> [Performance and Maintenance ->] Administrative Tools -> Internet Information Services

The 'Performance and maintenance' part will only be present if the control panel is set up to display in 'Category view' instead of 'Classic view'.

The configuration window should appear.



Expand the tree control in the left-hand pane until 'Default web site' is shown, then right click on 'Default web site' and select 'New->Virtual directory' from the context menu.



The 'Virtual directory creation wizard' will be shown (see below). Select 'Next'.

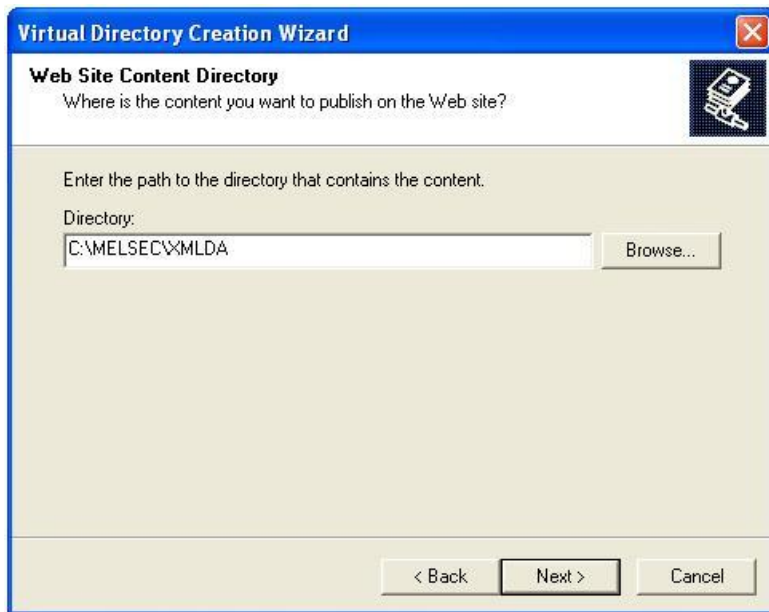


You will be prompted for the 'alias' name of the new virtual directory. This is the name that will be used to refer to the directory as part of the URL entered in the address bar in the web browser, for example http://computername/alias/...

In this example, we will use the name 'xmllda' as shown in the picture below.



After entering the name, click 'Next'. You will be prompted for the directory where the content is stored. Browse to the directory where you installed the OPC wrapper (see "[Installing the XML-DA wrapper](#)"<sup>[49b]</sup>), then click 'Next'.



The permissions screen will now be shown.



Ensure that 'Read', 'Run scripts (such as asp)' and 'Execute (such as ISAPI applications or CGI)' are all selected, then click 'Next'.



Select 'Finish' to exit the wizard.

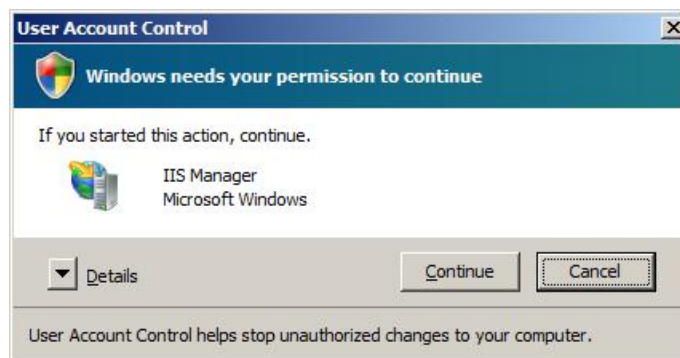
### 15.3.7.2 Configuring Internet Information Server (IIS) on later operating systems

On Windows Vista and later operating systems, there are some additional steps necessary to configure the security in IIS, before the application can be added.

Click the Windows icon (or 'Start' button) and enter 'inetmgr' before pressing enter:

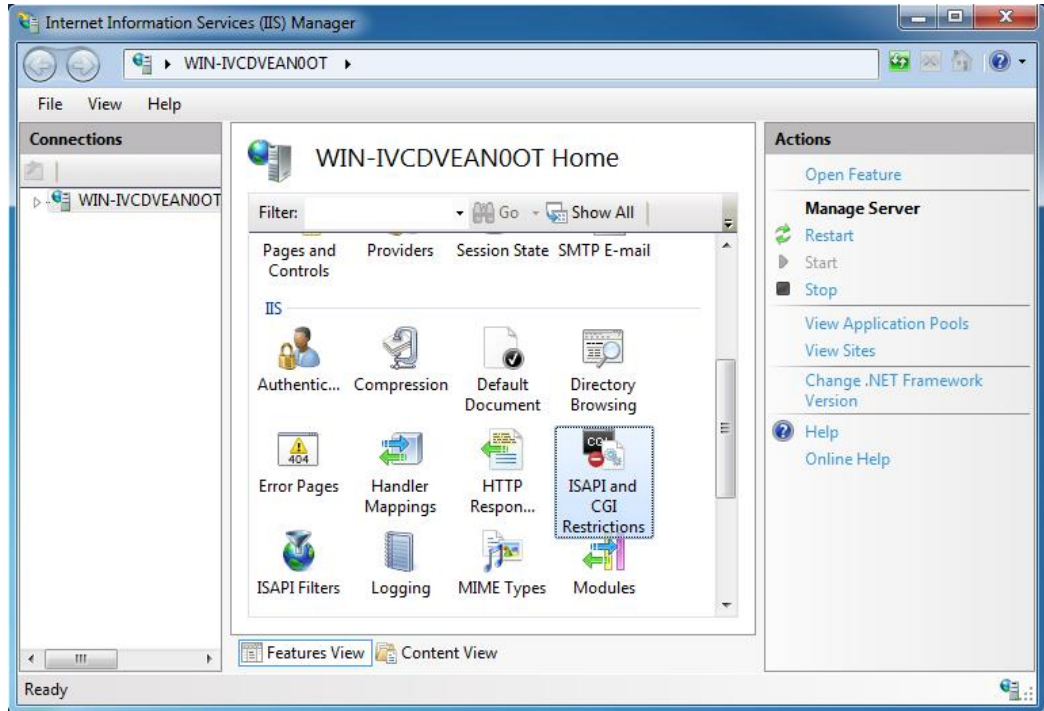


On Windows Vista, you may see a UAC prompt like the one below, in which case click 'OK' or 'Allow'.

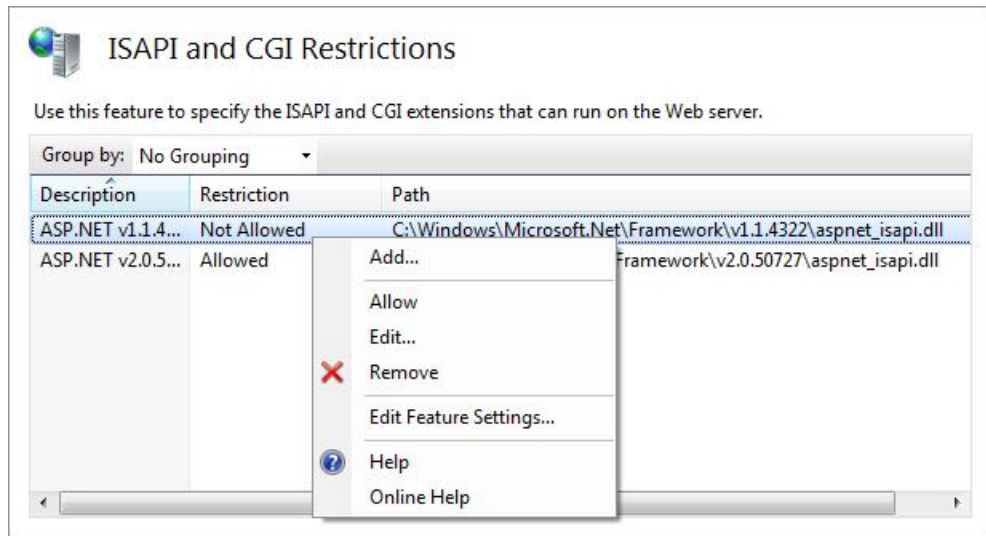




In the 'Internet Information Services (IIS) Manager' window, click on the local computer name in the 'Connections' tree at the top left, then find 'ISAPI and CGI restrictions' in the 'IIS' section and double-click it.

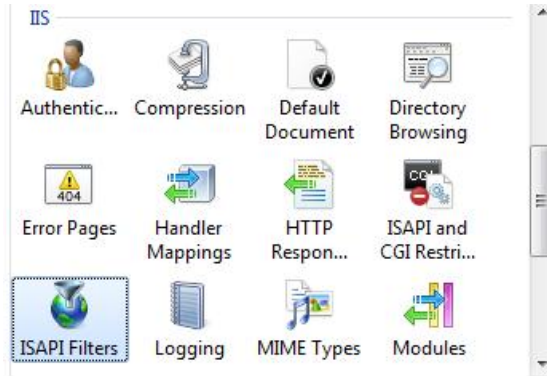


Find the entry for ASP.NET v1.1 in the list, right click it and select 'Allow':

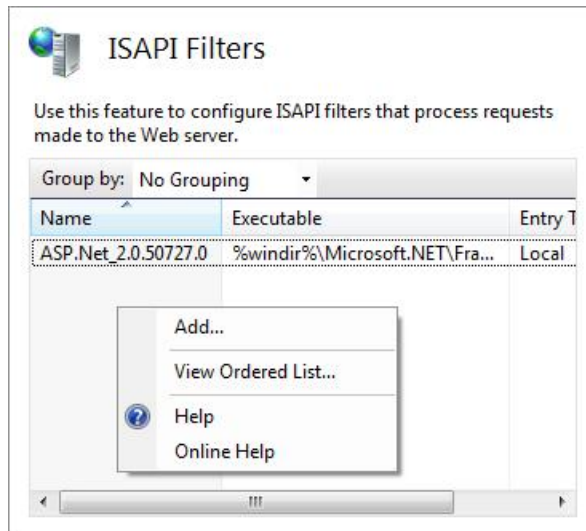




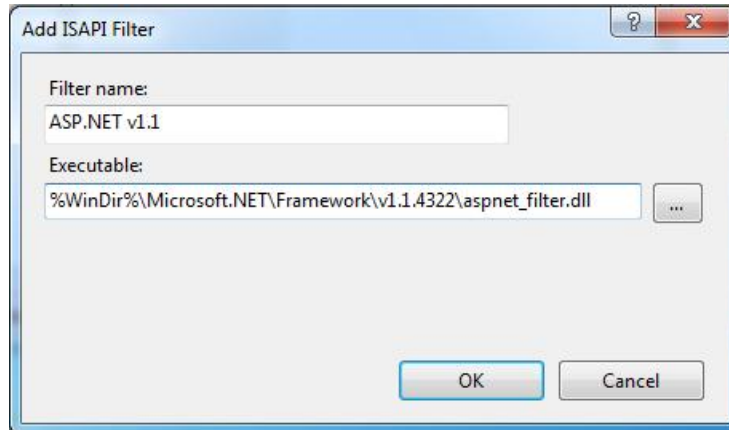
Return to the previous screen by clicking on the local PC name at the top level again, and this time double-click on 'ISAPI filters'.



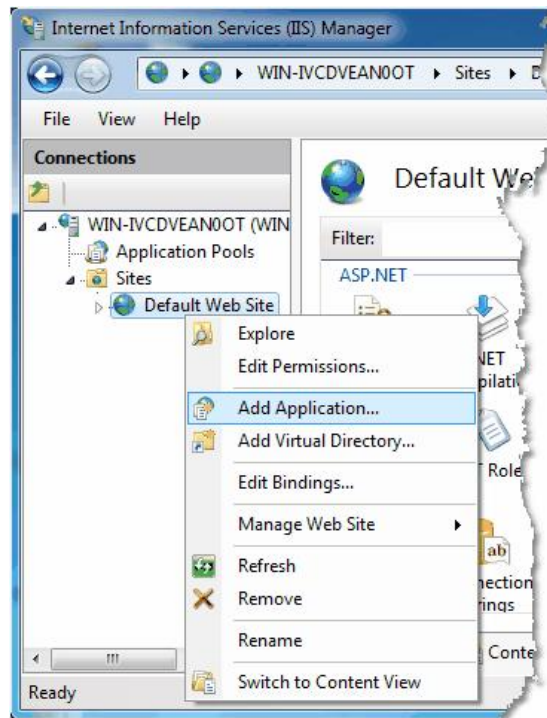
Right click on the background of the ISAPI Filters page, and select 'Add' from the context menu.



In the 'Add ISAPI Filters' dialog, enter a name e.g. 'ASP.NET v1.1' and path '%windir%\Microsoft.NET\Framework\v1.1.4322\aspnet\_filter.dll'.



After the security has been configured, a new application can be added. Return to the 'Internet Information Services (IIS) Manager' window, then expand the tree on the left side until 'Default Web Site' is visible, before right clicking on it and selecting 'Add Application...'.

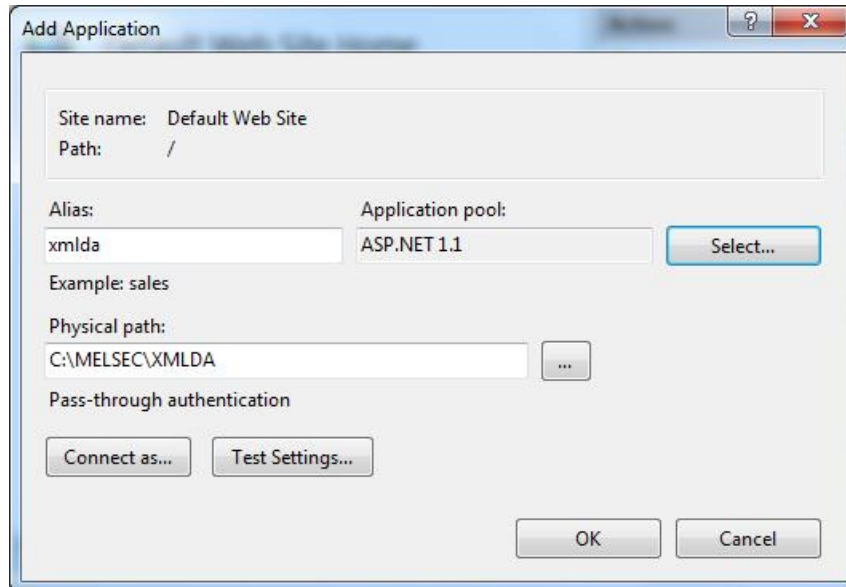


**Adding an application**

The 'Add Application' dialog will be shown (see below).

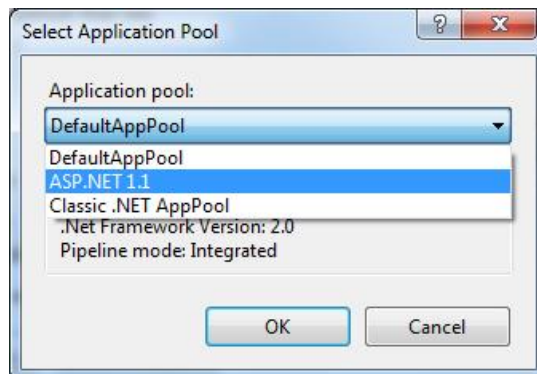
The 'Alias' name is the name that will be used to refer to the directory as part of the URL entered in the address bar in the web browser, e.g. 'http://computername/alias/'. In this example, we will use the name 'xmlda'.

The 'Physical path' is the directory where content is stored. Browse to the directory where you installed the OPC wrapper (see "[Installing the XML-DA wrapper](#)"<sup>[490]</sup>), then click 'Connect as'.

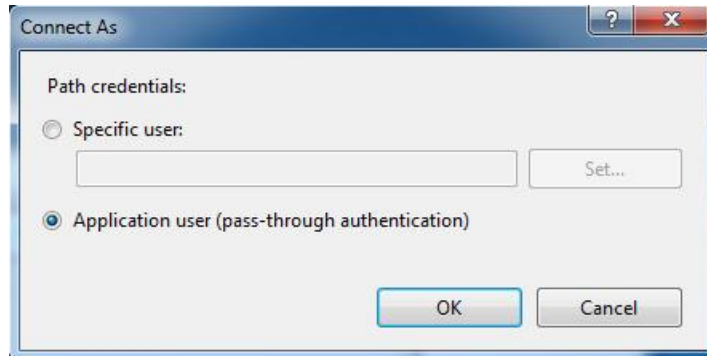


**Add application dialog**

The application pool must be changed to 'ASP.NET 1.1'. To do this, click on the 'Select' button to the right of the application pool, and select 'ASP.NET 1.1' from the list (see below) before clicking 'OK'.

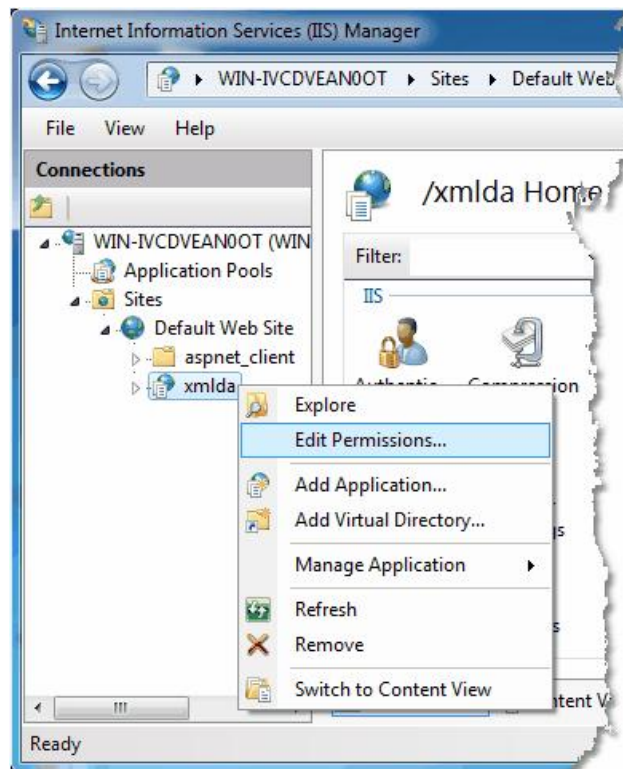


In the 'Connect as' settings on the 'Add application' dialog, you can select the user identity that will be used. Depending on your security configuration, this can either be a specific user account, or the application user. After selecting 'OK', the 'Test settings' button from the previous dialog can be used to check if the directory will be readable.



Click 'OK' to close the 'Add application' dialog and return to the main configuration page, where the new alias should appear as an item under the default web site.

You can optionally right click on the new alias, and select 'Edit permissions' from the context menu (as shown below) to view the standard folder properties dialog. This has a 'Security' tab where the folder permissions can be edited (e.g. to make sure that the IIS account used will be able to read the files).



IIS manager can now be closed.

### 15.3.8 COM configuration

If the user account you selected to run the OPC server has already been configured to have the correct COM access rights for MX OPC server (for example, through previously configuring a working DCOM connection with the same account), you can move on to the next section. Otherwise, you will need to configure COM security access for the user account (either the fixed user name and password you chose, or the ASPNET account).

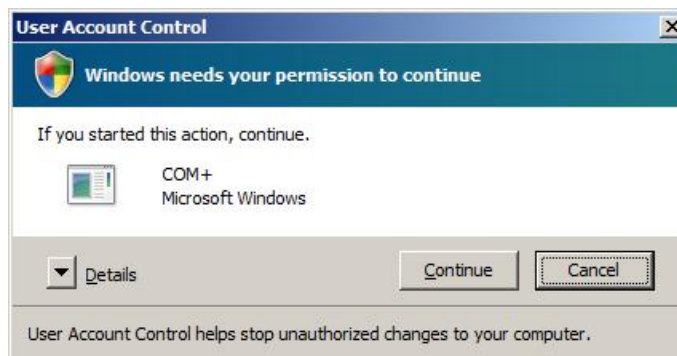
This section will describe how to do this. The instructions are different for [Windows Vista<sup>\[505\]</sup>](#) (or later) or [Windows XP service pack 2<sup>\[515\]</sup>](#), so follow the instructions in the section for your operating system.

#### 15.3.8.1 COM configuration on Windows Vista or later

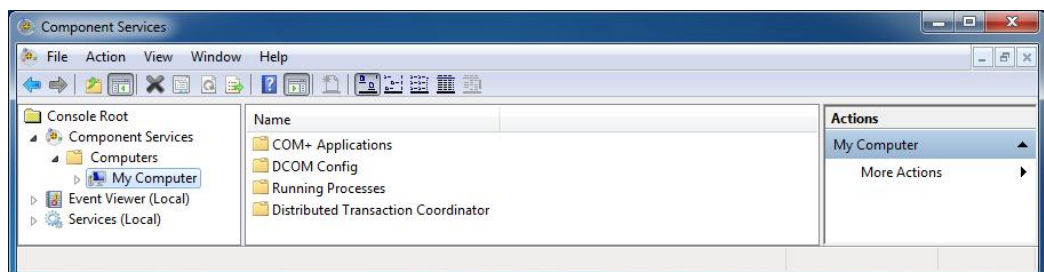
Click on the Windows icon or 'Start' button, then in the search box type 'dcomcnfg' and press enter.



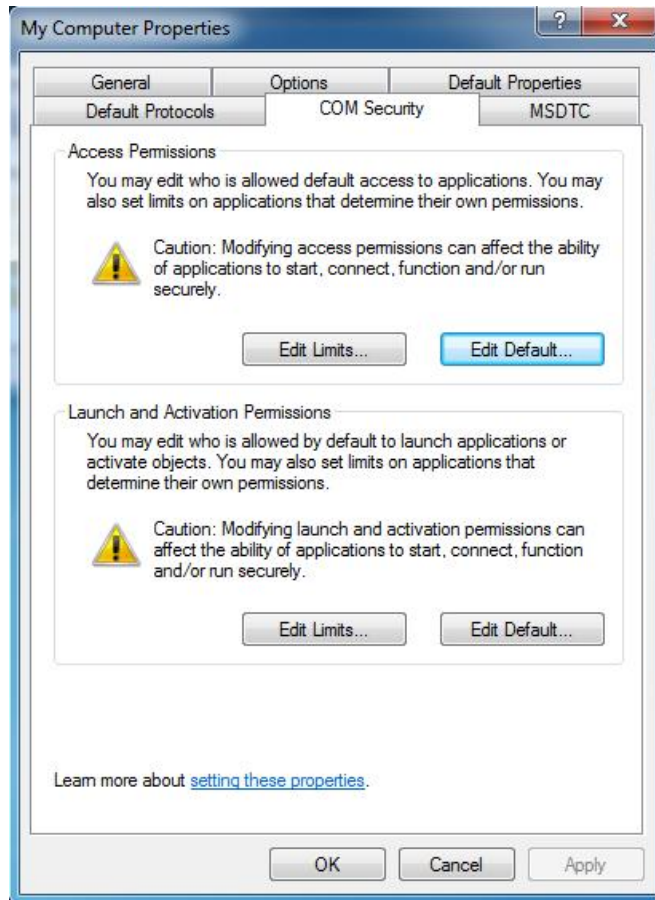
On Windows Vista, you may see the user account control prompt shown below. Click 'Continue' or 'Yes'.



