

***MX OPC Server 5.0
Help Documentation***

Contents

<i>1. Introduction to MX OPC Server</i>	<i>1-1</i>
<i>2. Starting MX OPC Server Configuration</i>	<i>2-1</i>
<i>3. Address Space</i>	<i>3-1</i>
<i>4. Alarm Definitions</i>	<i>4-1</i>
<i>5. Simulation Signals</i>	<i>5-1</i>
<i>6. Runtime Operations</i>	<i>6-1</i>
<i>7. Conversion Definitions</i>	<i>7-1</i>
<i>8. Poll Method Definitions</i>	<i>8-1</i>
APPENDIX A. Using an OPC Server from Visual Basic	A-1
APPENDIX B. ACT Error Codes	B-1
APPENDIX C. DCOM Configuration	C-1

Introduction to MX OPC Server

1

1.1 Introduction to MX OPC Server

The MX OPC Server 5.0 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.

1.1.1 Compatibility with Former Versions

The MX OPC Server 5.00 is fully backward compatible with previous versions (4.20, etc.). The configuration databases from all previous versions can be used. The MX OPC Server 5.00 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.

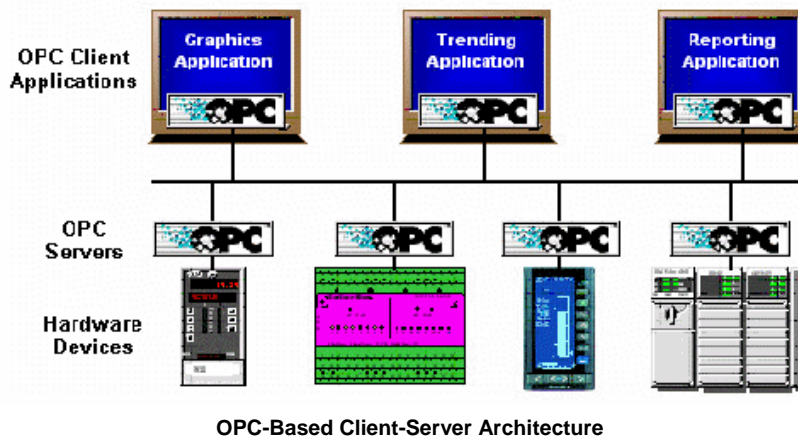
The MX OPC Server 5.00 configuration database cannot be used in the previous versions.

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 5.00 version will not import a CSV file created before version 5.00 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 5.00 configurator and then export to CSV.

1.1.2 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.



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).

1.1.3 MX OPC Runtime

The MX OPC Runtime maintains the driver's communication channels, devices, data blocks, 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).
- Automatically creates data blocks.
- Provides local configuration and control.
- Supports telephone modem communication.
- Provides phasing.
- Enables or disables Individual devices, data blocks and tags.
- Provides QuickFail Logic.
- Provides latched data.
- Provides a time/date stamp for data and alarms.
- Supports array element and individual bit reads
- Provides advanced diagnostics.
- MX Component

1.1.4 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, data blocks 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 and data blocks.
- A **Configuration View** for displaying and modifying device, data block 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.
- The feature to cut, copy and paste data blocks for fast configuration.
- Support for telephone modem communication.
- Templates for configuring default alarm settings.
- Simulation for tags and alarms using a built-in function pattern.

1.2 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 an NT service. NT 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.

1.2.1 System Requirements

To use MX OPC Server 5.0, you must have the following *minimum* system requirements:

- 233 MHz processor
- 64 MB RAM
- 100 MB disk space
- 8X speed CD-ROM drive
- VGA video card (256 or more colors for best results)
- Microsoft® Windows® 2000 with Service Pack 3, Windows XP with Service Pack 2, Windows NT® 4.0 with Service Pack 6a, or Windows Embedded NT and XP.

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

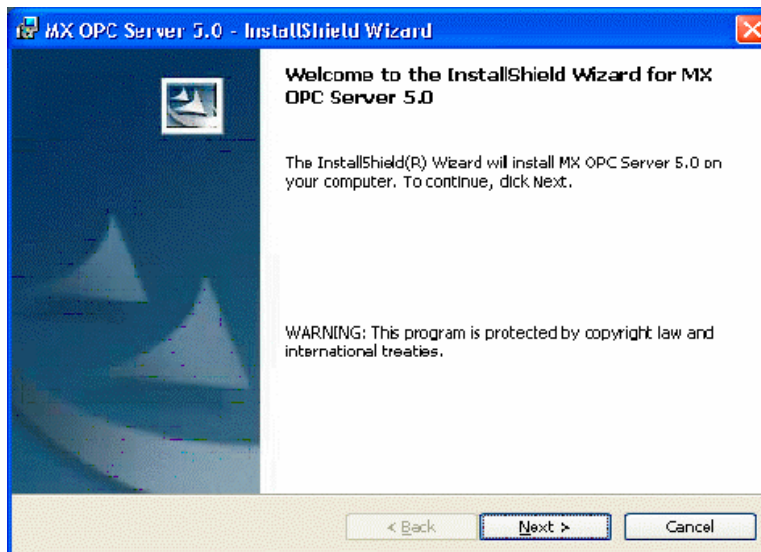
Note: Only versions of Microsoft Windows that accommodate **Unicode** character sets are supported.