In the first window that appears (see example below), click on the 'Component services' and 'Computers' items in the left hand pane to expand them, so that 'My Computer' is shown.

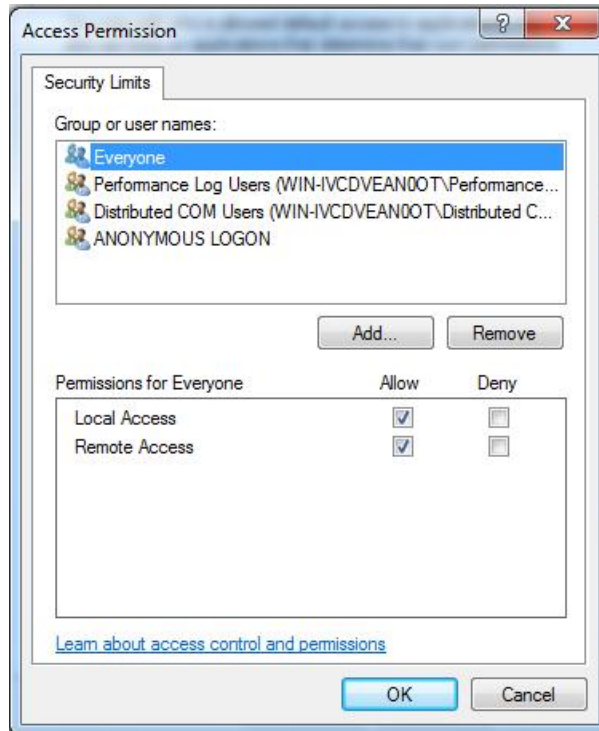


Right click on 'My computer' and select 'Properties' from the pop-up menu. In the next window, click on the 'COM security' tab to show the dialog below:



The 'Limits' on this page will override the COM security settings for individual components, and if the settings are too restrictive there can be problems accessing the OPC server.

In the 'Access permissions' section, click 'Edit limits' to display the window below.

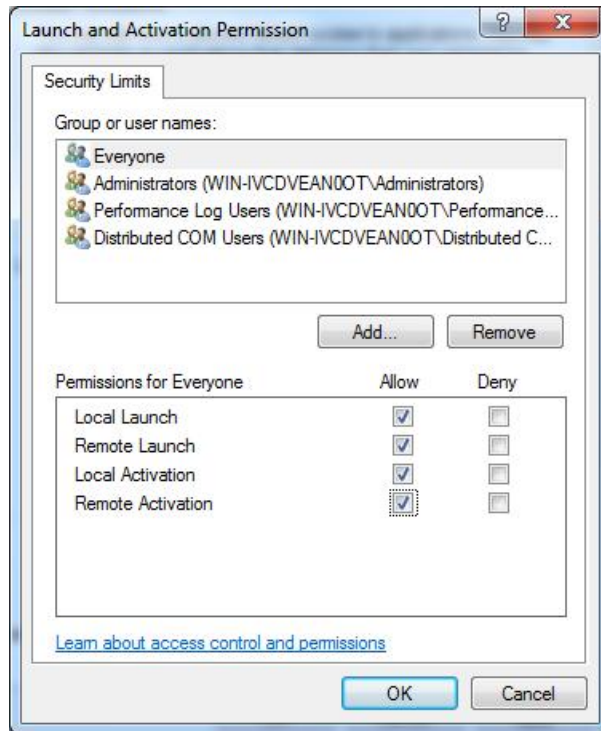


In the example above, DCOM can be used by all users (the 'Everyone' group) – if 'Everyone' has at least 'Local access' rights, you do not need to change these settings further. Otherwise, add the account(s) to be enabled (e.g. ASPNET) to this list with the 'add' button, and make sure that at least 'Local access' is enabled.

**Note:** Although using XML-DA with an OPC server on the same PC as IIS does not require 'Remote access' to be enabled, DCOM does require remote access. If you have already configured DCOM you may find that 'Remote access' is already enabled for some or all users – there is no need to remove it.



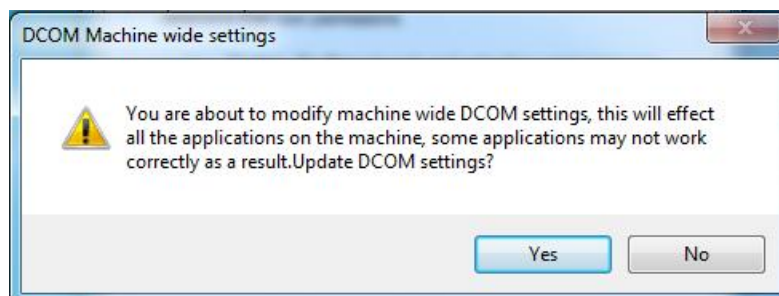
Click 'OK' to save the changes, then on the 'COM security' screen click the 'Edit limits' button in the 'Launch and Activation Permissions' section to show the window below:



Ensure that all the 'Allow' checkboxes are set for the 'Local launch' and 'Local activation' permissions of the user account that will be accessing the server (e.g. ASPNET). If the 'Everyone' group already has permissions (which is the case in the above picture), it should not be necessary to add the new user account separately. As before, there is no need to remove the 'Remote' permissions if they have already been set (for example as part of DCOM configuration).

Click 'OK' to return to the 'COM security' window. The 'Edit defaults' settings for 'Access permissions' and 'Launch permissions' on this page are used to set the default permissions for COM applications. If you are not too concerned with security (for example, if your computer(s) are running on an isolated network with no internet connection), one alternative at this point is to give the new user (e.g. ASPNET) default access and launch/activate permissions to *all* COM objects using these dialogs. However, it is preferable to set permissions for just the objects that the user needs, which we will do in the rest of this section.

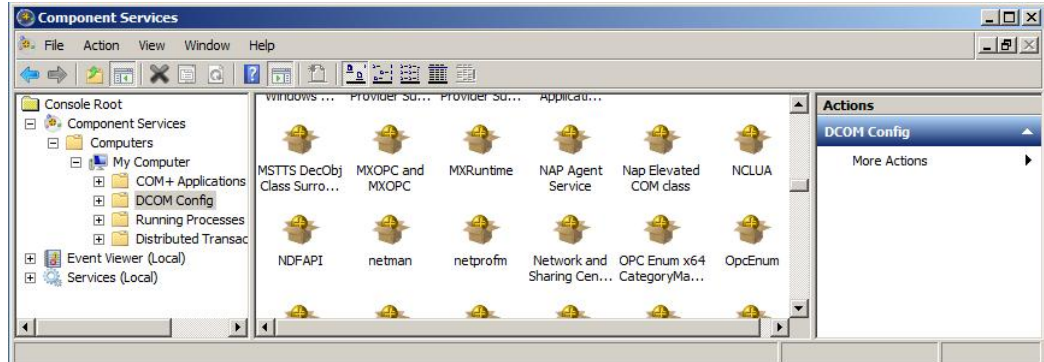
When you have finished, click 'OK' to return to the 'Component Services' dialog. You may see a warning message like the one below, in which case answer 'Yes'.



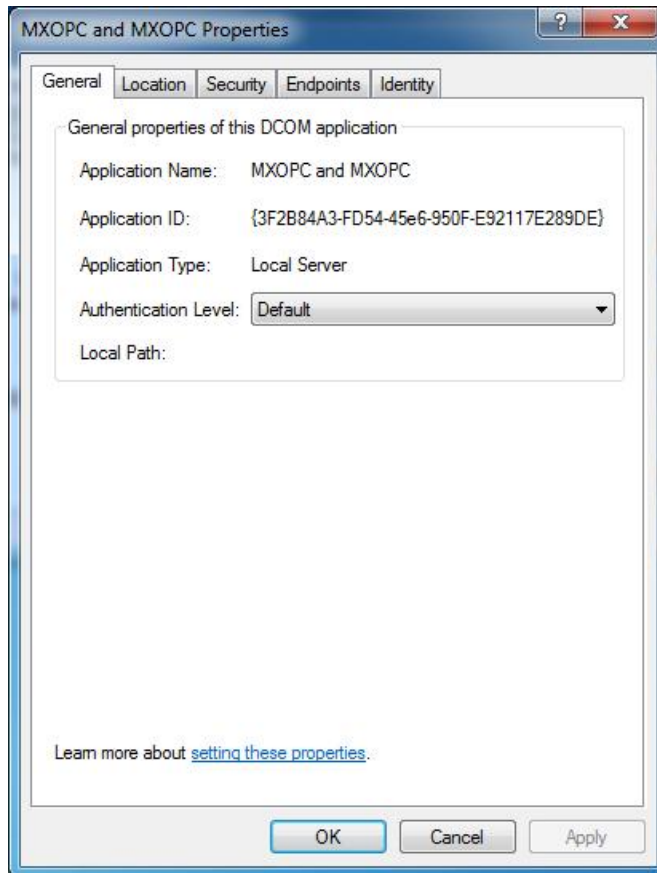


You will need to configure the security on the MX OPC server and MX Runtime services. To do this, double-click on the 'DCOM config' entry, and find 'MXOPC' and 'MXRuntime' in the list, as shown in the picture below. Depending on your operating system and installation, these items may not be visible in the window at the same time, and there may also be an 'OPC Enum x64 CategoryManager' item.

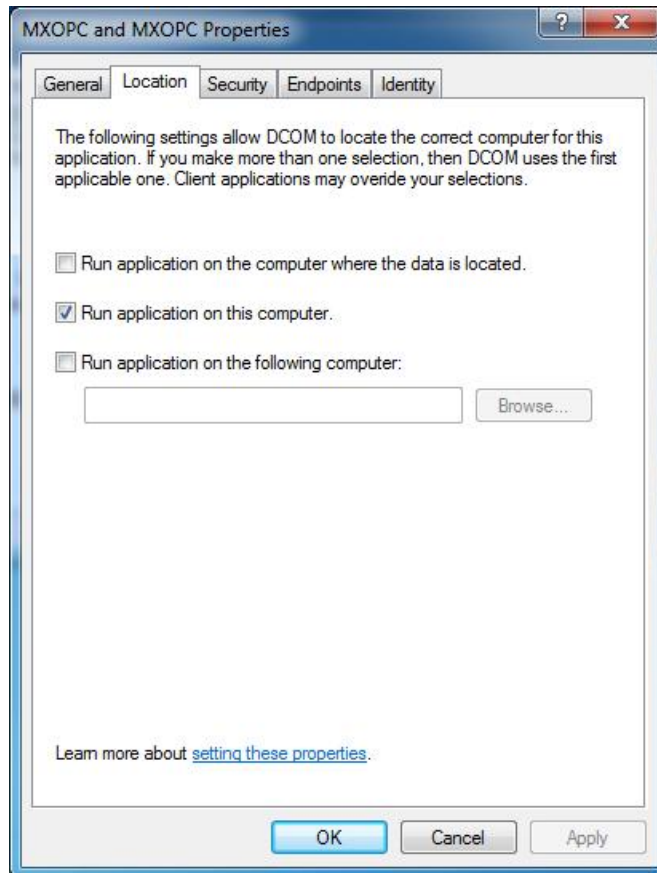
**Note:** As XML-DA does not support enumerating OPC servers, you do not need to set permissions for 'OPCenum'.



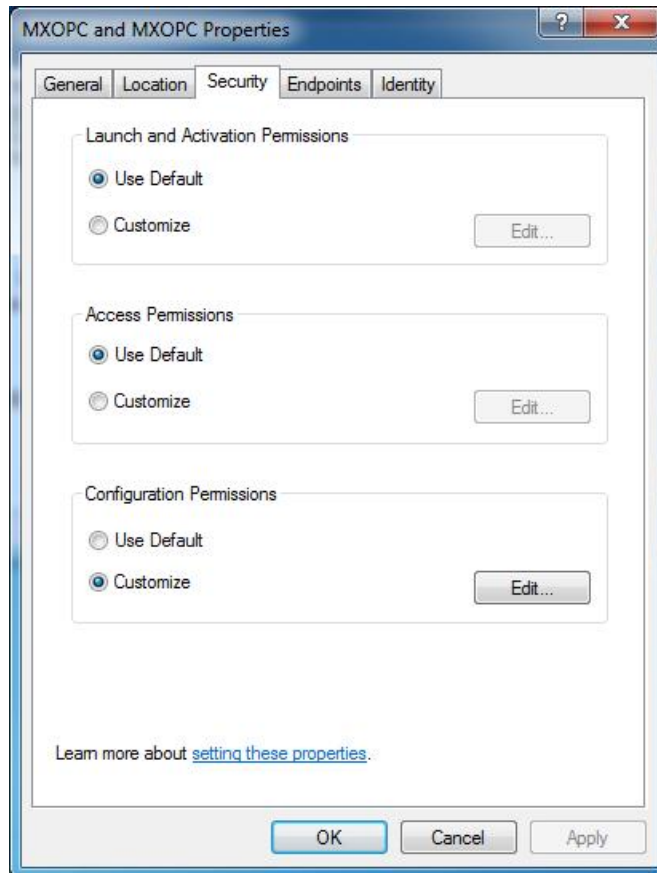
This step will need to be repeated for each item in turn – 'MXOPC' will be used for this example. Right click on the 'MXOPC' icon and select 'properties'. The window below should appear; confirm that the authentication level setting is 'Default'.



Switch to the 'Location' tab as shown below. Confirm that the 'Run application on this computer' check box is selected.

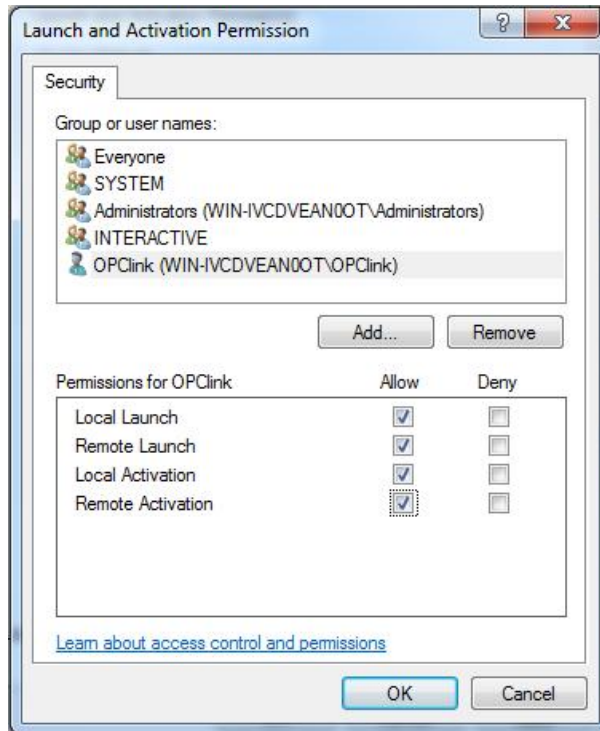


Now switch to the 'Security tab'.

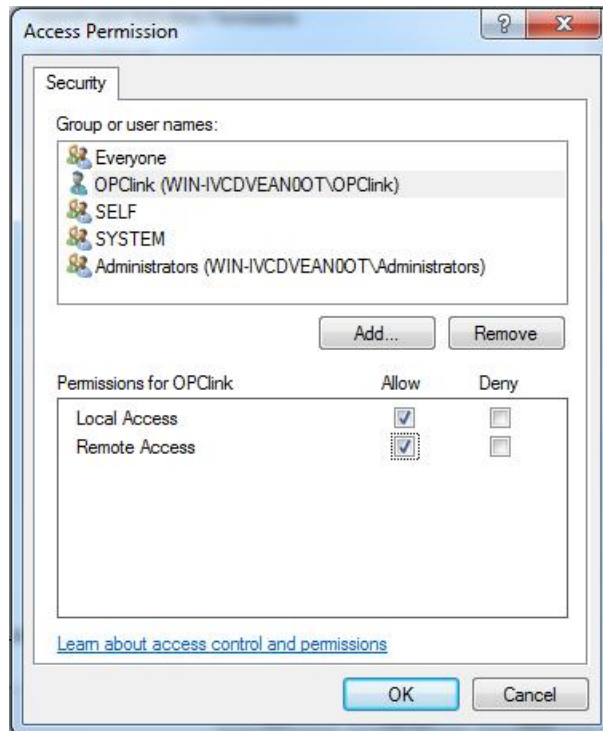


The 'Launch and activation permissions' and 'Access permissions' can both be left as 'Default' to use the default security settings configured earlier. Alternatively, they can be customized to allow more control over the users that can start and use each server. To do this, select 'customize' for each option and select 'Edit'.

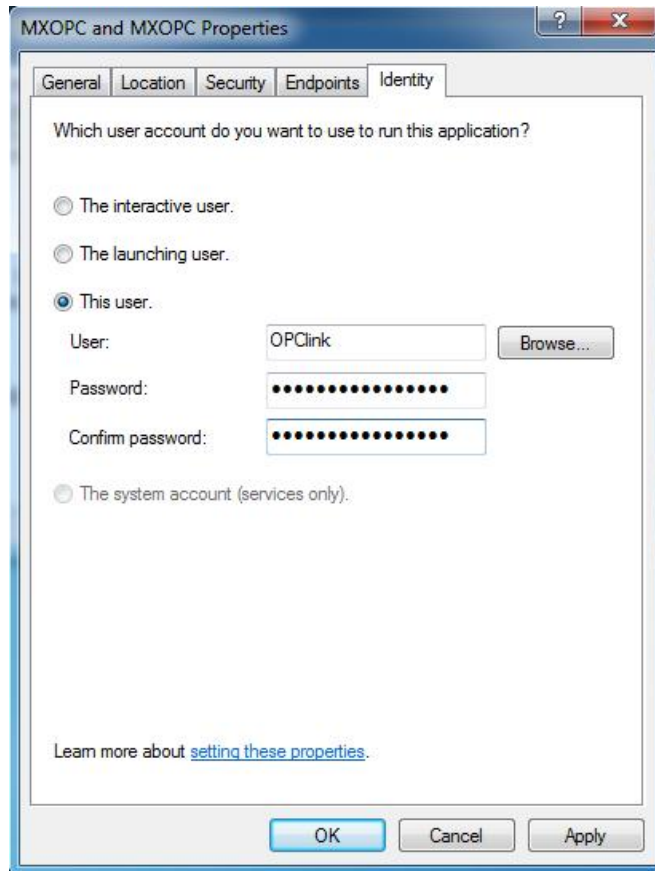
A sample 'Launch and activation permissions' edit window is shown below, in which a named user 'OPClink' has been added. As with the 'edit defaults' settings, all four 'allow' check boxes must be set for users requiring access through DCOM.



The 'Access permissions' configuration screen is shown below, with a named user 'OPClink' added.



After changing the settings on the security tab (if required), select the 'Identity' tab. In the sample screen below, the server has been configured to run as a named user.



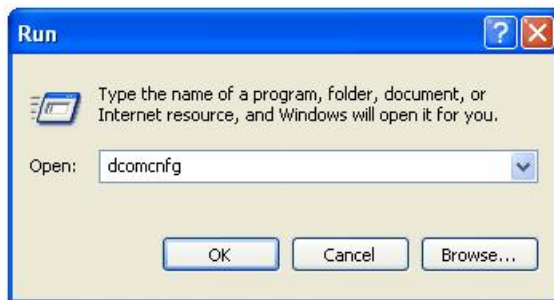
The options on this screen are:

- **The interactive user** – the user who is currently logged on to the machine. This can cause problems with DCOM, as there is no guarantee of which user (if any) will be logged on when the server is accessed from another computer.
- **The launching user** – the user who accessed the server, causing it to be started.
- **A named user** – the server runs as a specified user, whose name and password are provided. This option allows the server to be configured independently of the user that started the server and the user that is logged on, and works well for unattended computers.
- **The system account**, when the server is configured to run as a service.

Press 'OK' to return to the 'Component settings' page, which can then be closed.

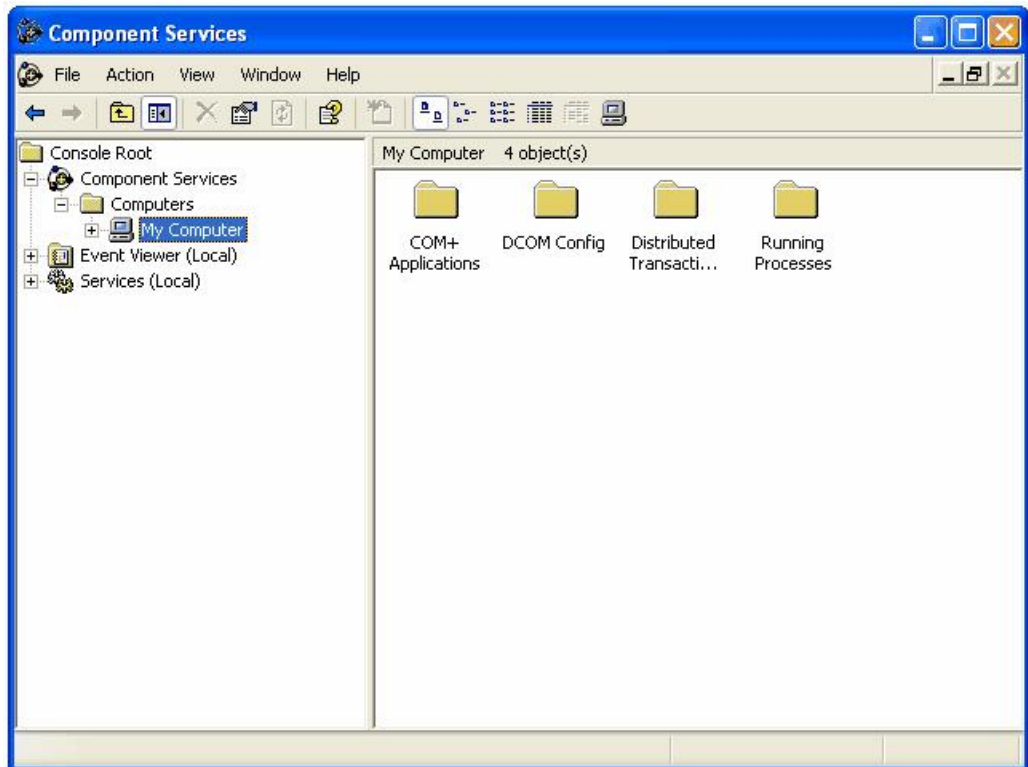
### 15.3.8.2 COM configuration on Windows XP service pack 2

From the 'Start' menu, select 'Run' to show the window below.

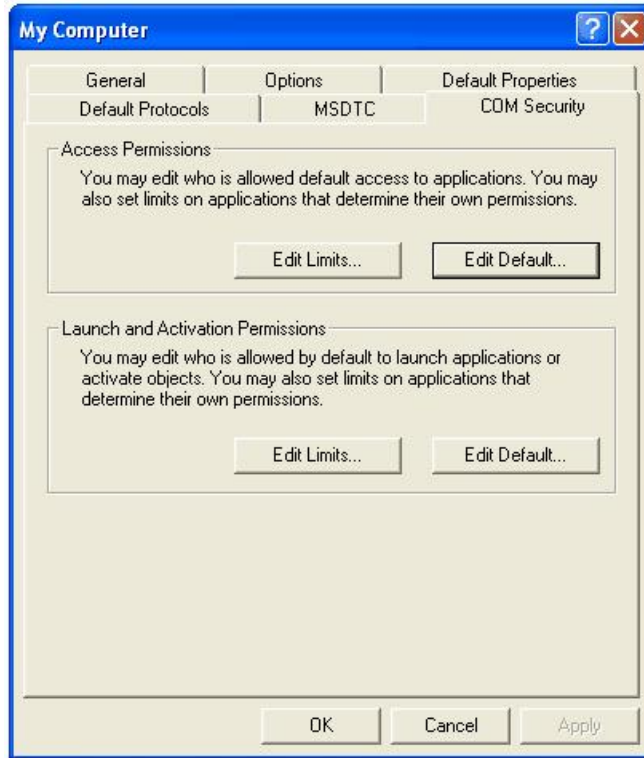


Enter 'dcomcnfg' and click 'OK' to start the DCOM configuration utility.

In the first window that appears (see example below), click on the 'Component services' and 'Computers' items in the left hand pane to expand them, so that 'My Computer' is shown.



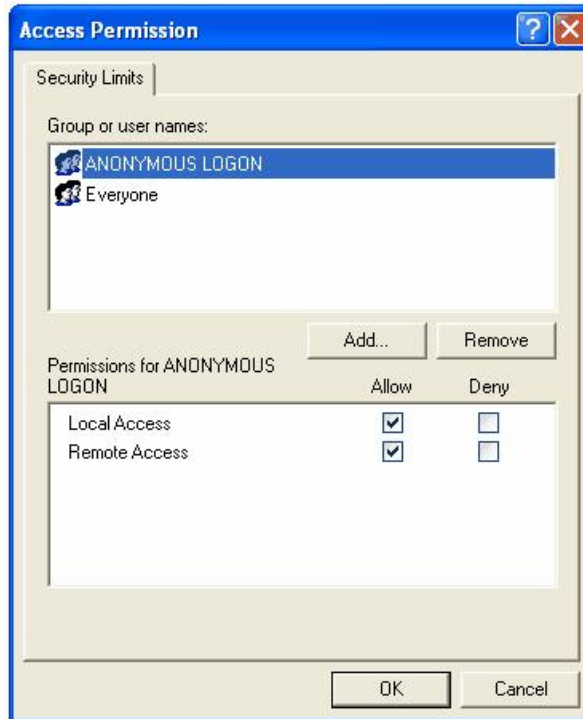
Right click on 'My computer' and select 'Properties' from the pop-up menu. In the next window, click on the 'COM security' tab and a window similar to the one below will be shown:



The 'Limits' on this page will override the COM security settings for individual components, and from XP service pack 2 onwards the default settings are too restrictive to allow OPC communications to work. The restrictions must be relaxed so it is possible to access the OPC server.



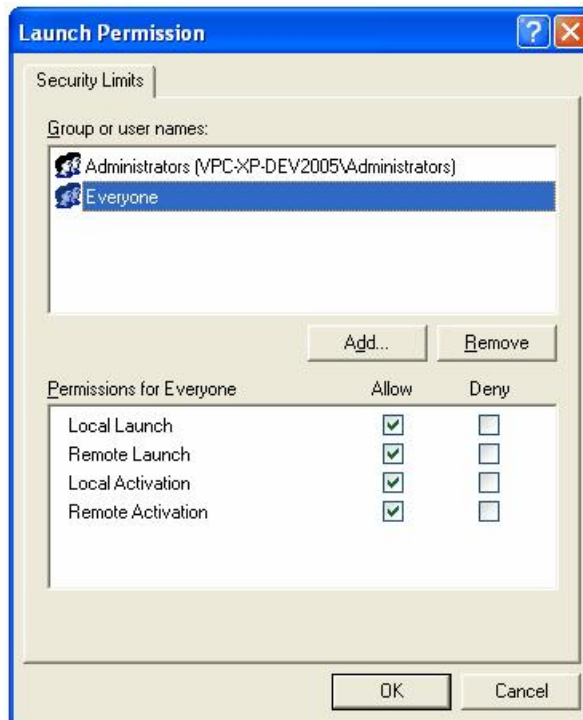
In the 'Access permissions' section, click 'Edit limits' to display the window below.



In the example above, DCOM can be used by all users (the 'Everyone' group) – if 'Everyone' has at least 'Local access' rights, you do not need to change these settings further. Otherwise, add the account(s) to be enabled (e.g. ASPNET) to this list with the 'add' button, and make sure that at least 'Local access' is enabled.

**Note:** Although using XML-DA with an OPC server on the same PC as IIS does not require 'Remote access' to be enabled, DCOM does require remote access. If you have already configured DCOM you may find that 'Remote access' is already enabled for some or all users – there is no need to remove it.

Click 'OK' to save the changes, then on the 'COM security' screen click the 'Edit limits' button in the 'Launch and Activation Permissions' section to show the window below:

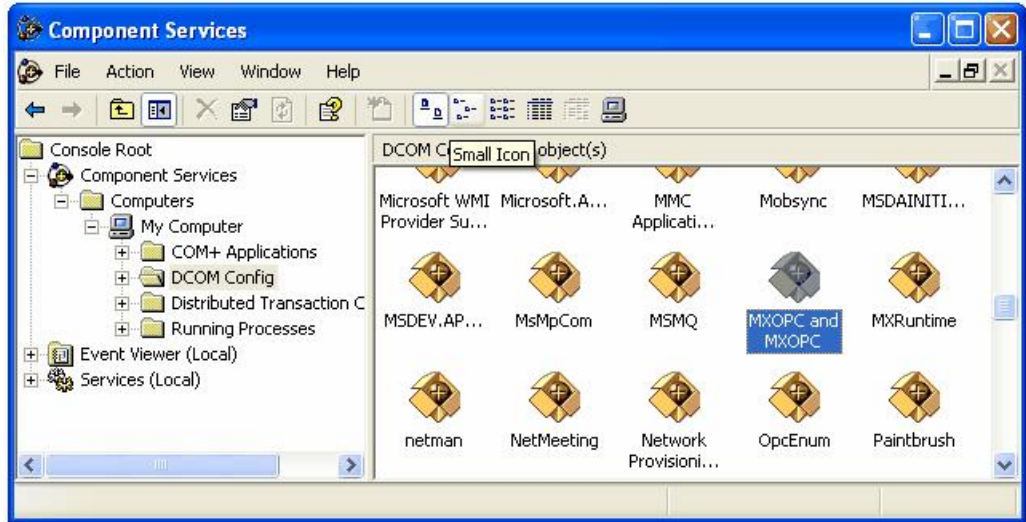


Ensure that all the 'Allow' checkboxes are set for the 'Local launch' and 'Local activation' permissions of the user account that will be accessing the server (e.g. ASPNET). If the 'Everyone' group already has permissions (which is the case in the above picture), it should not be necessary to add the new user account separately. As before, there is no need to remove the 'Remote' permissions if they have already been set (for example as part of DCOM configuration).

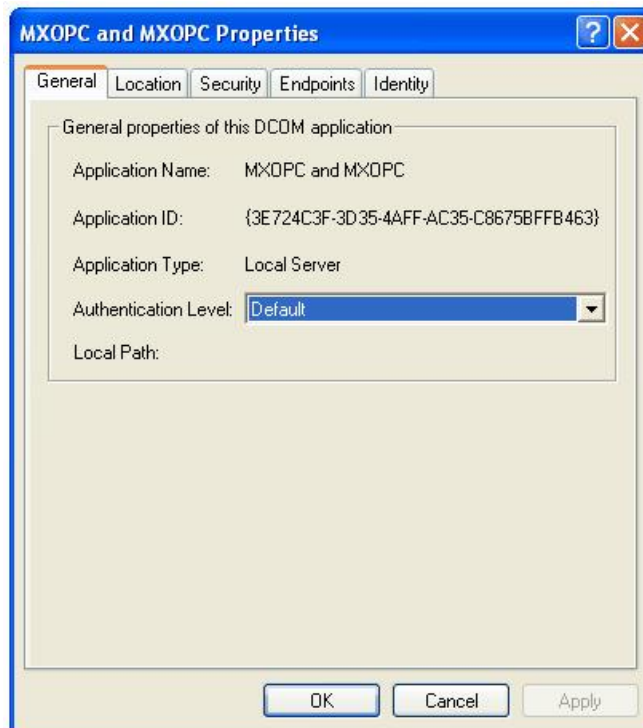
Click 'OK' to return to the 'COM security' window. The 'Edit defaults' settings for 'Access permissions' and 'Launch permissions' on this page are used to set the default permissions for COM applications. If you are not too concerned with security (for example, if your computer(s) are running on an isolated network with no internet connection), one alternative at this point is to give the new user (e.g. ASPNET) default access and launch/activate permissions to *all* COM objects using these dialogs. However, it is preferable to set permissions for just the objects that the user needs, which we will do in the rest of this section.

Click 'OK' to return to the 'Component Services' dialog. You will need to configure the security on the MX OPC server and MX Runtime services. To do this, double-click on the 'DCOM config' entry, and find 'MXOPC' and 'MXRuntime' in the list, as shown in the picture below.

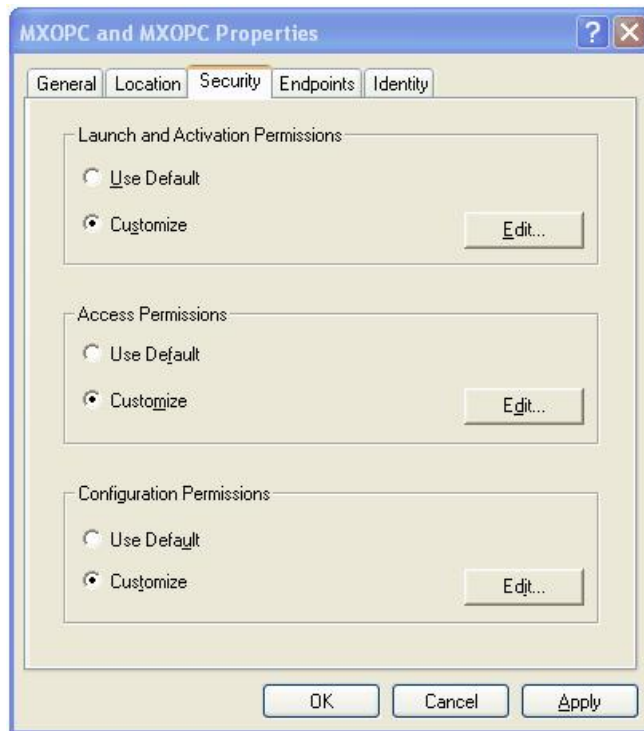
**Note:** As XML-DA does not support enumerating OPC servers, you do not need to set permissions for 'OPCenum'.



This step will need to be repeated for 'MXOPC' and 'MXRuntime' in turn – 'MXOPC' will be used for this example. Right click on the 'MXOPC' icon and select 'properties'. The window below should appear.

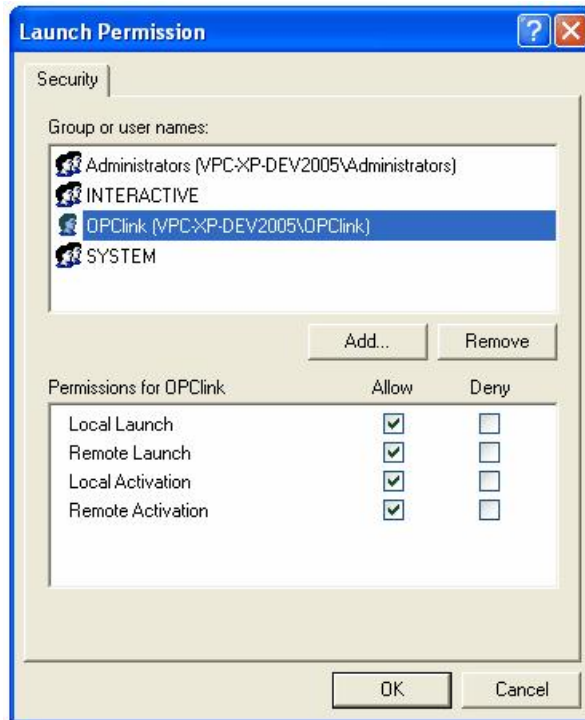


Switch to the 'Security tab'.

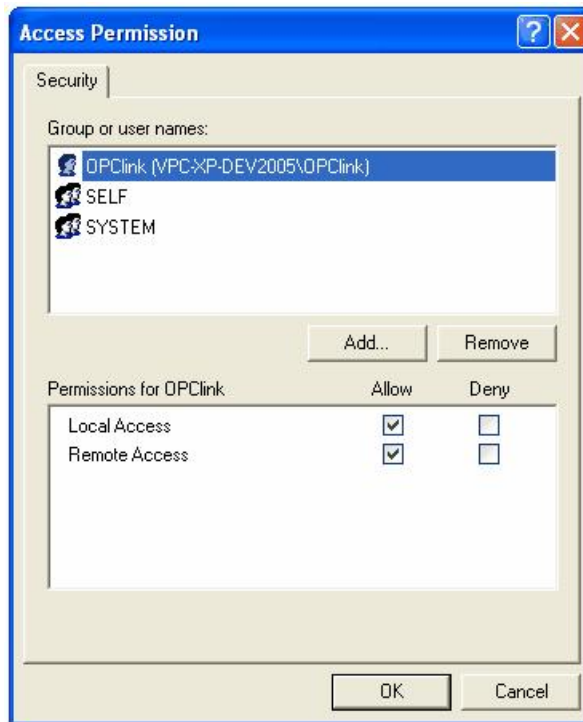


The 'Launch and activation permissions' and 'Access permissions' can be left as 'Default' to use the default COM security settings configured earlier. However, for better security, they can be customized to control which users can start and use each server. To do this, select 'customize' for each option and select 'Edit'.

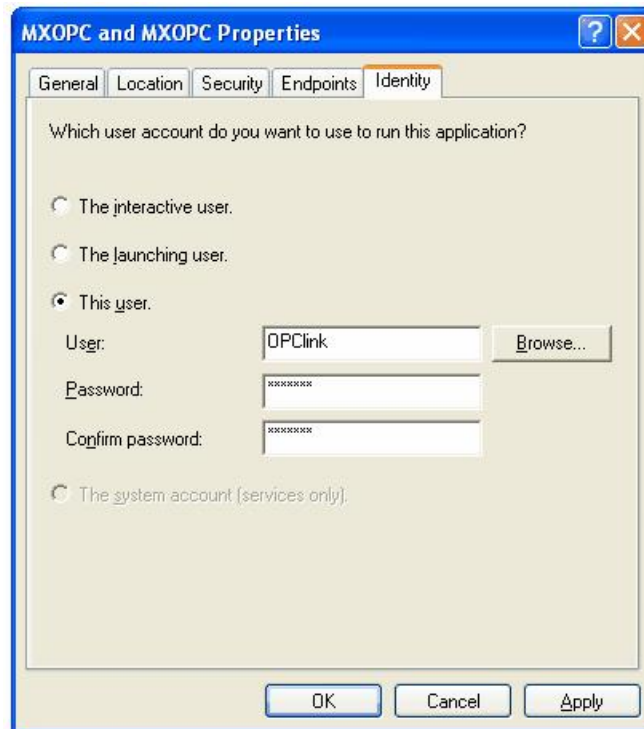
A sample 'Launch and activation permissions' edit window is shown below, in which a named user 'OPClink' has been added. As with the 'edit defaults' settings, the 'allow' check boxes for 'Local launch' and 'Local activation' must be set to allow use of the server with XML-DA. DCOM (if previously configured) also uses the 'Remote' permissions.



The 'Access permissions' configuration screen is shown below, with a named user 'OPClink' added. Only 'Local access' is needed for XML-DA usage.



After changing the settings on the security tab (if required), select the 'Identity' tab. In the sample screen below, the server has been configured to run as a named user.



The options on this screen are:

- **The interactive user** – the user who is currently logged on to the machine. This can cause problems if a remote user is using XML-DA with no-one logged on to the PC, as there is no guarantee of which user (if any) will be logged on when the server is accessed from another computer.
- **The launching user** – the user who accessed the server, causing it to be started.
- **A named user** – the server runs as a specified user, whose name and password are provided. This option allows the server to be configured independently of the user that started the server and the user that is logged on, and works well for unattended computers.
- **The system account**, when the server is configured to run as a service.

Press 'OK' to return to the 'Component settings' page, which can then be closed.

### 15.3.9 Configuring the Windows firewall

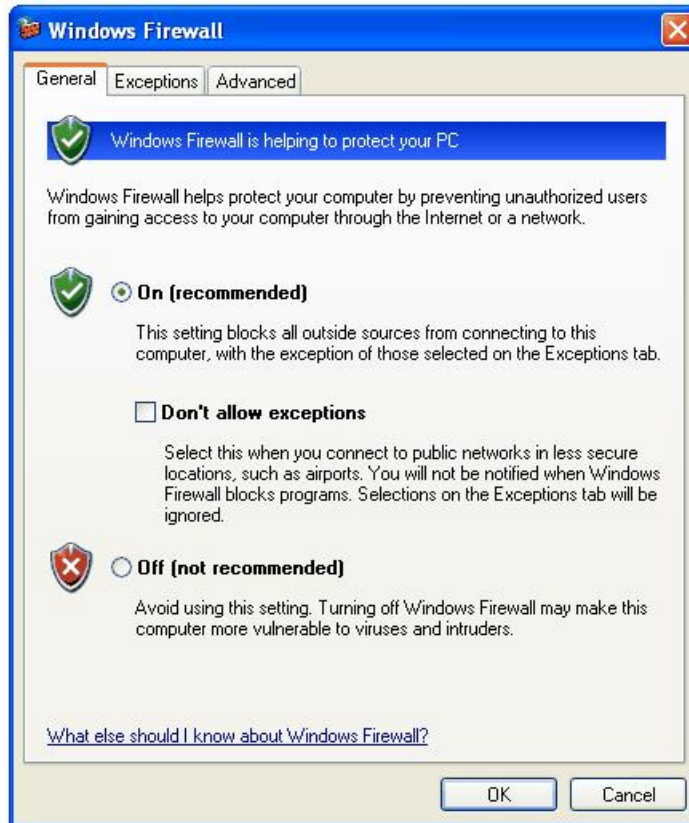
If you are using the Windows firewall provided with Windows XP service pack 2 (or later), you may need to configure it to allow TCP/IP network traffic to reach the XML-DA server in Internet Information Server (IIS). If you are using a third party software firewall, please refer to the manual to find out how to make the equivalent configuration changes. If XML-DA does not work when your third-party firewall is enabled, but works when it is switched off, the problem is likely to be with the firewall configuration.

To configure the firewall for XML-DA, following the instructions for either [Windows XP](#)<sup>[524]</sup> or [later operating systems](#)<sup>[526]</sup>.

**Note:** Windows 2000 does not include a firewall by default, but if you have installed a third-party software firewall, you will need to enable the same applications and ports as required by the XP firewall. Please refer to your firewall manual for instructions on how to do this.

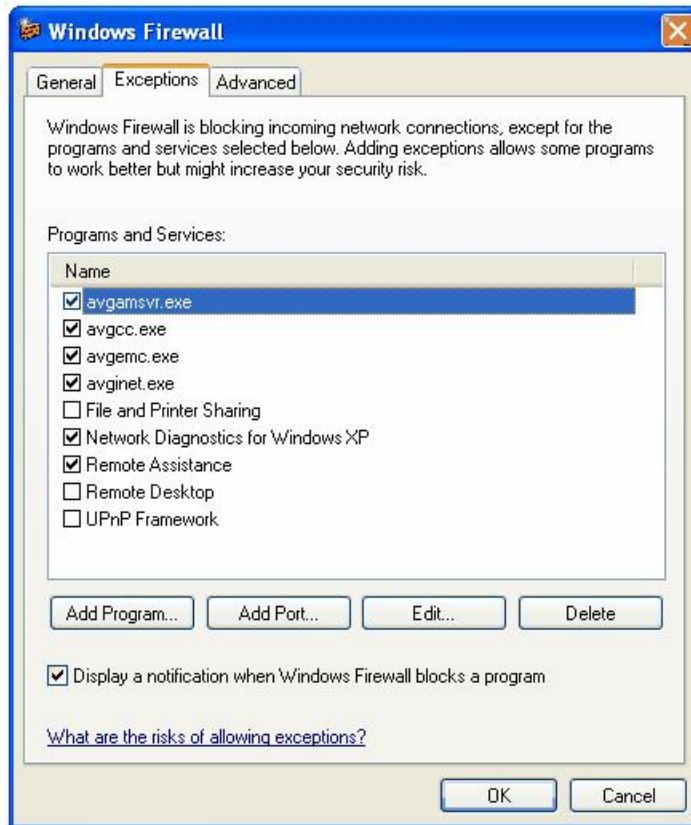
### 15.3.9.1 Configuring the Windows firewall (Windows XP)

From the 'Start' menu, open 'Control panel'. If you are using 'Classic view', the 'Windows firewall' icon will already appear in the list, otherwise in 'Category view' it will be under the 'Network and internet connections' category. Double-click the icon to view the firewall settings screen (shown below).

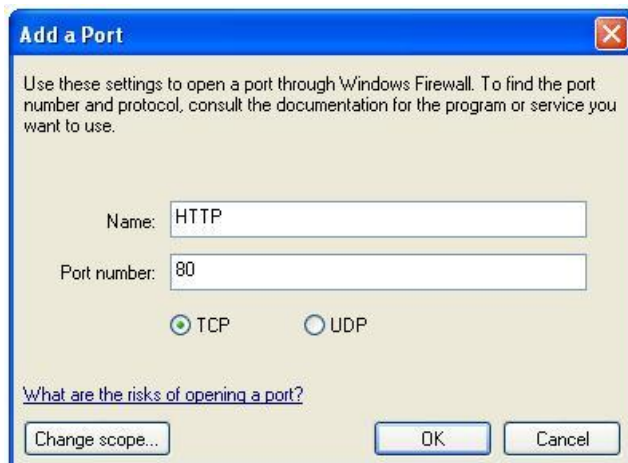




Click on the 'Exceptions' tab, and a screen similar to the one below will be shown.

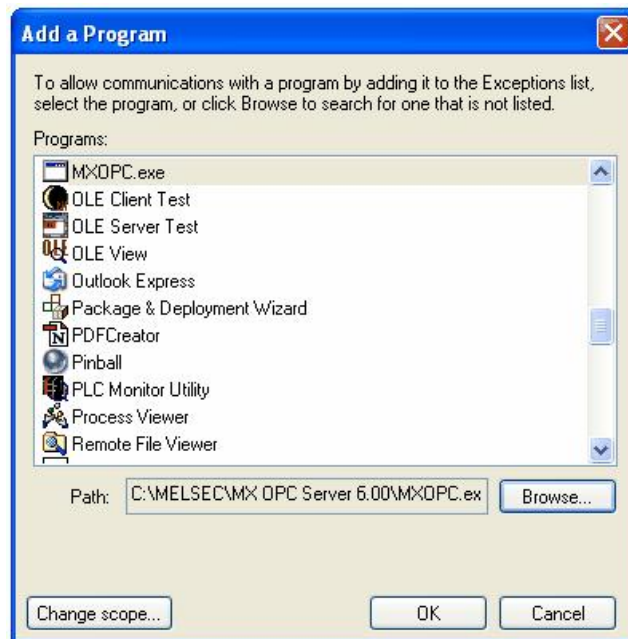


Click the 'Add port...' button, and enter the details below to allow other applications to connect to the HTTP port. If you have configured your web server on a port other than the usual '80', use the alternative port number instead and give it a different name.



Click 'OK' to return to the previous screen, where 'HTTP' will have been added to the exceptions list.

If you are connecting to XML-DA remotely from a Windows XP service pack 2 PC, you may also need to add your client program(s) to the exceptions list on the remote PC. Start by clicking on the 'Add program...' button on the exceptions screen. In the window that appears, use the 'Browse' button and find the EXE file for your client application. The screen should now look like the one below (where 'MXOPC.exe' has just been added):



Click 'OK' to return to the exceptions screen, where the client program should now appear in the list.

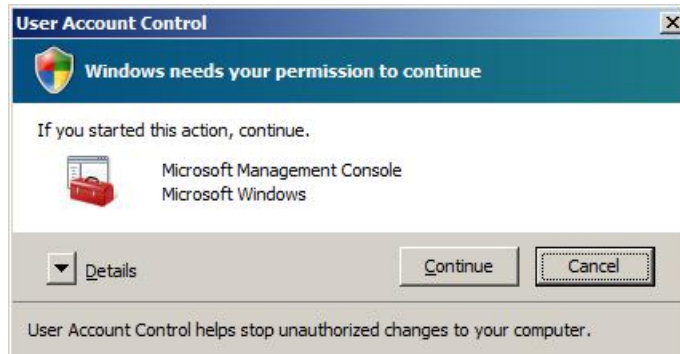
When all client applications have been added, click 'OK' on the exceptions screen to finish.

### 15.3.9.2 Configuring the Windows firewall (Windows Vista or later)

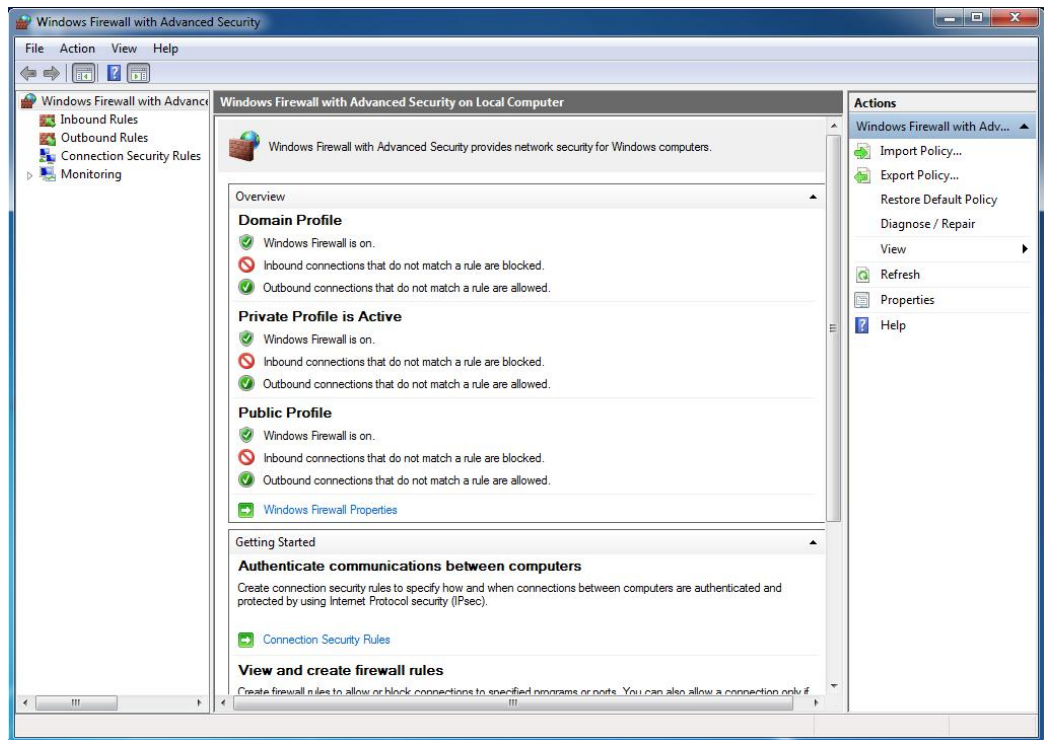
To edit the windows firewall settings, click on the Windows icon (or start button), and enter 'wf.msc' in the search box before pressing enter.



On Windows Vista, you may see a UAC prompt like the one below, in which case click 'Yes' or 'Allow'.

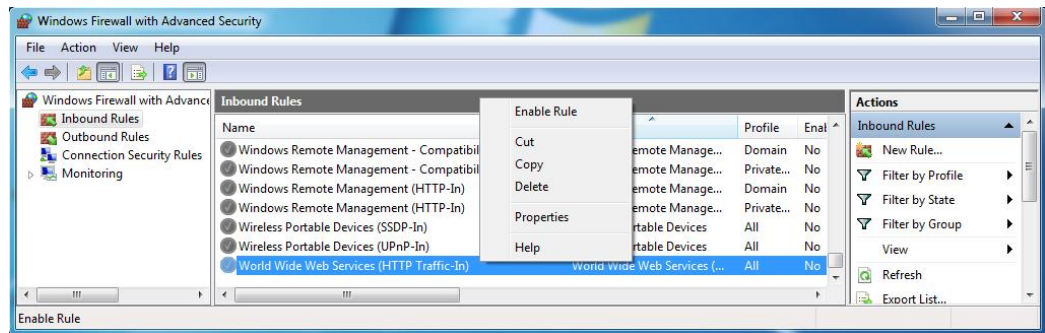


This will show the firewall advanced settings page, as in the sample below.



Advanced firewall settings

Click on 'Inbound Rules', and find 'World Wide Web Services (HTTP Traffic-In)' in the list. Right click on it and select 'Enable Rule'. If 'Disable Rule' is shown in the context menu instead, the rule is already enabled and does not need to be changed.



Enable HTTP traffic

## 15.4 Testing XML-DA

After all the software has been installed and configured, you can test the server by going to any browser and entering the URL of the XML-DA web service. This will be in the form:

```
http://<computername>/<virtualdirectory>/<servicename>.asmx
```

where

- <computername> is the PC name or address. 'localhost' can be used as a shorthand for the local PC
- <virtualdirectory> is the IIS directory linked to the web service – 'xmllda' in the examples
- <servicename> is the web service file – 'Mitsubishi.MXOPC.6.asmx' in the examples

So for the example configuration in these instructions, the URL would be:

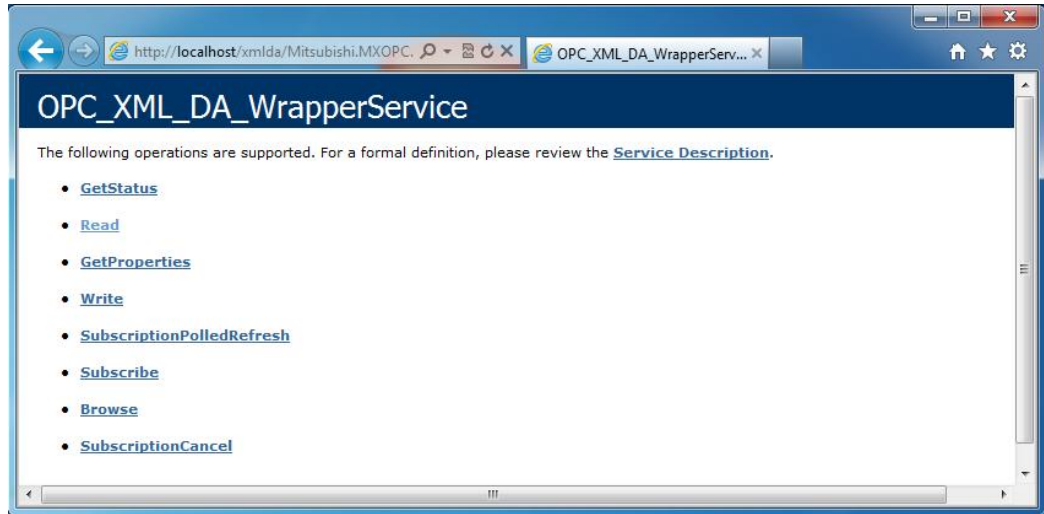
```
http://localhost/xmllda/Mitsubishi.MXOPC.6.asmx
```

Depending on your firewall settings, the service can be accessed from other PCs by substituting the server computer name or address in place of 'localhost'.

Note: If your IIS installation does not use the default web port (80), you will need to alter the URL to include the port number, separated from the computer name by a colon (:) character. For example, if IIS is configured for port 8080:

```
http://localhost:8080/xmllda/Mitsubishi.MXOPC.6.asmx
```

If the web server configuration is correct, you should see a page similar to this:



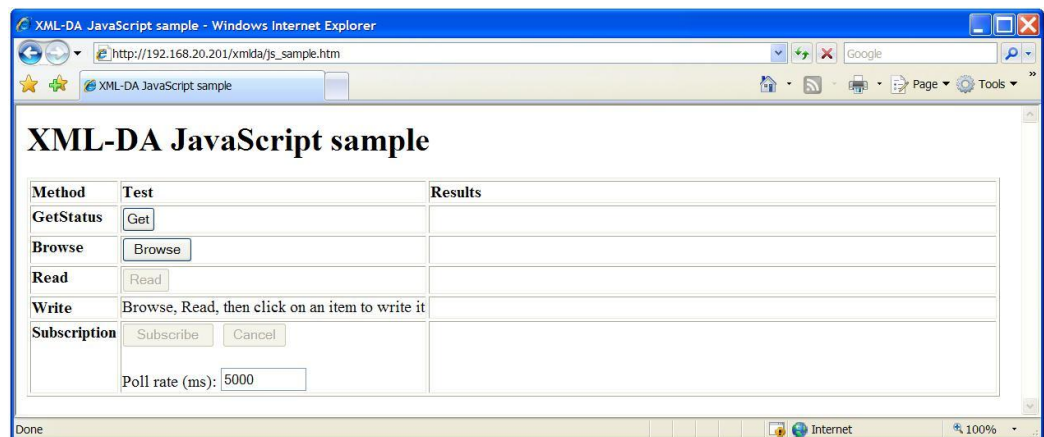
This demonstrates that the server is configured and listening, but does not necessarily prove that it can be used to read data, as the connection from IIS to the OPC server is not checked at this stage.

To prove that the OPC connection also works, you will need to connect to the server using an XML-DA client. The Javascript sample code can be used for this purpose. Find the file 'js\_sample.htm' from the 'Javascript' directory under 'XML-DA' on the CD, and copy this into the directory where you installed the XML-DA wrapper ('C:\MELSOFT\XMLDA' in the examples). If your wrapper service file was not called 'Mitsubishi.MXOPC.6.asm' (as in the example), use notepad to find this text in the 'js\_sample.htm' file, and modify it to match the name that you used.

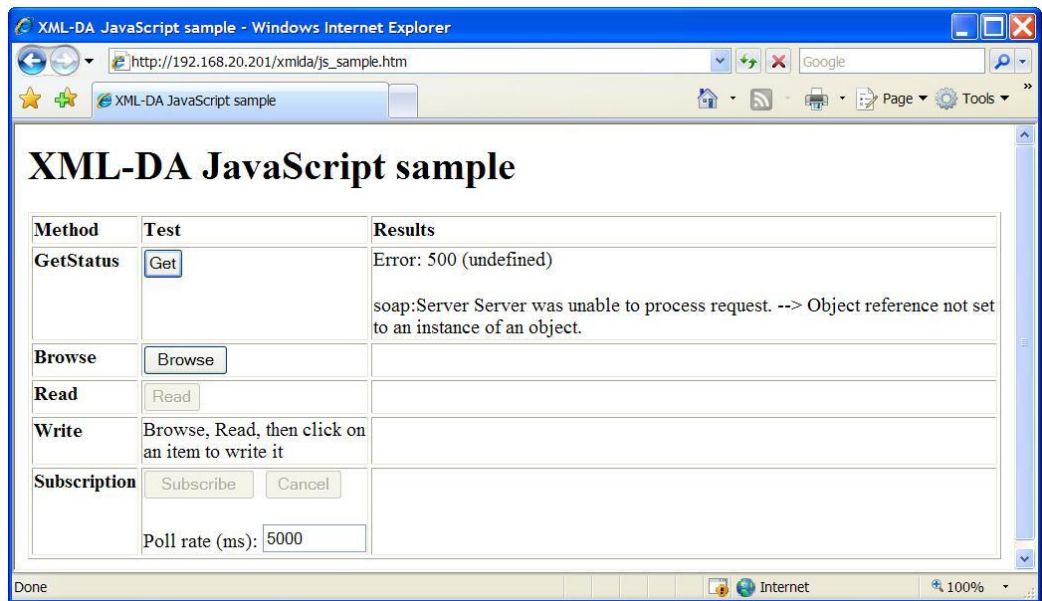
Now replace the 'Mitsubishi.MXOPC.6.asm' part of the test URL in Internet Explorer with 'js\_sample.htm', e.g.:

```
http://localhost/xmlda/js_sample.htm
```

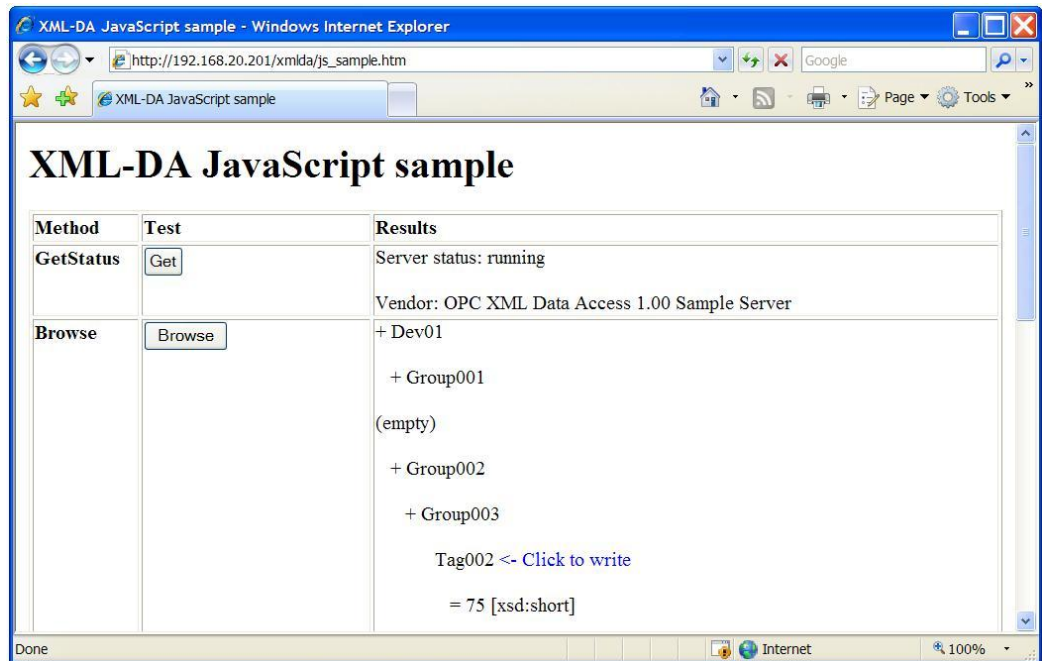
The javascript example page should appear, similar to the picture below.



If you are able to use the 'Get' [status], 'Browse' and 'Read' buttons on the example page, the XML-DA web service is working correctly. If there are errors reported, the service may need further configuration. The example picture below shows an error caused by incorrect DCOM permissions.



After fixing the permissions, the example can be used as normal (see picture below where Tag002 is about to be written).



## 15.5 Programming with XML-DA

XML-DA is well-suited to modern development tools which provide support for web services. From Visual Studio 2003 onwards, the development environment will automatically generate most of the code needed to access a web service.

XML-DA offers a cut-down version of the features available in OPC. The operations supported are:

Operation	Description and notes
GetStatus()	Determine the status of the OPC server.
Browse()	Read the tag hierarchy. It will only read one level at a time, so it must be called repeatedly to read the full hierarchy.
GetProperties()	Get the properties of an item.
Read()	Read the values of one or more tags.
Write()	Write new values to one or more tags.
Subscribe()	Applications can 'subscribe' to receive information about one or more groups of tags, then regularly poll the server to see which of the tags have changed. This is a more efficient way to poll for data with XML-DA than making repeated calls to Read().
SubscriptionPolledRefresh()	
SubscriptionCancel()	

### 15.5.1 Using XML-DA from JavaScript

As XML-DA will respond to standard 'http' requests, it can be used directly from client-side JavaScript through the 'XMLHttpRequest' object. This allows operations to be performed on the OPC database and the results displayed without having to reload the whole web page, using techniques often called 'AJAX' (Aynchronous Javascript And XML).

An example web page 'js\_sample.htm' is provided under the 'XML-DA' directory on the OPC server CD. The features of this example are:

- Read server status using 'GetServerStatus()'
- Browse tags hierarchically using multiple 'Browse()' calls.
- Read tag values using 'Read()' and display them.
- Write a new value to a selected tag using 'Write()'.
- Subscribe to data with 'Subscribe()', and display changed values in real time without reloading the page using 'SubscriptionPolledRefresh()'. Unsubscribe with 'SubscriptionCancel()'.

The example code has been tested with Internet Explorer only, but should be possible to convert to other browsers. It is mainly intended to show the use of asynchronous javascript to make calls to XML-DA and parse key parts of the result data. It does not cover the use of all the many different parameters available in the XML-DA specification.

### 15.5.1.1 Using the 'XMLHttpRequest' object to call web services

To call a method in a web service, there are two separate stages. Firstly, the request is created and sent, then once the result has been received a separate function will be called asynchronously to process the result.

To initiate a web service call, the general structure of the code is shown below. The format of the SOAP request message depends on the web service to be called – the test page for the web service will often show a list of methods and a sample of the request format for each one.

```

var reqObj; /* Declare object to use for request, at global level */

/* ... within a function */
function (...etc...)
{
    /* ... other code ... */
    reqObj = new XMLHttpRequest;
    reqObj.open("POST", strSvrName);
    reqObj.setRequestHeader("SOAPAction",
        "http://opcfoundation.org/webservices/XML-DA/1.0/methodname");
    reqObj.setRequestHeader("Content-Type", "text/xml; charset=utf-8");

    /* Must set a function to be called when results are ready */
    reqObj.onreadystatechange = resultfunctionname;

    // Prepare request message
    var strReq = '<?xml version="1.0" encoding="utf-8"?>'
        + '<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://
            schemas.xmlsoap.org/soap/envelope/">'
        + '<soap:Body>'
        + 'data here depends on the parameters to the method'
        + '</soap:Body>'
        + '</soap:Envelope>';

    reqObj.send(strReq); /* Transmit request */

    /* ... other code ... */
} /* ... end of function ... */

```



Once the request has been processed by the server, the data will be returned through a call to the function specified in the 'onreadystatechange' element of the request structure. The code to process these typically takes the form below, with a check to see if the call has completed (as the function will also be called for other changes in request state such as 'transmitting', 'waiting for response' etc.). If so, the result code can be checked to see if the request succeeded. Finally, the XML data in the '.responseXML' element of the request object can be processed to find the results. The '.responseText' element shows the text of the returned message, which can be useful for debugging.

```

/* Function called when soap result is returned */
function resultfunctionname()
{
    /* Check if status is 'result received' */
    if (reqObj.readyState == 4)
    {
        /* Check success */
        if (reqObj.status == 200)
        {
            /* Successful call, can process response message */
            /* ... code to process XML results goes here ... */
            /* ... 'reqObj.responseXML' contains the response message ... */
        }
        else
        {
            /* Error, show details in status bar */
            window.status = "GetStatus() failed error " + reqObj.status
                + " (" + reqObj.statusText + ")";
        }

        /* Display results */
        document.getElementById("StatusResult").innerHTML = strOut;
    }

} /* End if result available */

} /* End function GetServerStatusResult() */

```

The 'responseXML' element of the request object will contain the XML message returned from the server. The exact message format depends on the method called, and there will be an example of this in the test page of the web service itself. The relevant data can be extracted by traversing the hierarchy of XML data. Some examples are shown below for the 'GetStatus()' and 'Read()' methods.

```

/* Example of extracting data from the 'GetStatus()' method */
var resResponse = reqObj.responseXML.getElementsByTagName("GetStatusResponse");
var resStatus = resResponse[0].getElementsByTagName("GetStatusResult")[0]
    .getAttribute("ServerState");
var resVendor = resResponse[0].getElementsByTagName("VendorInfo")[0].text;

```

```

/* Example of extracting data from the 'Read()' method */
var resResp = reqObj.responseXML.getElementsByTagName("ReadResponse");
var resList = resResp[0].getElementsByTagName("RItemList");
/* Check for 'no items' (subscription polling may return this) */
if (resList.length > 0)
{
    var resElems = resList[0].getElementsByTagName("Items");
    for (i = 0; i < resElems.length; i++)
    {
        /* Recover client item handle */
        var strHandle = resElems[i].getAttribute("ClientItemHandle");
        var resVal = resElems[i].getElementsByTagName("Value");

        /* Get type/value (value may have several nodes, combine as list) */
        var datavalue = "";
        for (l = 0; l < resVal[0].childNodes.length; l++)
        {
            if (datavalue != "")
            {
                datavalue += ", ";
            }
            datavalue += resVal[0].childNodes.item(l).text;
        }

        var datatype = resVal[0].getAttribute("xsi:type");

        /* ... do something with datatype and datavalue here ... */

    } /* End for (items read) */
}

```

### 15.5.1.2 Updating parts of the web page dynamically

One simple way to update part of a web page without reloading is to use a '<div>' tag, which has an 'innerHTML' property that can be dynamically replaced with new HTML content to update the page. This is the method used in the example code.

In the definition of the web page, each '<div>' tag can be given an ID:

```
<div id="Result">&nbsp;  </div>
```

In script, the content can then be replaced with code such as:

```
document.getElementById("Result").innerHTML = "<p>Ready</p>";
```

### 15.5.1.3 Sample application structure

As the sample application is already available as a whole on the CD, the code will not be duplicated here. Instead, this section will summarize the structure of the sample application and how it works.

Feature	Function / Variable	Description
Shared code	<code>arrTagList</code>	Stores a list of tags returned by hierarchical browsing, for use by other operations.
	<code>MakeTag()</code>	Creates a tag object for addition to <code>arrTagList</code> .
GetStatus	<code>GetServerStatus()</code>	Reads OPC server status with <code>GetStatus()</code> .
	<code>GetServerStatusResult()</code>	Result function for the <code>GetStatus()</code> operation. Processes the result message and displays key data.
Browsing	<code>arrBrowseQueue</code>	Since XML-DA browsing only returns one level at a time, a single browse call cannot fetch the whole tag hierarchy. To get round this, a queue of browse requests is defined. When an item with child nodes is found by a <code>Browse()</code> call, it is pushed on the queue. After all items are processed, another <code>Browse()</code> call is made for the next item in the queue. When the queue is empty, there are no more nodes to 'expand' (browse).
	<code>MakeBrowseRequest()</code>	Create an object to add to the browse queue.
	<code>StartBrowse()</code>	Initiates a browse of the whole tag hierarchy by adding the top-level (empty <code>ItemPath</code> ) item to the queue and calling <code>StartBrowseSub()</code> .
	<code>StartBrowseSub()</code>	If there is anything in the browse queue, it removes the next item from the queue and initiates a <code>Browse()</code> call to process it. When the queue is empty, the browse operation is treated as complete.
	<code>BrowseResult()</code>	Called when a <code>Browse()</code> request returns data. Items with child nodes are added to the hierarchy 'tree' using placeholder '<div>' tags that can be filled in later, and pushed onto the browse queue. Details of readable items are added to the <code>arrTagList</code> array for use by other calls. When all items have been processed, ' <code>StartBrowseSub()</code> ' is called to process anything left in the queue. Items are given additional script code to call ' <code>StartWrite()</code> ' when they are clicked.

Feature	Function / Variable	Description
Reading	ProcessReadData()	Updates the hierarchy 'tree' with data received from a Read() or SubscriptionPolledRefresh() operation, by replacing the contents of the '<div>' placeholder for the value of the tag, located using its stored id.
	ReadResult()	Deals with the return message from a Read() call by passing it to ProcessReadData().
	StartRead()	Initiates a 'Read()' call using the tag list provided by the Browse() operation.
Writing	StartWrite()	Prompts the user for the value to write, then looks up the data type for the tag in arrTagList if there has been a previous 'Read()' or 'SubscriptionPolledRefresh()' call. Starts a write operation using the data and type (defaulting to 'short' if the type is not known).
	WriteResult()	Processes the return message from a 'Write()' call, checking for write errors.
Subscribing	strSvrSub	Stores the 'subscription handle' returned by the server, which is needed to read or cancel the subscribed data.
	StartSubscribe()	Calls the 'Subscribe()' method to subscribe to updates for the tag list returned by the 'Browse()' operation.
	SubscribeResult()	Called with the result from 'Subscribe()'. Checks for errors then stores the returned subscription handle. Sets up the browse timer if successful.
	SubscribeTimer()	Called on a timer after subscribing, to initiate SubscriptionPolledRefresh() calls regularly.
	SubscribePollResult()	Called when a SubscriptionPolledRefresh() call returns data, and processes it with the ProcessReadData() function to display the new values in the hierarchy 'tree'.
	CancelSubscribe()	Cancels the current subscription using the SubscribeCancel() call and stops the timer.
	SubscribeCancelResult()	Called with the result of a SubscribeCancel() operation. Displays a confirmation message.

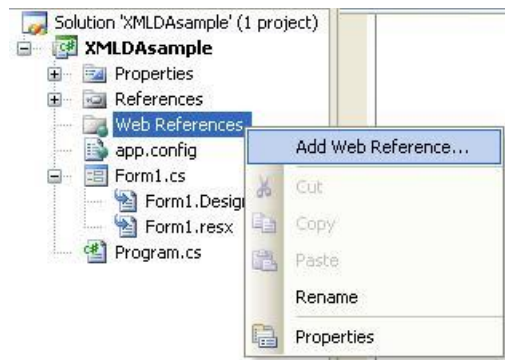
## 15.5.2 Using XML-DA from Visual Studio

Versions 2003 and above of Microsoft Visual Studio have built-in support for web services, which simplifies the development of XML-DA client applications.

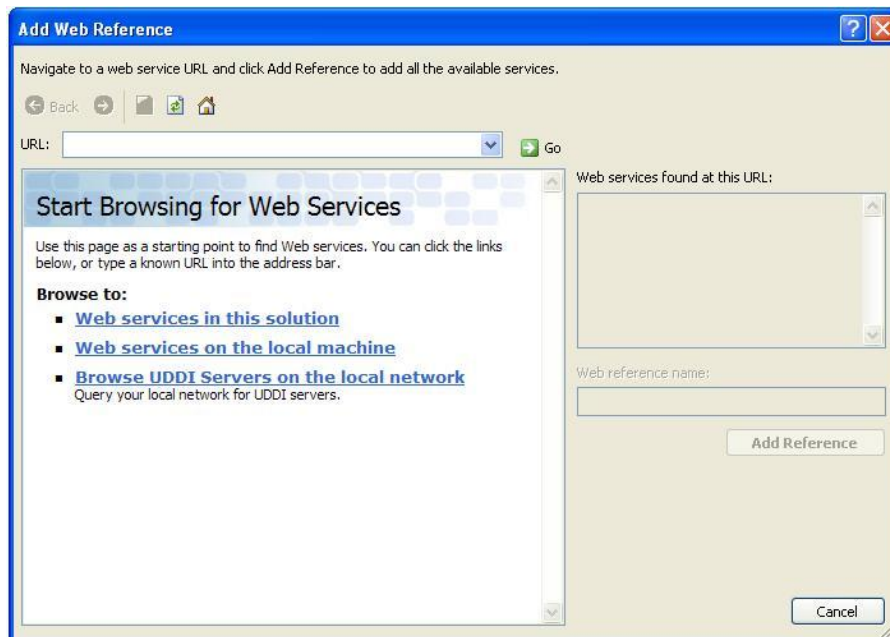
Visual Studio 2005 is used for the examples in this section, but Visual Studio 2003 uses web services in a very similar way. In both cases, the development environment removes a lot of the effort needed to use 'SOAP' methods, by automatically formatting and decoding XML messages based on the web service definition.

### 15.5.2.1 Adding a web service reference to a Visual Studio project

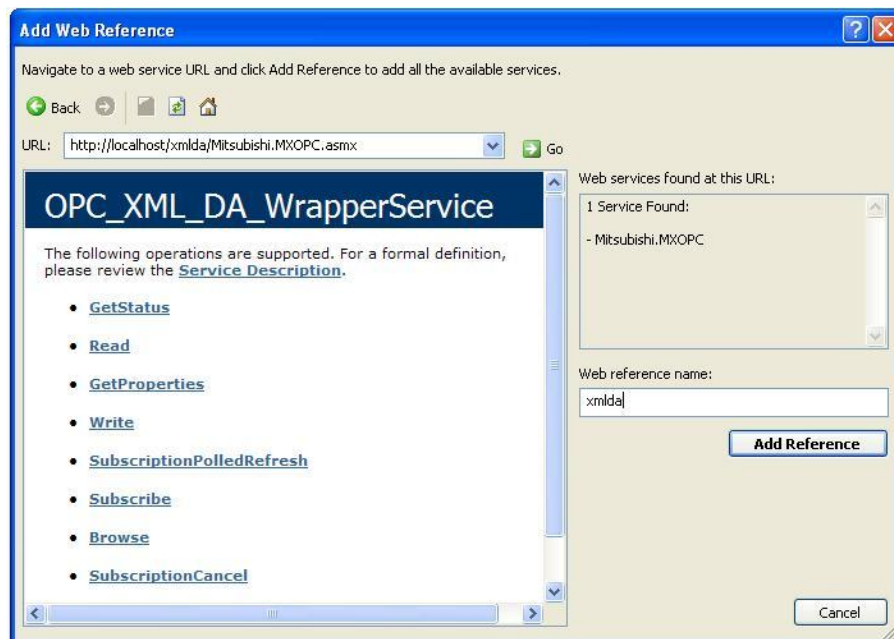
Regardless of the language used for the project, a reference to a web service can be added to the project from the 'solution explorer' tree. Right click on 'Web references' in the tree and select 'Add web reference'.



In the next dialog, you will be prompted to select the URL of the web service to add, which will be the same as the URL used in ["Testing XML-DA"](#)<sup>[528]</sup>. Alternatively, you can browse for a web service on the local computer, or on the network.



Enter the URL of the web service wrapper for OPC server, then select 'Go'. The web service test page should appear as it would in a normal browser, and the service should be identified correctly in the 'Web services found at this URL' box. Enter a name for the reference, such as 'XML-DA'.



Finally, select 'Add reference' to add the new web service reference to your project.

Using the new web reference, you can now declare an object that can be used to call methods on the XML-DA service. The example declaration below is in the 'C#' language:

```
// Web service binding
XML-DA.OPC_XML_DA_WrapperService srvXML = new XML-DA.OPC_XML_DA_WrapperService();
```

Methods such as GetStatus(), Browse() and Read() can be called from the new wrapper service object 'srvXML', e.g. 'srvXML.Read()' etc. The web service reference 'XML-DA' itself contains additional type declarations needed by the other methods.

### 15.5.2.2 C# sample application

The source code for the sample C# application can be found in the 'XML-DA\SampleC#' directory on the MXOPC server installation CD. Although this has been developed using Visual Studio 2005, there is little difference between this code and the equivalent in Visual Studio 2003.

The sample is a form-based application which demonstrates the use of the most common XML-DA methods. The code uses a web reference defined as described in the previous section to access the XML-DA server. Most of the calls are made synchronously, except for the 'SubscriptionPolledRefresh()' call where an asynchronous call is used, as there are situations where it is useful to receive regularly updates of data without delaying the front-end application.

This example only demonstrates the use of the most common parameters. More information on the full capabilities of XML-DA can be found in the XML-DA technical specification, available from the OPC foundation.

## 15.5.2.2.1 Shared code and definitions

The code and definitions below are shared between the different functions. The service itself is defined once as a single instance, and the list of available tags is available as an ArrayList. The other items will be described later with the operations in which they are used.

```
// Web service binding used by all items
XML-DA.OPC_XML_DA_WrapperService srvXML = new XML-DA.OPC_XML_DA_WrapperService();

// Structure to store information about readable items
struct tTag
{
    public string strItemName; // Name of item
    public string strItemPath; // Path to item
    public TreeNode refNode; // Tree node entry
    public string strName; // Item name
    public object objValue; // Current value
}

// ArrayList of above items
System.Collections.ArrayList lstTags = new System.Collections.ArrayList();

// Declare structure for queue of browse requests
struct tQueueEnt
{
    public string strPath; // Item path
    public string strItem; // Item name
    public TreeNode refNode; // Reference to parent node where must add
};

/*
 * Stored data for subscription handling
 */
private string strSubHandle = ""; // Subscription handle from svr.
private string strCliHandle = "sub"; // Client handle
bool bRequestInProgress = false; // Flag - poll request active?
```

## 15.5.2.2.2 Initialisation

When the form is constructed, an event handler for the 'Subscription polled refresh' result is added to the XML-DA web service, and the service URL is set again to match the contents of the URL textbox. This is done by simulating a click on the 'Reload' button.

```
public frmXML-DA()
{
    // Construction
    InitializeComponent();

    // Add 'subscription poll complete' event
    srvXML.SubscriptionPolledRefreshCompleted +=
        new XML-DA.SubscriptionPolledRefreshCompletedEventHandler
            (srvXML_SubscriptionPolledRefreshCompleted);

    // Set initial URL using 'reload' click subroutine below
    btnLoad_Click(this, new EventArgs() );
} // End frmXML-DA()

private void btnLoad_Click(object sender, EventArgs e)
{
    // Update web service with new URL
    try
    {
        srvXML.Url = txtURL.Text;
    }
    catch (Exception exc)
    {
        MessageBox.Show("Error '" + exc.Message + "' while changing URL");
    }
} // End btnLoad_Click()
```

## 15.5.2.2.3 GetStatus() method

This method returns the status of the target XML-DA server.

```
private void btnStatus_Click(object sender, EventArgs e)
{
    // Variables for out parameters
    XML-DA.ServerStatus Status; // Server status

    try
    {
        // Get server status
        srvXML.GetStatus("en", // (in) Locale ID
            "1", // (in) Client handle
            out Status); // (out) Server status

        labStatus.Text = "GetStatus() OK - vendor '" + Status.VendorInfo
            + "', version '" + Status.ProductVersion + "'";
        MessageBox.Show(labStatus.Text);
    }
    catch (Exception exc)
    {
        labStatus.Text = "GetStatus() error " + exc.Message;
        MessageBox.Show(labStatus.Text);
    }
} // End btnStatus_Click()
```



## 15.5.2.2.4 Browse() method

The 'Browse()' method can be used to find the details of the server hierarchy, and build a list of tags.

As each 'Browse()' call will only return one section of the hierarchy at a time, it is necessary to make many 'Browse()' calls to read the full hierarchy tree. In this example, a 'queue' of requests of type 'tQueueEnt' is used. One 'Browse()' call is made for each queue entry, and when a node with further child nodes is found, it is added to the end of the queue. When there are no more items on the queue, the full tree has been read. In fact, the order in which the queue is processed does not matter as long as all items are eventually read.

When a readable ('Item') node is found, it is added to the separate ArrayList 'lstTags' for use in the 'Read()' and 'Subscribe()' operations.

In this example, the nodes are also stored in a TreeView control for simple display. By storing a reference to the 'TreeNode' of the parent item in the 'tQueueEnt' structure, it is easy to find the insertion point for nodes. The 'TreeNode' 'Tag' property is used to store the index of the lstTags entry, for use with the 'Write()' call.

```
private void btnBrowse_Click(object sender, EventArgs e)
{
    // Clear existing list of usable tags
    lstTags.Clear();

    // Prepare treeview
    tvwBrowse.BeginUpdate();
    tvwBrowse.Nodes.Clear();

    // Queue of nodes with child items that still need browsing
    System.Collections.ArrayList qBrowse = new System.Collections.ArrayList();

    // Put root browse request on queue
    tQueueEnt qTop = new tQueueEnt();
    qTop.strPath = "";
    qTop.strItem = "";
    qTop.refNode = tvwBrowse.Nodes.Add("Root");
    qBrowse.Add(qTop);

    // Process browse queue items until queue is empty
    while (qBrowse.Count > 0)
    {
        // Remove first item from queue
        tQueueEnt tItem = (tQueueEnt) qBrowse[0];
        qBrowse.RemoveAt(0);

        // Use XML-DA 'browse' method to get list of items at this point
        // and add them to the treeview under the specified node
        try
        {
            // Declare variables to return results
            XML-DA.BrowseElement[] arrElems;
            XML-DA.OPCError[] arrErrs;
            bool bMoreElems = false;
            string strCont = null;    // Continuation point (not used)

            srvXML.Browse(
                null,                // Property names
                "en",                // Locale ID
                "1",                 // Client handle (not used in sample)
                tItem.strPath,       // Item path
                tItem.strItem,       // Item name
                ref strCont,         // Continuation point
                0,                   // Max elements (unlimited)
                XML-DA.browseFilter.all, // Browse Filter
                null,                // Element name filter
                null,                // Vendor filter
                false,               // Return all property values
                false,               // Return property values
                false,               // Return error text
                out arrElems,        // Items returned
                out arrErrs,        // Error data
                out bMoreElems       // Flag - more data available
            );
        }
    }
}
```

```

// Process results
foreach (XML-DA.BrowseElement tElem in arrElems)
{
    // Add to tree
    TreeNode tNew = tItem.refNode.Nodes.Add(tElem.Name);

    // If it has subitems, will need to browse for them
    if (tElem.HasChildren)
    {
        tQueueEnt tNewQ;
        tNewQ.refNode = tNew;
        tNewQ.strItem = tElem.ItemName;
        tNewQ.strPath = tElem.ItemPath;
        qBrowse.Add (tNewQ);

    } // End if (has sub-items)

    // If it is a readable item, store details for later use
    if (tElem.IsItem)
    {
        // Box index of new item into 'Tag' for later use
        int iIdx = new int();
        iIdx = lstTags.Count;
        tNew.Tag = (object)iIdx;

        // Add to tags list
        tTag tNewTag = new tTag();
        tNewTag.strItemName = tElem.ItemName;
        tNewTag.strItemPath = tElem.ItemPath;
        tNewTag.strName = tElem.Name;
        tNewTag.refNode = tNew;
        lstTags.Add(tNewTag);

    } // End if (is readable item)

    } // End foreach (element to process)

    labStatus.Text = "Browse() complete";

}
catch (Exception exc)
{
    labStatus.Text = "Browse() exception " + exc.Message;
    MessageBox.Show(labStatus.Text);
    break; // out of while
}

} // End while (more items on queue)

// Expand treeview, then allow display updates again
tvwBrowse.ExpandAll();
tvwBrowse.EndUpdate();

// Enable/disable buttons that depend on a tag list
btnRead.Enabled = (lstTags.Count > 0);
btnWrite.Enabled = btnRead.Enabled;
btnSubscribe.Enabled = btnRead.Enabled;

} // End btnBrowse_Click

```

## 15.5.2.2.5 Read() method

The 'Read()' call re-uses the tag list produced by browsing, and fetches the current value of each item before updating it in the tree.

The 'Read()' call is normally intended for one-off read operations. If you need to regularly poll for new data, the subscription mechanism described in ["Reading data using subscriptions"](#)<sup>[547]</sup> is more suitable.

```
private void btnRead_Click(object sender, EventArgs e)
{
    // Parameters used to return data from server
    XML-DA.RequestOptions objOpts = new XML-DA.RequestOptions();
    XML-DA.ReadRequestItemList lstItems = new XML-DA.ReadRequestItemList();
    XML-DA.ReplyItemList lstRItems;
    XML-DA.OPCError[] lstErrs;

    // Options - set locale and return item names/paths for convenience
    objOpts.LocaleID = "en";
    objOpts.ReturnItemName = true;
    objOpts.ReturnItemPath = true;
    objOpts.ReturnErrorText = true;

    // Create and fill in list of items to read
    lstItems.ItemPath = "";
    lstItems.Items = new XML-DA.ReadRequestItem[lstTags.Count];
    for (int i = 0; i < lstTags.Count; i++)
    {
        // Get ref. to tag data
        tTag tT = (tTag) lstTags[i];
        XML-DA.ReadRequestItem tI = new XML-DA.ReadRequestItem();
        tI.ItemName = tT.strItemName;
        tI.ItemPath = tT.strItemPath;
        tI.ClientItemHandle = i.ToString(); // Use index as handle
        lstItems.Items[i] = tI;
    }
}
```

```
// Call read method
try
{
    srvXML.Read(objOpts,          // Request options
               lstItems,         // Items list
               out lstRItems,    // Reply items list
               out lstErrs      // List of errors
            );

    // Prepare to update result tree
    tvwBrowse.BeginUpdate();

    int iErrs = 0; // Error count

    // Process results
    foreach (XML-DA.ItemValue tV in lstRItems.Items)
    {
        tTag tResultTag = (tTag)lstTags[int.Parse(tV.ClientItemHandle)];
        if (tV.ResultID == null)
        {
            tResultTag.objValue = tV.Value;
            tResultTag.refNode.Text = tResultTag.strName + " = "
                                     + tV.Value.ToString();
        }
        else
        {
            // Read error
            iErrs++;
            tResultTag.refNode.Text = tResultTag.strName + " (read error "
                                     + tV.ResultID.Name + ")";
        }
    } // End for (result items)

    // Finished updating result tree
    tvwBrowse.EndUpdate();

    labStatus.Text = "Read() completed with " +
                    ((iErrs == 0) ? "no" : iErrs.ToString()) + " errors";
    MessageBox.Show(labStatus.Text);
}
catch(Exception exc)
{
    labStatus.Text = "Read() exception " + exc.Message;
    MessageBox.Show(labStatus.Text);
}

} // End btnRead_Click()
```

## 15.5.2.2.6 Write() method

The 'Write()' method uses the stored 'Tag' property of the current select TreeNode to find the index of the OPC tag data in the 'lstTags' array. This holds the ItemPath and ItemName details for the item to write.

The value in the edit control is converted to an appropriate type by trial and error (in a real application, the type is usually known), and then writes this back to the server. More than one value can be written at once by providing a larger array of items, in the same way as the 'Read()' call.

```
private void btnWrite_Click(object sender, EventArgs e)
{
    // Find currently-selected node in treeview
    TreeNode tN = tvwBrowse.SelectedNode;
    if (tN != null && tN.Tag != null)
    {
        // Get index in tag list
        int iIdx = (int)(tN.Tag);

        // Use index to find tag entry
        tTag tT = (tTag) lstTags[iIdx];

        // Get new value and guess at type
        object objNewVal = null;
        try
        {
            // Try int first
            objNewVal = (object)Int32.Parse(txtWrite.Text);
        }
        catch
        {
            try
            {
                // Not int - try double
                objNewVal = (object) Double.Parse(txtWrite.Text);
            }
            catch
            {
                // Not double - try bool
                try
                {
                    objNewVal = (object)bool.Parse(txtWrite.Text);
                }
                catch
                {
                    // Default to string
                    objNewVal = (object) txtWrite.Text;
                }
            }
        }
    } // End conversion to object
}
```

```

// Build data for Write call
// Options - set locale and return item names/paths for convenience
XML-DA.RequestOptions objOpts = new XML-DA.RequestOptions();
objOpts.LocaleID = "en";
objOpts.ReturnItemName = true;
objOpts.ReturnItemPath = true;

// List of items to write - just one for this example
XML-DA.WriteRequestItemList lstItems = new XML-DA.WriteRequestItemList();
lstItems.Items = new XML-DA.ItemValue[1];
lstItems.Items[0] = new XML-DA.ItemValue();
lstItems.Items[0].Value = objNewVal;
lstItems.Items[0].ItemName = tT.strItemName;
lstItems.Items[0].ItemPath = tT.strItemPath;
lstItems.Items[0].ClientItemHandle = iIdx.ToString();

// Parameters used to return data from server
XML-DA.ReplyItemList lstRItems;
XML-DA.OPCError[] lstErrs;

// Attempt write
try
{
    srvXML.Write(objOpts,           // Options
                lstItems,          // Items to write
                false,              // Return items on reply
                out lstRItems,      // Reply items list
                out lstErrs        // OPC errors
                );

    // Check results
    if (lstRItems.Items[0].ResultID == null)
    {
        labStatus.Text = "Write() completed successfully";
        MessageBox.Show("Write successful");
    }
    else
    {
        labStatus.Text = "Write failed error "
            + lstRItems.Items[0].ResultID.Name;
        MessageBox.Show(labStatus.Text);
    }
}
catch(Exception exc)
{
    labStatus.Text = "Write failed error " + exc.Message;
    MessageBox.Show(labStatus.Text);
}

} // End if (anything with tag selected)
} // End btnWrite_Click()

```

## 15.5.2.2.7 Reading data using subscriptions

The 'subscription' mechanism of XML-DA is intended for use when the same tags will be read regularly. Unlike a straightforward 'Read()' call, using subscriptions allows the server to maintain more state information and even prepare a list of the tags that have changed since the last call – reducing the amount of bandwidth used, as only the changed tags need to be sent.

An initial 'Subscribe()' call is used to set up the subscription, with the server returning a handle which must be passed to later calls. The 'SubscriptionPolledRefresh()' method can be made at regular intervals to check for new data, and when the data is no longer needed the 'SubscriptionCancel()' method should be called.

A timer is used to call 'SubscriptionPolledRefresh()' regularly. In this implementation, a boolean flag is used to indicate that a request is in progress, in case the operation takes longer than the timer interval.

```
private void btnSubscribe_Click(object sender, EventArgs e)
{
    // Options - set locale and return item names/paths for convenience
    XML-DA.RequestOptions objOpts = new XML-DA.RequestOptions();
    objOpts.LocaleID = "en";
    objOpts.ReturnItemName = true;
    objOpts.ReturnItemPath = true;
    objOpts.ClientRequestHandle = strCliHandle;

    // Items to read - everything in tree
    XML-DA.SubscribeRequestItemList lstItems =
        new XML-DA.SubscribeRequestItemList();

    lstItems.ItemPath = "";
    lstItems.EnableBuffering = true;
    lstItems.RequestedSamplingRate = Int32.Parse(txtRate.Text);
    lstItems.RequestedSamplingRateSpecified = true;
    lstItems.Items = new XML-DA.SubscribeRequestItem[lstTags.Count];
    for (int i = 0; i < lstTags.Count; i++)
    {
        // Get ref. to tag data
        tTag tT = (tTag) lstTags[i];
        XML-DA.SubscribeRequestItem tI = new XML-DA.SubscribeRequestItem();
        tI.ItemName = tT.strItemName;
        tI.ItemPath = tT.strItemPath;
        tI.ClientItemHandle = i.ToString(); // Use index as handle
        lstItems.Items[i] = tI;
    } // End for (items to read)

    // Parameters for results from server
    XML-DA.SubscribeReplyItemList lstRItems;
    XML-DA.OPCError[] arrErrs;
```

```
// Prepare subscription request
try
{
    srvXML.Subscribe(objOpts,           // Request options
                    lstItems,          // Items to subscribe to
                    false,             // Return values on reply
                    60000,             // Subscription ping rate
                    out lstRItems,     // Reply list
                    out arrErrs,       // Errors
                    out strSubHandle   // Server subscription handle
                    );

    labStatus.Text = "Subscribe() succeeded, handle " + strSubHandle;
    MessageBox.Show(labStatus.Text);

    // Enable cancel, disable further subscribes
    btnCancelSub.Enabled = true;
    btnSubscribe.Enabled = false;

    // Enable timer to poll for changes
    timSub.Interval = Int32.Parse(txtRate.Text);
    timSub.Enabled = true;
}
catch(Exception exc)
{
    labStatus.Text = "Subscribe() failed error " + exc.Message;
    MessageBox.Show(labStatus.Text);
}
} // End btnSubscribe_Click()

private void btnCancelSub_Click(object sender, EventArgs e)
{
    // Stop timer
    timSub.Enabled = false;

    // Cancel subscription
    try
    {
        srvXML.SubscriptionCancel(strSubHandle, ref strCliHandle);

        labStatus.Text = "SubscriptionCancel() completed successfully";
        MessageBox.Show(labStatus.Text);
    }
    catch (Exception exc)
    {
        labStatus.Text = "SubscriptionCancel() failed error " + exc.Message;
        MessageBox.Show(labStatus.Text);
    }

    // Allow subscribe again
    btnCancelSub.Enabled = false;
    btnSubscribe.Enabled = true;
} // End btnCancelSub_Click()
```



```
private void timSub_Tick(object sender, EventArgs e)
{
    // Check for overruns
    if (bRequestInProgress)
    {
        // Last request has not finished yet
        // May be caused if poll rate is too fast for server
        labStatus.Text =
            "SubscriptionPolledRefresh() Warning - poll request overrun";
    }
    else
    {
        // Set flag - make no more requests until this one completes
        bRequestInProgress = true;

        // Options - set locale and return item names/paths for convenience
        XML-DA.RequestOptions objOpts = new XML-DA.RequestOptions();
        objOpts.LocaleID = "en";
        objOpts.ReturnItemName = true;
        objOpts.ReturnItemPath = true;
        objOpts.ReturnErrorText = true;

        // Create array with one handle in it
        string[] strHandles = new string[1] { strSubHandle };

        // Start asynchronous request for data, the result will
        // be returned through the completion event
        try
        {
            srvXML.SubscriptionPolledRefreshAsync(
                objOpts,           // Options
                strHandles,       // Sub handles
                DateTime.Now, false, // Hold time
                0,                // Wait time
                false              // Return all items
            );

            labStatus.Text = "SubscriptionPolledRefresh() initiated";
        }
        catch (Exception exc)
        {
            bRequestInProgress = false;
            labStatus.Text = "SubscriptionPolledRefresh() failed error "
                + exc.Message;
        }
    } // End if (no request already active)
} // End timSub_Tick()
```

```

private void srvXML_SubscriptionPolledRefreshCompleted(object sender,
    XML-DA.SubscriptionPolledRefreshCompletedEventArgs args)
{
    // This event is called when subscription polled refresh returns data
    if (!args.Cancelled)
    {
        try
        {
            int iErrs = 0; // Error count

            // Will have at most one reply list - depends on
            // no. of subscriptions polled at once
            if (args.RItemList.GetLength(0) > 0)
            {

                // Start drawing
                tvwBrowse.BeginUpdate();

                foreach (XML-DA.ItemValue tV in args.RItemList[0].Items)
                {
                    tTag tResultTag = (tTag) lstTags
                        [int.Parse(tV.ClientItemHandle)];
                    if (tV.ResultID == null)
                    {
                        tResultTag.objValue = tV.Value;
                        tResultTag.refNode.Text = tResultTag.strName + " = "
                            + tV.Value.ToString();
                    }
                    else
                    {
                        // Read error
                        iErrs++;
                        tResultTag.refNode.Text = tResultTag.strName
                            + " (read error " + tV.ResultID.Name + ")";
                    }
                }

                // End for (returned items)

                // Drawing complete
                tvwBrowse.EndUpdate();

                labStatus.Text = "SubscriptionPolledRefresh() completed with "
                    + ((iErrs == 0) ? "no" : iErrs.ToString()) + " errors";
            }
            else
            {
                labStatus.Text =
                    "SubscriptionPolledRefresh() completed, no data";
            }
        }
        catch (Exception exc)
        {
            labStatus.Text = "SubscriptionPolledRefresh() exception "
                + exc.Message;
        }

        // Ready for next request
        bRequestInProgress = false;
    } // End if (not cancelled)
} // End srvXML_SubscriptionPolledRefreshCompleted()

```

## 15.6 Troubleshooting

If the XML-DA configuration is not working, this section can be used to identify common problems and how to fix them.

### 15.6.1 Problems and solutions

This section will describe possible solutions to some of the problems that may occur when configuring XML-DA.

**Problem:** When connecting to the XML-DA server from a web browser, IIS reports error 500.

**Solution:**

1. Check that the virtual directory is set up correctly and points to the location where the web service wrapper was installed.
2. Check the 'virtual directory' properties in IIS to make sure that the correct user access has been configured for the directory. To do this, go to 'Start -> [Settings] -> Control Panel -> [Performance and maintenance] -> Administrative Tools -> Internet Information Services [XP] or Internet services manager [Win2K]'. Expand the tree in the left hand pane to find the name of your computer, then 'Web sites' and then the default web site (or the site in which XML-DA is configured, if this is different). Expand the web site entry, then find the virtual directory which was configured earlier ('XML-DA' in the example). Right click on this and select 'Properties', then switch to the 'Directory security' tab. In the 'Anonymous access and authentication control' frame, click 'Edit'. The setting in this dialog is usually 'Anonymous access' with the default IIS user name 'IUSR\_(COMPUTER)'.

**Problem:** When attempting to open a .asmx page, the page is not shown and instead there is a prompt to open or save the .asmx file.

**Solution:** ASP.NET may not be correctly registered with Internet Information Server. Follow the instructions in ["Adding ASP.NET support to a new installation of Internet Information Server"](#)<sup>[489]</sup> to fix this.

**Problem:** Although the test web page shows correctly from Internet Explorer, the XML-DA service still does not work from client applications.

**Solution:**

1. Make sure that the user account used for the web service has sufficient access rights to start and use the OPC server. Check the event log for error messages (see next section). If you are using XP service pack 2, make sure that the DCOM 'Edit limits' settings are not overriding the account settings.
2. Check that the web service .asmx file matches the 'progID' of the server (i.e. Mitsubishi.MXOPC.6 or Mitsubishi.MXOPC) if no config file is used. If there is a .config file to match the .asmx file, check that its settings refer to the correct computer name and 'progID'.

**Problem:** XML-DA connection works on the local PC, but not from another PC.

**Solution:**

1. Check that there is a working network connection between the PCs.
2. Check the firewall settings to make sure that the port number you are using for the server (the default is 80 'HTTP') is not being blocked, either on the server PC or the client PC.

**Problem:** Reading items returns error 'E\_UNKNOWNITEMPATH'.

**Solution:** Ensure that the 'ItemPath' parameter is set to "" (empty string) if not used. It can no longer be left to take the default value.

**Problem:** Reading items throws a SOAP exception described as 'bad character in XML'.

**Solution:** If you are reading strings, make sure that the strings do not contain any characters that are not supported in the XML stream used by SOAP to return the response.

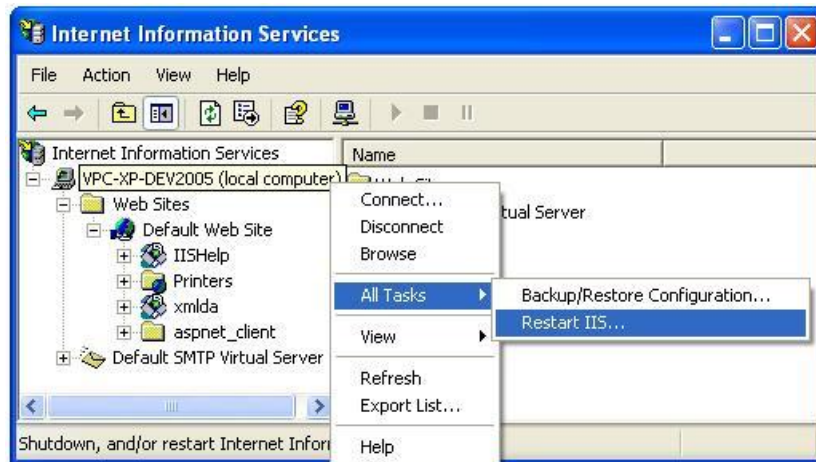
**Problem:** Other general issues

**Solution:** Try restarting Internet Information Server. To do this, run the IIS configuration tool from:

Start -> [Settings] -> Control Panel -> [Performance and Maintenance ->] Administrative Tools -> Internet Information Services [XP] or Internet services manager [Win2K]

The 'Settings' part is only needed on Windows 2000, and the 'Performance and Maintenance' part will only be present on XP if control panel is set up to display in 'Category view' instead of 'Classic view'.

In the dialog that appears, find your computer name in the list, then right click on it and select 'Restart IIS...' (on Windows 2000) or 'All tasks -> Restart IIS' (on XP, as in the picture below).



If restarting IIS does not solve the problem, try restarting your computer.

### 15.6.2 Using event viewer for troubleshooting

When XML-DA is not working as expected, it is often possible to get additional diagnostic information from the Windows event logs. To start the event viewer, select 'Start->Run' and enter 'eventvwr' before clicking 'OK', or on later operating systems click the Windows icon (or start button) and enter 'eventvwr' in the search box. Check each of the available logs (Application, Security, etc.) for error or warning messages that occurred around the time that XML-DA or DCOM access was being attempted.

Some example errors are shown in the table below.

Source	Description	Notes
DCOM	The machine-default permission settings do not grant Local Activation permission for the COM Server application with CLSID ( <i>CLSID</i> ) to the user ( <i>USER</i> ). This security permission can be modified using the Component Services administrative tool.	Messages similar to this suggest that the web service may be running as a user without sufficient COM access rights to start or access the OPC server.  Either use a different user account, or assign OPC rights to the account as described in " <a href="#">COM configuration</a> " <sup>[505]</sup> .
DCOM	DCOM was unable to communicate with the computer ( <i>COMPUTER</i> ) using any of the configured protocols.	If you are using a remote DCOM connection, the target computer could not be accessed. There may be a problem with the DCOM or security configuration, or a problem with the network connection or routing between the two computers. It may just be that the computer name has been mistyped.  Also remember that DCOM cannot cross network address translation (NAT) boundaries and can be stopped by certain network switches and routers.
W3SVC	The server was unable to logon the Windows NT account '( <i>ACCOUNT</i> )' due to the following error: Logon failure: user account restriction. Possible reasons are blank passwords not allowed, logon hour restrictions, or a policy restriction has been enforced. The data is the error code.	IIS could not use the specified account due to policy restrictions (e.g. 'Accounts: Limit local account use of blank passwords to console logon only').  Find the affected policy and either comply with it (in the above example, set a non-blank password) or if it is permitted by your security standards and reasonable to do so, relax the policy using the Group policy editor (gpedit.msc).



# Index

## - . -

.mxcsv 224  
.NET 488

## - 0 -

0x00000000 374  
0x00000001 374  
0x80004003 374  
0x80004005 374  
0x8000FFFF 374  
0x80070005 374  
0x80070006 374  
0x8007000E 374  
0x80070057 374

## - 1 -

184 device type (Modbus) 84, 87

## - 3 -

384 device type (Modbus) 84, 87

## - 4 -

484 device type (Modbus) 84, 87

## - 5 -

584 device type (Modbus) 84, 87  
584L device type (Modbus) 84, 87

## - 8 -

884 device type (Modbus) 84, 87

## - 9 -

984 device type (Modbus) 84, 87

## - A -

A2AS 103

A6TEL 103  
About application (Help menu) 63  
Access rights 38, 105, 116  
Accessing array items 157  
Accessing bits 158  
ACT error codes 375  
Activating the Database 130, 145  
Active configuration bar (View menu) 54  
Active database bar 173, 193  
Active file 56  
Add Point Calls 150  
Adding a New Group 125  
Adding a new Modbus device 87  
Adding a New MX Data Tag 116  
Adding a new MX device 77  
Adding labels to iQ Works 211  
Additional device statistics 150  
Address ranges 108  
Address space 141, 148, 151  
    Configuring 64  
Address space (Go menu) 55  
Addressing 108  
Administrator 14  
Advanced device properties 69  
Advanced digital alarm properties 136  
Advanced Limit Alarm Properties 132  
Advanced Modbus Data Tag Properties 119  
Advanced Modbus Device Properties 86  
Advanced MX Data Tag Properties 109  
Advanced options 151  
Advanced tab (Options) 58  
AE 9  
Aggregation 76  
AJ71C24 103  
AJ71E71 307, 419  
AJ71QC24N 103  
AJ71QE71 307, 419  
AJ71UC24 103  
Alarm and events 10  
Alarm definitions 130  
Alarm definitions (Go menu) 55  
Alarm state value 134  
Alarm/Events 9  
Alarms 38, 113  
    Deadband 132  
    Definitions 130  
    Digital 130, 134, 135, 136  
    Limit 130, 131, 132, 133  
    Message body 135  
    Requires ack 135  
    Return to normal 135  
    Severity 135, 136  
    Update rate 132, 133, 136

Amplitude 138  
 Array element size 212  
 Array items  
   Reading 157  
   Writing 157  
 Array lower bound 212  
 Arrays 155, 157  
 Arrays of DUT variables 188  
 ASCII 89  
 ASP.NET 489  
 Assigning conversion definitions to tags 168  
 Assigning poll method definitions to tags 172  
 Assignment project (iQ Works) 193  
 Automatic backup 162  
 Automation interface 362

## - B -

Back (Go menu) 55  
 Back up 191  
 Back up to MXChange 191  
 Backup device 36, 69, 77  
 Backup to File (File menu) 49  
 Backwards compatibility 9  
 Base text 114, 121  
 Basic Device properties 65  
 Basic Limit Alarm Properties 131  
 Basic Modbus Data Tag Properties 118  
 Basic modbus properties 84, 89, 90  
 Basic MX Data tag properties 105  
 Basic poll method properties 170  
 Basic simulation signal properties 138  
 Baud rate 25, 89  
 Bit access for integers 105, 106, 158  
 Bit field 118  
 Bit notation 158  
 Bit num 118  
 Bits 106, 158  
   Reading/writing within a tag 158  
   Reading/writing within a word 158  
 Bits within tags 105, 106, 158  
 Bitwise access 158  
 Blank addresses 190  
 Blank addresses (iQ Works) 213  
 Block Poll Count 150  
 Board number 95  
 Browse for MXChange database 176  
 Browse for MXChange server 174, 176  
 Browse for workspace 195, 198  
 Browse specific path 56

## - C -

C# 538  
 C24 285  
 C24 module 103  
 Cable 25  
 CC-Link 269, 419  
 CC-Link IE 258  
 CD-ROM 13, 14  
 Change MXChange password 180  
 Change notification (iQ Works) 217  
 Change password (MXChange database actions) 50, 180  
 Check PLC ranges 76  
 Checksum 95  
 Choose destination location 18  
 Clamp on EU 165  
 Clamping 165  
 Clamping range 165  
 Clamping type 166  
 Client count 163  
 Client scale 165  
 Close (iQ Works workspace actions menu) 202  
 Close (MXChange database actions menu) 179  
 Close (MXChange database actions) 50  
 Close workspace 202  
 Close workspace (iQ Works workspace actions menu) 51  
 Coil 118  
 Coils 90  
 Collapse item (Go menu) 55  
 COM Port 25  
 Comment 69, 212, 350  
 Communication setting wizard 36, 95  
 Communication settings wizard 26  
 Communications channels  
   Creating 26  
 Compact 62  
 Compact (MXChange database actions) 50, 180  
 Compact and repair (Tools menu) 62  
 Compacting and repairing Microsoft Access databases 62  
 Compacting the MXChange database 180  
 Compatibility 9  
 Configuring DCOM security  
   Windows 7 438  
   Windows Server 2008 438  
   Windows Vista 438  
   Windows XP 458  
 Configuration change logging 60  
 Configuration database 9  
 Configuration databases



- Configuration databases
  - Creating 46
- Configurator 13
  - Starting 24, 45
- Configuring a new device 36
- Configuring Alarm Definitions 130
- Configuring conversion definitions 165
- Configuring DCOM
  - Windows 7 431
  - Windows Server 2008 431
  - Windows Vista 431
- Configuring OPC server service 21
- Configuring poll method definitions 169
- Configuring service logon
  - Windows 7 452
  - Windows Server 2008 452
  - Windows Vista 452
  - Windows XP 471
- Configuring simulation signals 138
- Configuring the address space 64
- Connect module 95
- Connect project to current workspace 195
- Connect to an existing iQ Works workspace 195, 198
- Connect to an existing MXChange database 176
- Connect to MXChange 174
- Connecting to a new MXChange database 174
- Connecting to iQ Works 193
- Connecting to MXChange 173
- Connecting to the server with OPC DataSpy 42
- Connection status 173, 193
- Connection test 225
- Consecutive Faults 150
- Conversion definition 165
- Conversion definitions 109, 119, 165, 166, 168
- Conversion definitions (Go menu) 55
- Conversion properties 165
- Conversion type 165, 166
- Convert to word 109, 119, 168
- Copy (Edit menu) 52
- Copy to project 208, 214
- CPU 278
- CPU board 278
- CPU type 77, 95
- CPU, module and network board errors 419
- Create new items 128
- Creating a new conversion definition 166
- Creating a new digital alarm definition 136
- Creating a new limit alarm definition 133
- Creating a new poll method definition 171
- Creating a new simulation signal 139
- Creating a serial communication channel 26
- Creating an ethernet communication channel 31

- Creating communications channels 26
- Creating configuration databases 46
- Creating data tags 38
- Creating iQ Works labels 211
- Creating new transfer setups 92
- CSV 9
- CSV database load 58
- CSV Export (File menu) 49, 127
- CSV file 9
- CSV Import (File menu) 49, 128
- CTS signal 89
- Custom device type (Modbus) 84, 87
- Customer information 17
- Cut (Edit menu) 52

## - D -

- DA 9
- Data access 9, 10
- Data bits 25, 95
- Data conversion 116
- Data conversions 109, 119
- Data length 118
- Data manipulation (View menu) 54
- Data manipulation toolbar 48
- Data tag properties 38, 105, 116
- Data tag statistics 148
- Data tags 24, 64, 105, 141, 172
  - Creating 38
  - Properties 38
- Data type 105, 116, 212
- Data types 109
- Database 46
  - Activating 130, 145
- DataSpy 142
- DCOM configuration
  - Before you begin 421
  - Compatibility 420
  - Configuring DCOM security 424
  - Configuring DCOM security (Windows 7) 438
  - Configuring DCOM security (Windows XP) 458
  - Configuring for a specific user account (Windows 7) 452
  - Configuring for a specific user account (Windows Server 2008) 452
  - Configuring for a specific user account (Windows Vista) 452
  - Configuring for a specific user account (Windows XP) 471
  - Configuring the windows firewall (Windows 7) 432
  - Configuring the windows firewall (Windows Server 2008) 432

- DCOM configuration
    - Configuring the windows firewall (Windows Vista) 432
    - Configuring the windows firewall (Windows XP) 455
    - Introduction 420
    - Security considerations 420
    - User accounts 422
    - Windows 7 431
    - Windows Server 2008 431
    - Windows Vista 431
    - Windows XP 454
  - Deadband 109, 132, 133
  - Delay time 69, 86
  - Delete (Edit menu) 52
  - Delete (MXChange database actions) 50, 181
  - Delete OPC project (iQ Works project actions menu) 51, 202
  - Delete project (MXChange project actions menu) 50, 179
  - Delete unused projects (iQ Works workspace actions menu) 51, 223
  - Deleting the MXChange database 181
  - Description 118
  - Detaching from iQ Works 202
  - Detaching from MXChange 179
  - Details (View menu) 54
  - Device 64
  - Device layout 26, 77, 95
  - Device names 52
  - Device properties 36, 65, 84, 87
    - Advanced 69
    - Basic 65
  - Device ranges 106, 110
  - Device scale 165
  - Device statistics 148
  - Device statistics (additional) 150
  - Device tuning 76
  - Device Type
    - 184 84, 87
    - 384 84, 87
    - 484 84, 87
    - 584 84, 87
    - 584L 84, 87
    - 884 84, 87
    - 984 84, 87
    - Custom 84, 87
    - Micro 84 84, 87
    - ModCell 84, 87
    - Quantum 84, 87
  - Devices 24, 64, 141
  - Diagnostics 163
  - Diagnostics (View menu) 54
  - Dial 103
  - Digital alarm 113, 130
  - Digital alarm definition 113
  - Digital alarm definitions 134
  - Digital alarms 135, 136
  - Directory
    - Installation 18
    - Target 18
  - Directory for log files 60
  - Disable monitoring 155
  - Disconnecting from iQ Works 202
  - Disconnecting from MXChange 179
  - Display errors 128
  - Display hints 45, 56
  - Downward compatibility 9
  - Driver
    - Setting up 25
    - Starting 145
    - Stopping 145
  - DTR 95
  - DUT 188
  - DUT arrays 188
  - DUT variables 190
    - Limitations 190
    - Restrictions 190
  - Dynamic tag support 159
  - Dynamic tags 159, 162, 366
- E -**
- E\_ACCESSDENIED 374
  - E\_FAIL 374
  - E\_HANDLE 374
  - E\_INVALIDARG 374
  - E\_OUTOFMEMORY 374
  - E\_POINTER 374
  - E\_UNEXPECTED 374
  - Earlier versions 9
  - EasySocket 19
  - Edit menu 52
    - Copy 52
    - Cut 52
    - Delete 52
    - Invert selection 52
    - Paste 52
    - Properties 52
    - Select all 52
  - EG 203
  - Elements (array) 212
  - Elements of a structure 221
  - Enable 84, 87
  - Enable device 69
  - Enable group 124, 125
  - Enable hover selection 56

Enable monitoring 154  
 Enable primary rate 171  
 Enable tag 38, 119  
 End (Go menu) 55  
 Ending address 108  
 Engineering Units (EU) 165  
 Errors 148  
   0x00000000 374  
   0x00000001 374  
   0x01010000 to 0x010FFFFFF 419  
   0x80004003 374  
   0x80004005 374  
   0x8000FFFF 374  
   0x80070005 374  
   0x80070006 374  
   0x8007000E 374  
   0x80070057 374  
   ACT 375  
   CPU 419  
   E\_ACCESSDENIED 374  
   E\_FAIL 374  
   E\_HANDLE 374  
   E\_INVALIDARG 374  
   E\_OUTOFMEMORY 374  
   E\_POINTER 374  
   E\_UNEXPECTED 374  
   HRESULT 374  
   Module 419  
   Network board 419  
   S\_FALSE 374  
   S\_OK 374  
   Understanding 373  
 Ethernet 65, 236, 278, 307, 419  
 Ethernet communication channel 31  
   Creating 31  
 Ethernet Device 84, 87  
 EU (Engineering Units) 165  
 Execute verification synchronous 208, 214  
 Exit (File menu) 49  
 Expand item (Go menu) 55  
 Export devices to GID (MXChange project actions menu) 182  
 Export devices to GX Works2 (iQ Works project actions menu) 51, 203  
 Export to GX Developer 182  
 Export to GX IEC Developer 182  
 Export to iQ Works 203  
 Export to MXChange 174  
 Exporting Configuration Data to a CSV File 127  
 Express DCOM configuration 424  
 Extended port 296

## - F -

File import results 128  
 File menu 49  
   CSV Export 127  
   CSV Import 128  
   iQ Works project actions 49, 51  
   iQ Works workspace actions 49, 51  
   MXChange database actions 50  
   MXChange project actions 49, 50  
   New 46  
 File name 90  
 Find CPU 278  
 Firewalls 432, 455, 523, 524, 526  
 First number 114, 121  
 Folder 188  
 Folders 38, 123  
 Forward (Go menu) 55  
 FX 296  
 FX2N 103  
 FXCPU 103  
 FX-ENET 307

## - G -

G4 Module 321  
 Genbroker 142  
 General tab (Options) 56  
 Generate alarms 38, 113, 116, 119  
 Genesis 12  
 Go menu 55  
   Address space 55  
   Alarm definitions 55  
   Back 55  
   Collapse item 55  
   Conversion definitions 55  
   End 55  
   Expand item 55  
   Forward 55  
   Home 55  
   Next item 55  
   Next pane 55  
   Page down 55  
   Page up 55  
   Poll method definitions 55  
   Previous item 55  
   Previous pane 55  
   Simulation definitions 55  
   Up one level 55  
 GOT 307  
 Group 188

Group count 163  
 Group properties  
   Advanced tab 124, 125  
   Basic tab 124, 125  
 Groups 38, 123  
 GX Developer 173, 182, 184  
 GX IEC Developer 53, 173, 182, 184, 188, 190  
 GX Simulator 273  
 GX Works2 193, 203, 210, 217, 219, 220

## - H -

Hard drive space 13  
 Help menu 63  
   About application 63  
   Help topics 63  
 Help topics (Help menu) 63  
 Hints 45, 56, 164  
 Historical data access 10  
 HMI 10  
 Holding register 118  
 Holding registers 90  
 Home (Go menu) 55  
 Host 65, 330  
 Hover time 56  
 HRESULT error codes 374

## - I -

I/O address 38, 105, 116  
 I/O driver 11, 24, 141  
 I/O server 9, 141  
 iFIX 12  
 IIS (Internet Information Server) 474  
 Import 184  
 Import CSV (iQ Works project actions menu) 51, 224  
 Import from iQ Works 210  
 Import GID projects (MXChange project actions menu) 50, 184  
 Import GX Works2 project (iQ Works project actions menu) 51, 210  
 Importing Configuration Data from a CSV File 128  
 Importing from MXChange 184  
 Incorrect password (MXChange) 177  
 Input register 118  
 Input registers 90  
 Insert missing parent items 128  
 Install as service 18  
 Installation 13, 14  
 Installation complete 20  
 Installing Internet Information Server (IIS) 479  
   Set up security for XML-DA 499

Windows 7 481  
 Windows Server 2008 482  
 Windows Vista 481  
 Windows XP 480  
 Installing MX OPC server 13, 14  
 Instrument Range 165  
 Instrument Range (IR) 165  
 Internet explorer 60  
 Internet Information Server (IIS) 474  
 inTouch 12  
 Introduction 9  
 Invert selection (Edit menu) 52  
 IP Address 84, 87  
 iQ Works 51, 223  
   Adding labels 211  
   Assignment project 193  
   Blank addresses 213  
   Browse workspace 195, 198  
   Change notification 217  
   Connecting 193, 195, 198  
   Creating labels 211  
   Deleting unused projects 223  
   Detaching 202  
   Disconnecting 202  
   Exporting 203  
   Importing 210  
   Linked MX devices 193  
   Local workspace 195  
   New labels 211  
   Reference project 193  
   Remote workspace 198  
   Removing 202  
   Restrictions 193, 216  
   Select devices to export 203  
   Structure support 221  
   Structures 216  
   Supported labels 203  
   Switch to MXChange 223  
   Synchronising 219  
   System label list 193, 204  
   Unallocated addresses 213  
   Unlinking 193  
   Unlinking devices 220  
   Unsupported labels 203  
   Using 193  
   Workspace 193  
 iQ Works project actions  
   Delete OPC project 202  
   Export devices to GX Works2 203  
   Import CSV 224  
   Import GX Works2 project 210  
   Show changes 218  
   Synchronise 219

- iQ Works project actions
  - Unlink device 220
- iQ Works project actions (File menu) 49, 51
- iQ Works project actions menu
  - Delete OPC project 51
  - Export devices to GX Works2 51
  - Import CSV 51
  - Import GX Works2 project 51
  - Show changes 51
  - Synchronise 51
  - Unlink device 51
- iQ Works workspace actions
  - Close 202
  - Delete unused projects 223
  - Open workspace 193
- iQ Works workspace actions (File menu) 49, 51
- iQ Works workspace actions menu
  - Close workspace 51
  - Delete unused projects 51
  - Open workspace 51
- IR (Instrument Range) 165
- Item IDs 159
- Item names 52

**- J -**

- JavaScript 531

**- L -**

- LabView 12
- Large icons (View menu) 54
- Last error 148
- Last error time 148
- Last Read Time 150
- Last receive time 148
- Last transmit time 148
- Last update time 163
- Last Write Time 150
- Limit 135
- Limit alarm 113, 130
- Limit alarm definition 113
- Limit alarm definitions 131
- Limit alarms 119, 131, 132, 133
- Limit on number of system labels 216
- Limitation 76
- Limitations on DUT variables 190
- Linear 165
- List (View menu) 54
- List View 64
- LLT 273
- Load CSV 58

- Load MDB 58
- Load project from MXChange (MXChange project actions menu) 50, 191
- Local workspace 195
- Location type 118
- Log files 60
- Logging 60
  - Configuration 60
  - Directory 60
  - OPC layer 60
  - Protocol and wire 60
  - User changes 60
  - Wire 60
- Logging (Options) 60
- Logging configuration changes 60
- Lower bound (array) 212

**- M -**

- Make active 145
- Make float 165
- Managing MXChange databases 179
- Manual value 109
- Master station 95
- Max count for R/W operation 69
- Max diagnostic messages 58
- Max request count 76
- Max transaction count 76
- MDB 9
- MDB file 9
- MELSECNET/10 95, 264, 419
- MelsecNet/B 95
- MELSECNET/H 261, 419
- MelsecNet/II 95
- Memory 13
- Menu bar 48
- Menus 48
  - Edit 52
  - File 49
  - Go 55
  - Help 63
  - Tools 56
  - View 54
- Message body 113, 131, 135
- Message prefix 113, 119
- Micro 84 device type (Modbus) 84, 87
- Microsoft .NET 488
- Microsoft Access Database 46
- Minimum read rate 76
- Missing GX Works2 project 204
- Missing projects 194
- Modbus 50, 64, 84, 86, 87, 90, 119, 121, 173
- Modbus device

Modbus device  
   Adding 87  
 Modbus device parameters 90  
 Modbus Device Properties  
   Advanced 86  
 Modbus ethernet 64  
 Modbus serial 64  
 Modbus serial port details 89  
 Modbus tag 121  
 Modbus type 118, 119  
 ModCell device type (Modbus) 84, 87  
 Mode 95  
 Modem 248, 249, 302, 333  
 Modem initialisation string 103  
 Modem support 103  
 Modems 103  
 Module list 225  
 Module type 65, 95  
 Monitor  
   Disabling 155  
   Enabling 154  
 Monitor CTS for output flow control 89  
 Monitor view 153, 154  
 Monitor view (View menu) 54  
 Monitoring 153  
 Monitoring data 142  
 Multi-cpu 351  
 Multiply 114, 121  
 Multiply (Modbus Tags) 121  
 MX Component 10, 19  
 MX Data Tag Properties  
   Alarms 113  
 MX device  
   Adding 77  
 MX OPC Configurator 11, 24, 45  
 MX OPC Runtime 10, 24, 41  
 MX OPC Server 9  
 MX OPC server runtime module 141  
 MX Runtime 9, 24  
 MX Transfer Setup dialog 26, 77  
 MXChange 50, 173, 174, 176, 179, 180, 181, 182, 184, 186, 188, 191, 223  
   Incorrect password 177  
   Switch to iQ Works 223  
 MXChange actions (File menu) 49  
 MXChange database actions 50  
   Change password 50, 180  
   Close 50, 179  
   Compact 50, 180  
   Delete 50, 181  
   New 50, 174  
   Open 50, 176  
   Save as 50, 181

MXChange database actions (File menu) 50, 179  
 MXChange project actions  
   Delete project 50, 179  
   Export devices to GID 182  
   Import GID projects 50, 184  
   Load project from MXChange 50, 191  
   Show changes 50, 186  
   Synchronise 50, 186  
 MXChange project actions (File menu) 49, 50  
 MXChange project actions menu  
   Import GID projects 184  
 MXChange server 19  
 MXChange2 193, 223  
 MXComponent 145  
 MXConfigurator.mdb 24  
 MXCSV files 224

## - N -

NAT (Network address translation) 420, 473, 553  
 Network 198, 330  
 Network address translation (NAT) 420, 473, 553  
 Network number 95  
 New (File menu) 46, 49  
 New (MXChange database actions) 50, 174  
 New conversion definition 166  
 New data tag 38, 116  
 New database dialog 174  
 New iQ Works label(s) 211  
 New iQ Works labels 211  
 New Modbus device 87  
 New MX Device 26  
 New polling method 171  
 New transfer setup 225  
 Next item (Go menu) 55  
 Next pane (Go menu) 55  
 No unused projects were found 223  
 Number of Data Blocks 150  
 Number of elements 118  
 Number of items 114, 121  
 Number of Points 150  
 Number of retries 69, 86  
 Number of Steps 138  
 Numeric places 114, 121

## - O -

OLE for process control 10  
 Online changes 174, 176  
 Online changes rate 58  
 OPC 9, 10, 362  
 OPC architecture 10

- OPC client 9, 141, 142
  - OPC client software 12
  - OPC condition 135
  - OPC Data Access 9
  - OPC data spy 12
  - OPC DataSpy 23, 42, 142
  - OPC layer logging 60
  - OPC monitoring refresh rate 56, 154
  - OPC project details 174, 176
  - OPC project node 176
  - OPC project node dialog 174
  - OPC server 141
  - OPC subcondition 135
  - OPC\_DS\_\* 364
  - OPC\_LIMIT\_\* 364
  - OPC\_LIMIT\_MASK 364
  - OPC\_QUALITY\_\* 364
  - OPC\_STATE\_\* 364
  - OPC\_STATUS\_MASK 364
  - Open (File menu) 49
  - Open (MXChange database actions) 50, 176
  - Open active file 56
  - Open database dialog 176
  - Open recently used file 56
  - Open specific file 56
  - Open workspace (iQ Works workspace actions menu) 51, 193
  - Open workspace dialog 195, 198
  - Optimization 58, 76
  - Optimizations 86
  - Options 56
    - Advanced tab 58
    - General tab 56
    - Logging 60
  - Options (Tools menu) 56
  - Options dialog 151
  - Organisation 17
  - Other station 95, 330, 341
  - Output flow control 89
  - Overruns 148
- P -**
- Packet type 95
  - Page down (Go menu) 55
  - Page up (Go menu) 55
  - Parity 25, 95
  - Parity checking 89
  - Password 350
  - Paste (Edit menu) 52
  - PC side I/F 26, 69, 77, 95
  - Period 138
  - Phase 138, 170
  - PLC 10
  - PLC side I/F 26, 95
  - Poll Count 150
  - Poll method definition 170
  - Poll method definitions 169, 171, 172
  - Poll method definitions (Go menu) 55
  - Poll methods 169
  - Polling method definitions 105
  - Polling methods 105
  - Polling rate 86
  - Pop-up menus 48
  - Port number 95
  - Position 138
  - Previous item (Go menu) 55
  - Previous pane (Go menu) 55
  - Previous versions 9
  - Primary device 36, 69, 77
  - Primary polling rate 170, 171
  - Process control 9, 10
  - Process hardware 25
  - Processor 13
  - Project node was not found in this database 197, 201
  - Properties (Edit menu) 52
  - Protocol 95
  - Protocol and wire 60
  - Pulse 103
- Q -**
- Q series 271
  - Q00 353
  - Q00J 353
  - Q01 353
  - Q06 103
  - Q2AS 103
  - Q6TEL 103
  - QJ71C24 103
  - QJ71E71 307
  - QnPRH 354
  - QnUDE 278
  - Quantum device type (Modbus) 84, 87
  - Quick start 23
  - QuickFail 10, 69
- R -**
- Ramp 138
  - Random 138
  - Ratio 138
  - Read aggregation 76
  - Read Count 138

Read Point Calls 150  
 Read/write priority 76  
 Reading bits 158  
 Reading bits within tags 106, 158  
 Reading individual array items 157  
 Ready to install 19  
 Rebooting the PC 23  
 Receives 148  
 Recent projects (File menu) 49  
 Recently used file 56  
 Redundancy 354  
 Redundant 354  
 Reference project (iQ Works) 193  
 Refresh (View menu) 54  
 Refresh dynamic tags 162  
 Refresh rate 147, 154  
 Register ranges 164  
 Registering ASP.NET 489  
 Remote workspace 198  
 Remove iQ Works 202  
 Remove MXChange 179  
 Remove Point Calls 150  
 Remove transfer setup 225  
 Removing transfer setups 93  
 Rename 52  
 Rename transfer setup 225  
 Rename/Remove... 93  
 Renaming transfer setups 93  
 Repair 62  
 Replace transfer setup 225  
 Reply timeout 69, 86, 148  
 Requires ack 131, 135  
 Reset statistics 148  
 Restart computer 20  
 Restore 191  
 Restore from Backup (File menu) 49  
 Restore from MXChange 191  
 Restrictions on devices linked to iQ Works 193  
 Restrictions on DUT variables 190  
 Restrictions on editing labels (iQ Works) 216  
 Retries 69, 148  
 Return to normal 131, 135  
 Reuse 178  
 Reuse project 197  
 RS232 234  
 RSview 12  
 RTS 95  
 RTS flow control 89  
 RTU 89  
 Runtime 13, 41  
 Runtime logging 60  
 Runtime module 141  
 Runtime polling rate 58

## - S -

S\_FALSE 374  
 S\_OK 374  
 Save As (File menu) 49  
 Save as (MXChange database actions) 50, 181  
 Save MXChange database as 181  
 SCADA 159, 420, 473  
 Select all (Edit menu) 52  
 Select devices for export to iQ Works 203  
 Select local workspace 195  
 Select MXChange database 176  
 Select remote workspace 198  
 Select system label list 204  
 Select transfer setup 225  
 Select workspace 195, 198  
 Serial 234, 278, 285  
 Serial communication channel 26  
     Creating 26  
 Serial Device 84, 87  
 Serial number 17  
 Serial Port 84, 87, 89  
 Serial port details dialog 90  
 Server status 163  
 Service 13, 18, 21  
 Service installation 18  
 Setting the monitor view refresh rate 154  
 Setting the Statistics Mode Refresh Rate 147  
 Setting up the driver 25  
 Severity 131, 135, 136  
 Show / hide columns (View menu) 54  
 Show changes (iQ Works project actions menu) 51, 218  
 Show changes (MXChange project actions menu) 50, 186  
 Show register ranges for MX devices 56  
 Showing register ranges 164  
 Signal type 138, 139  
 Simple DCOM configuration 424  
 Simulate 69, 84, 87  
 Simulation 109, 139  
 Simulation definitions 138  
 Simulation definitions (Go menu) 55  
 Simulation signal 109, 119, 139  
 Simulation signals 138  
 Sine 138  
 Small icons (View menu) 54  
 Software licence agreement 17  
 Sort by (View menu) 54  
 Special item names 157  
 Square 138  
 Square Root 165



- Standard buttons (View menu) 54
- Standard toolbar 47
- Start button 41
- Start driver 145
- Start GX IEC Developer 53, 190
- Start menu 24, 45
- Start time 163
- Starting address 108, 118
- Starting and stopping the driver 145
- Starting the Configurator 45
- Starting the driver 41
- Starting the MX OPC configurator 24
- Starting the MX OPC Runtime Driver 41
- Startup settings 56
- Station number 95
- Station type 77, 95
- Statistical tags in OPC address space 151
- Statistics 373
  - Add Point Calls 150
  - Block Poll Count 150
  - Connected 151
  - Consecutive Faults 150
  - Data blocks 151
  - Data tag 148
  - Device 148
  - Device statistics (additional) 150
  - Last Read Time 150
  - Last Write Time 150
  - Number of Data Blocks 150
  - Number of Points 150
  - PointCount 151
  - Poll Count 150
  - QueEvents 151
  - QueReads 151
  - QueTrans 151
  - QueWrites 151
  - Read Point Calls 150
  - Receives 151
  - Remove Point Calls 150
  - Reset 148
  - Tags 151
  - Transmits 151
  - Update Count 150
  - Write Point Calls 150
  - Write Poll Count 150
- Statistics mode 146, 147
- Statistics mode (View menu) 54
- Statistics refresh rate 56
- Statistics tags 58, 151
- Statistics update rate 58
- Status bar (View menu) 54
- Step 138
- Stop bits 25, 89, 95

- Stop button 41
- Stop driver 145
- Stopping the driver 41
- Stopping the MX OPC Runtime Driver 41
- structure offset values 222
- Structure support (iQ Works) 221
- Structures (iQ Works) 216
- Stylesheet 60
- Subfolders 123
- Sum check 95
- Supported devices 106, 110
- Supported labels in iQ Works 203
- Swap word/byte order 109, 119, 168
- Switch between iQ Works and MXChange 223
- Switch to iQ Works 223
- Switch to MXChange 223
- Synchronise (iQ Works project actions menu) 51, 219
- Synchronise (MXChange project actions menu) 50, 186
- Synchronising with iQ Works 219
- Synchronising with MXChange 186
- System label list 204
- System label list limit 216
- System label list, selecting 204
- System label name 212
- System requirements 13

## - T -

- Tag properties 114, 168
  - Advanced 109
  - Basic 105
- Target directory 18
- TCP Port 84, 87
- test 229
- Testing 42
- Timeout 97
- Timeouts 148
- Tone 103
- Toolbars 47
  - Data manipulation 48
  - Standard 47
- Toolbars (View menu) 54
- Tools menu 56
  - Compact and repair 62
  - Options 56
- Tracking cable 354
- Transaction close delay 58
- Transaction max. time 58
- Transfer setup 26, 77
- Transfer setup already exists 65
- Transfer setup editor 91

Transfer setup management 91  
 Transfer setups  
   Creating 92  
   Removing 93  
   Renaming 93  
 Transmission mode 89  
 Transmission speed 95  
 Transmission wait time 95  
 Tree View 64  
 Triangle 138  
 Troubleshooting XML-DA 551  
 Tuning 76  
 Turn UAC on or off 477  
 Type class 212

## - U -

UAC, turning on or off 477  
 UDP connections  
   Timeouts 97  
 Unallocated addresses 190  
 Unallocated addresses (iQ Works) 213  
 Unavailable device(s) 194  
 Understanding ACT error codes 373  
 UNICODE 13  
 Unit Identifier 84, 87  
 Unlink device (iQ Works project actions menu) 51, 220  
 Unlink device(s) 194  
 Unlinking devices from iQ Works 220  
 Unsupported labels in iQ Works 203  
 Up one level (Go menu) 55  
 Up time 163  
 Update Count 150  
 Update existing items 128  
 Update frequency 56  
 Update rate 132, 133, 136  
 Upgrade 9  
 USB 233  
 Use advanced types 109  
 Use as default polling rate 170  
 Use conversion 109, 119, 168  
 Use manual value 109, 119  
 User account control 14  
 User changes logging 60  
 User name 17  
 Using iQ Works 193  
 Using MXChange 173  
 Using MXChange2 193  
 Using MXCSV files 224  
 Using the configurator 44

## - V -

Valid data block registers 108  
 Validation 76  
 VBA 361  
 Vendor info 163  
 Verification synchronous 208, 214  
 Version 163  
 Versions  
   Previous 9  
 Video card 13  
 View menu 54  
   Active configuration bar 54  
   Data manipulation 54  
   Details 54  
   Diagnostics 54  
   Large icons 54  
   List 54  
   Monitor view 54  
   Refresh 54  
   Show / hide columns 54  
   Small icons 54  
   Sort by 54  
   Standard buttons 54  
   Statistics mode 54  
   Status bar 54  
   Toolbars 54  
 Virtual Directory Wizard 495  
 Virtual private network (VPN) 474  
 Vista 14  
 Visual Basic  
   Adding the OPC libraries to your project 363  
   Browsing OPC server items 366  
   Connecting to an OPC server 365  
   Disconnecting from an OPC server 372  
   Introduction 361  
   Listing installed OPC servers 365  
   OPC constants 364  
   Prerequisites 361  
   Reading data values 369  
   References 372  
   Terminology 362  
   Writing data values 371  
 Visual Studio 537  
 VPN (Virtual private network) 474

## - W -

Web service 537  
 Welcome screen 16  
 Window Server 2008

- Window Server 2008
    - Install Internet Information Server (IIS) 482
  - Windows 13
  - Windows 7
    - COM configuration 505
    - Configuring DCOM 431
    - Configuring DCOM security 438
    - Configuring Internet Information Server (IIS) 499
    - Configuring service logon 452
    - DCOM configuration 431
    - Firewall configuration 432
    - Install Internet Information Server (IIS) 481
    - Turning UAC on or off 477
  - Windows firewall 432, 455
    - Configuring for XML-DA 523
    - Configuring for XML-DA (Windows 7) 526
    - Configuring for XML-DA (Windows Server 2008) 526
    - Configuring for XML-DA (Windows Vista) 526
    - Configuring for XML-DA (Windows XP) 524
  - Windows NT service 18
  - Windows Server 2008
    - COM configuration 505
    - Configuring DCOM 431
    - Configuring DCOM security 438
    - Configuring Internet Information Server (IIS) 499
    - Configuring service logon 452
    - DCOM Configuration 431
    - Firewall configuration 432
    - Turning UAC on or off 477
  - Windows Update 490
  - Windows Vista 14
    - COM configuration 505
    - Configuring DCOM 431
    - Configuring DCOM security 438
    - Configuring Internet Information Server (IIS) 499
    - Configuring service logon 452
    - DCOM configuration 431
    - Firewall configuration 432
    - Install Internet Information Server (IIS) 481
    - Turning UAC on or off 479
  - Windows XP
    - COM configuration 515
    - Configuring DCOM security 458
    - Configuring Internet Information Server (IIS) 495
    - Configuring service logon 471
    - DCOM configuration 454
    - Firewall configuration 455
    - Install Internet Information Server (IIS) 480
  - Wire logging 60
  - Wizard 230
  - Wizards
    - Communication settings 26
  - Workspace settings 56
  - WR 203
  - Write aggregation 76
  - Write Count 138
  - Write Point Calls 150
  - Write Poll Count 150
  - Write to tag 155
  - Writing array items 157
  - Writing bits 158
  - Writing bits within tags 158
  - Writing individual array items 157
  - Writing to arrays 155
  - Writing to strings 155
  - Writing to tags 155, 157, 158
  - WW 203
- X -**
- XML 9, 60
  - XML data access 9
  - XML-DA 499
    - Add application dialog 502
    - Add Virtual Directory dialog 495, 499
    - Adding a web service reference 537
    - Adding ASP.NET support 489
    - C# sample - Browse() 541
    - C# sample - definitions 539
    - C# sample - GetStatus() method 540
    - C# sample - Initialisation 540
    - C# sample - read using subscriptions 547
    - C# sample - Read() 543
    - C# sample - shared code 539
    - C# sample - subscriptions 547
    - C# sample - Write 545
    - C# sample application 538
    - COM configuration 505
    - COM configuration (Windows 7) 505
    - COM configuration (Windows Server 2008) 505
    - COM configuration (Windows Vista) 505
    - COM configuration (Windows XP) 515
    - Configuring Internet Information Server (IIS) 494, 495, 499
    - Configuring the Windows firewall 523
    - Configuring the Windows firewall (Windows 7) 526
    - Configuring the Windows firewall (Windows Server 2008) 526
    - Configuring the Windows firewall (Windows Vista) 526

- XML-DA 499
  - Configuring the Windows firewall (Windows XP) 524
  - Configuring the XML-DA wrapper 493
  - Installing .NET framework v1.1 488
  - Installing Internet Information Server (IIS) 479
  - Installing Internet Information Server (IIS) on Windows 7 481
  - Installing Internet Information Server (IIS) on Windows Server 2008 482
  - Installing Internet Information Server (IIS) on Windows Vista 481
  - Installing Internet Information Server (IIS) on Windows XP 480
  - Installing the XML-DA wrapper 490
  - Introduction 473
  - Javascript 531
  - JavaScript sample application 535
  - Problems and solutions 551
  - Programming 531
  - Programming with JavaScript 531
  - Sample application structure (JavaScript) 535
  - Security considerations 474
  - Set up Internet Information Server (IIS) security 499
  - Testing 528
  - Troubleshooting 551, 553
  - Updating system components 490
  - Updating web page dynamically 534
  - Using event viewer for troubleshooting 553
  - Using from JavaScript 531
  - Using from Visual Studio 537
  - Using the XMLHttpRequest object 532
  - Virtual Directory Wizard 494, 495
  - Visual Studio 537
  - Web service 537
  - XMLHttpRequest object 532
- XSL 60

Microsoft, Windows, Windows 7, Windows Server 2008, Windows NT, Windows XP and Windows Vista are registered trademarks of Microsoft Corporation in the United States and other countries. Pentium is a registered trademark of Intel Corporation in the United States and other countries. Other company and product names herein are either trademarks or registered trademarks of their respective owners.

***MX OPC Configurator Version 6.05***

**Operating Manual**



Specifications subject to change without notice.



HEADQUARTERS		EUROPEAN REPRESENTATIVES		EUROPEAN REPRESENTATIVES		EURASIAN REPRESENTATIVES	
Mitsubishi Electric Europe B.V. German Branch <b>Gothaer Straße 8</b> D-40880 Ratingen Phone: +49 (0)2102 / 486-0 Fax: +49 (0)2102 / 486-1120	<b>EUROPE</b>	GEVA Wiener Straße 89 <b>A-2500 Baden</b> Phone: +43 (0)2252 / 85 55 20 Fax: +43 (0)2252 / 488 60	<b>AUSTRIA</b>	Beijer Electronics SIA Ritausmas iela 23 <b>LV-1058 Riga</b> Phone: +371 (0)6 / 784 2280 Fax: +371 (0)6 / 784 2281	<b>LATVIA</b>	TOO Kazpromavtomatika UL. ZHAMBYLA 28, <b>KAZ-100017 Karaganda</b> Phone: +7 7212 / 50 10 00 Fax: +7 7212 / 50 11 50	<b>KAZAKHSTAN</b>
Mitsubishi Electric Europe B.V. Czech Branch Radlická 751/113e Avenir Business Park <b>CZ-158 00 Praha 5</b> Phone: +420 251 551 470 Fax: +420 251 551 471	<b>CZECH REP.</b>	000 TECHNIKON Prospect Nezavisimosti 177-9 <b>BY-220125 Minsk</b> Phone: +375 (0)17 / 393 1177 Fax: +375 (0)17 / 393 0081	<b>BELARUS</b>	Beijer Electronics UAB Goštautų g. 3 <b>LT-48324 Kaunas</b> Phone: +370 37 262707 Fax: +370 37 455605	<b>LITHUANIA</b>	<b>MIDDLE EAST REPRESENTATIVE</b>	
Mitsubishi Electric Europe B.V. French Branch 25, Boulevard des Bouvets <b>F-92741 Nanterre Cedex</b> Phone: +33 (0)1 / 55 68 55 68 Fax: +33 (0)1 / 55 68 57 57	<b>FRANCE</b>	ESCO DRIVES Culliganlaan 3 <b>BE-1831 Diegem</b> Phone: +32 (0)2 / 717 64 60 Fax: +32 (0)2 / 717 64 61	<b>BELGIUM</b>	ALFATRADE Ltd. 99, Paola Hill <b>Malta-Paola PLA 1702</b> Phone: +356 (0)21 / 697 816 Fax: +356 (0)21 / 697 817	<b>MALTA</b>	I.C. SYSTEMS Ltd. 23 Al-Saad-Al-Alee St. <b>EG-Sarayat, Maadi, Cairo</b> Phone: +20 (0) 2 / 235 98 548 Fax: +20 (0) 2 / 235 96 625	<b>EGYPT</b>
Mitsubishi Electric Europe B.V. Irish Branch Westgate Business Park, Ballymount <b>IRL-Dublin 24</b> Phone: +353 (0)1 4198800 Fax: +353 (0)1 4198890	<b>IRELAND</b>	KONING & HARTMAN B.V. Woluwelaan 31 <b>BE-1800 Vilvoorde</b> Phone: +32 (0)2 / 257 02 40 Fax: +32 (0)2 / 257 02 49	<b>BELGIUM</b>	INTEHSIS SRL bld. Traian 23/1 <b>MD-2060 Kishinev</b> Phone: +373 (0)22 / 66 4242 Fax: +373 (0)22 / 66 4280	<b>MOLDOVA</b>	GIRIT CELADON Ltd. 12 Haomanut Street <b>IL-42505 Netanya</b> Phone: +972 (0)9 / 863 39 80 Fax: +972 (0)9 / 885 24 30	<b>ISRAEL</b>
Mitsubishi Electric Europe B.V. Italian Branch Viale Colleoni 7 Palazzo Sirio <b>I-20864 Agrate Brianza (MB)</b> Phone: +39 039 / 60 53 1 Fax: +39 039 / 60 53 312	<b>ITALY</b>	INEA RBT d.o.o. Stegne 11 <b>SI-1000 Ljubljana</b> Phone: +386 (0)1/513 8116 Fax: +386 (0)1/513 8170	<b>BOSNIA AND HERZEGOVINA</b>	HIFLEX AUTOM. B.V. Wolweverstraat 22 <b>NL-2984 CD Ridderkerk</b> Phone: +31 (0)180 / 46 60 04 Fax: +31 (0)180 / 44 23 55	<b>NETHERLANDS</b>	ILAN & GAVISH Ltd. 24 Shenkar St., Kiryat Ariet <b>IL-49001 Petah-Tikva</b> Phone: +972 (0)3 / 922 18 24 Fax: +972 (0)3 / 924 0761	<b>ISRAEL</b>
Mitsubishi Electric Europe B.V. Polish Branch ul. Krakowska 50 <b>PL-32-083 Balice</b> Phone: +48 (0) 12 630 47 00 Fax: +48 (0) 12 630 47 01	<b>POLAND</b>	AKHNATON 4, Andrei Ljapchev Blvd., PO Box 21 <b>BG-1756 Sofia</b> Phone: +359 (0)2 / 817 6000 Fax: +359 (0)2 / 97 44 06 1	<b>BULGARIA</b>	KONING & HARTMAN B.V. Haarlerbergweg 21-23 <b>NL-1101 CH Amsterdam</b> Phone: +31 (0)20 / 587 76 00 Fax: +31 (0)20 / 587 76 05	<b>NETHERLANDS</b>	CEG LIBAN Cebaco Center/Block A Autostrade DORA <b>Lebanon-Beirut</b> Phone: +961 (0)1 / 240 445 Fax: +961 (0)1 / 240 193	<b>LEBANON</b>
Mitsubishi Electric Europe B.V. Russian Branch 52, bld. 3 Kosmodamianskaya nab 8 floor <b>RU-115054 Moscow</b> Phone: +7 495 / 721 2070 Fax: +7 495 / 721 2071	<b>RUSSIA</b>	INEA CR Losinjska 4 a <b>HR-10000 Zagreb</b> Phone: +385 (0)1 / 36 940 - 01/ -02/ -03 Fax: +385 (0)1 / 36 940 - 03	<b>CROATIA</b>	Beijer Electronics AS Postboks 487 <b>NO-3002 Drammen</b> Phone: +47 (0)32 / 24 30 00 Fax: +47 (0)32 / 84 85 77	<b>NORWAY</b>	<b>AFRICAN REPRESENTATIVE</b>	
Mitsubishi Electric Europe B.V. Spanish Branch Carretera de Rubi 76-80 Apdo. 420 <b>E-08190 Sant Cugat del Vallés (Barcelona)</b> Phone: +34 (0) 93 / 5653131 Fax: +34 (0) 93 / 5891579	<b>SPAIN</b>	AutoCont C. S. S.R.O. Kačkova 1853/3 <b>CZ-702 00 Ostrava 2</b> Phone: +420 595 691 150 Fax: +420 595 691 199	<b>CZECH REPUBLIC</b>	Fonseca S.A. R. João Francisco do Casal 87/89 <b>PT-3801-997 Aveiro, Esqueiroa</b> Phone: +351 (0)234 / 303 900 Fax: +351 (0)234 / 303 910	<b>PORTUGAL</b>	ADROIT TECHNOLOGIES 20 Waterford Office Park 189 Witkoppen Road <b>ZA-Fourways</b> Phone: + 27 (0)11 / 658 8100 Fax: + 27 (0)11 / 658 8101	<b>SOUTH AFRICA</b>
Mitsubishi Electric Europe B.V. Swedish Branch Fjellievägen 8 <b>SE-22736 Lund</b> Phone: +46 (0) 8 625 10 00 Fax: +46 (0) 46 39 70 18	<b>SWEDEN</b>	HANS FÖLPGAARD A/S Theilgaardsgade 1 <b>DK-4600 Køge</b> Phone: +45 4320 8600 Fax: +45 4396 8855	<b>DENMARK</b>	SIRIUS TRADING & SERVICES SRL Aleea Lacul Morii Nr. 3 <b>RO-060841 Bucuresti, Sector 6</b> Phone: +40 (0)21 / 430 40 06 Fax: +40 (0)21 / 430 40 02	<b>ROMANIA</b>		
Mitsubishi Electric Europe B.V. UK Branch Travellers Lane <b>UK-Hatfield, Herts. AL10 8XB</b> Phone: +44 (0)1707 / 28 87 80 Fax: +44 (0)1707 / 27 86 95	<b>UK</b>	BEIJER ELECTRONICS ESTI OÜ Pärnu mnt.160i <b>EE-11317 Tallinn</b> Phone: +372 (0)6 / 51 81 40 Fax: +372 (0)6 / 51 81 49	<b>ESTONIA</b>	INEA SR d.o.o. Ul. Karadjordjeva 12/217 <b>SER-11300 Smederevo</b> Phone: +381 (0)64 / 68 55 187	<b>SERBIA</b>		
Mitsubishi Electric Europe B.V. Dubai Branch Dubai Silicon Oasis <b>United Arab Emirates - Dubai</b> Phone: +971 4 3724716 Fax: +971 4 3724721	<b>UAE</b>	Beijer Electronics OY Vanha Nurmijärventie 62 <b>FIN-01670 Vantaa</b> Phone: +358 (0)207 / 463 500 Fax: +358 (0)207 / 463 501	<b>FINLAND</b>	SIMAP SK (Západné Slovensko) Jána Derku 1671 <b>SK-911 01 Trenčín</b> Phone: +421 (0)32 743 04 72 Fax: +421 (0)32 743 75 20	<b>SLOVAKIA</b>		
Mitsubishi Electric Corporation Tokyo Building 2-7-3 Marunouchi, Chiyoda-ku <b>Tokyo 100-8310</b> Phone: +81 (3) 3218-2111 Fax: +81 (3) 3218-2185	<b>JAPAN</b>	HANS FÖLPGAARD A/S Theilgaardsgade 1 <b>DK-4600 Køge</b> Phone: +45 4320 8600 Fax: +45 4396 8855	<b>DENMARK</b>	INEA RBT d.o.o. Stegne 11 <b>SI-1000 Ljubljana</b> Phone: +386 (0)1 / 513 8116 Fax: +386 (0)1 / 513 8170	<b>SLOVENIA</b>		
Mitsubishi Electric Automation, Inc. 500 Corporate Woods Parkway <b>Vernon Hills, IL 60061</b> Phone: +1 (847) 478-2100 Fax: +1 (847) 478-0328	<b>USA</b>	PROVENDOR OY Teljänkatu 8 A3 <b>FIN-28130 Pori</b> Phone: +358 (0) 2 / 522 3300 Fax: +358 (0) 2 / 522 3322	<b>FINLAND</b>	Beijer Electronics Automation AB Box 426 <b>SE-20124 Malmö</b> Phone: +46 (0)40 / 35 86 00 Fax: +46 (0)40 / 93 23 01	<b>SWEDEN</b>		
		UTECO A.B.E.E. 5, Mavrogenou Str. <b>GR-18542 Piraeus</b> Phone: +30 (0)211 / 1206-900 Fax: +30 (0)211 / 1206-999	<b>GREECE</b>	OMNI RAY AG Im Schörl 5 <b>CH-8600 Dübendorf</b> Phone: +41 (0)44 / 802 28 80 Fax: +41 (0)44 / 802 28 28	<b>SWITZERLAND</b>		
		MELTRADE Kft. Fertő utca 14. <b>HU-1107 Budapest</b> Phone: +36 (0)1 / 431-9726 Fax: +36 (0)1 / 431-9727	<b>HUNGARY</b>	OOO "CSC-AUTOMATION" 4-B, M. Raskovoy St. <b>UA-02660 Kiev</b> Phone: +380 (0)44 / 494 33 44 Fax: +380 (0)44 / 494-33-66	<b>UKRAINE</b>		