1.2.2 Installation Procedure

The following steps detail the MX OPC Server installation:

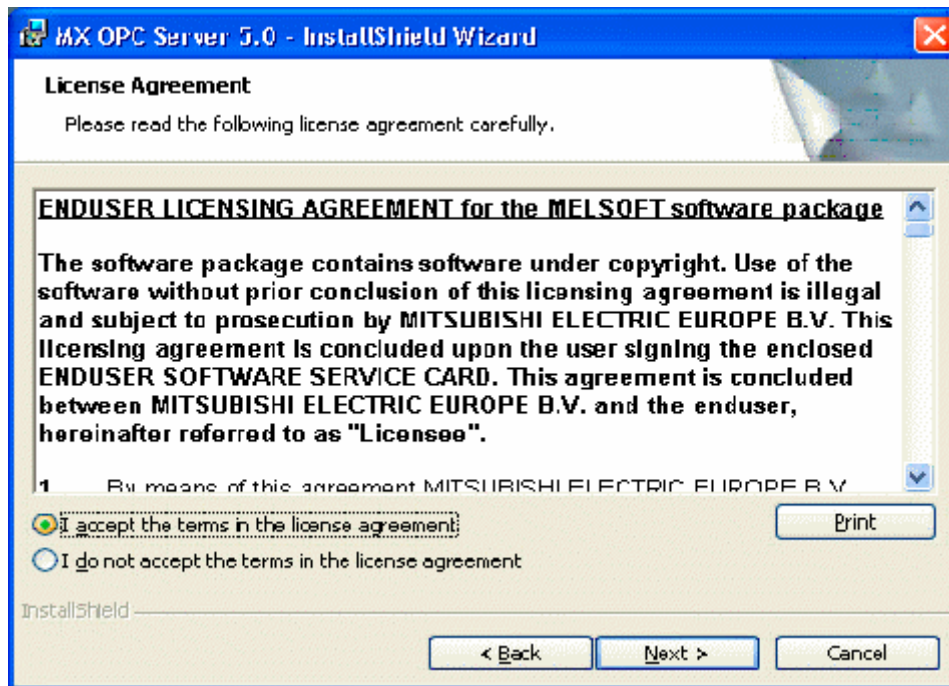
Note: If your operating system (e.g., Windows NT) requires a login name, you must log in with administrator capability before installing MX OPC Server.

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. The Setup screen briefly appears, followed by the Welcome screen, as shown in the figure below. Click the Next button to continue.



Welcome Screen

4. 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.



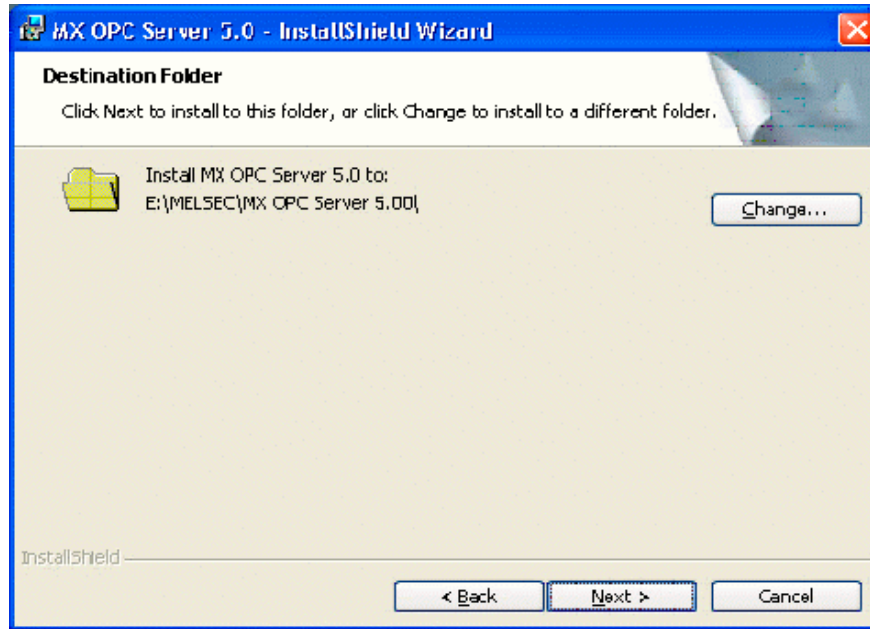
Software License Agreement

5. 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.



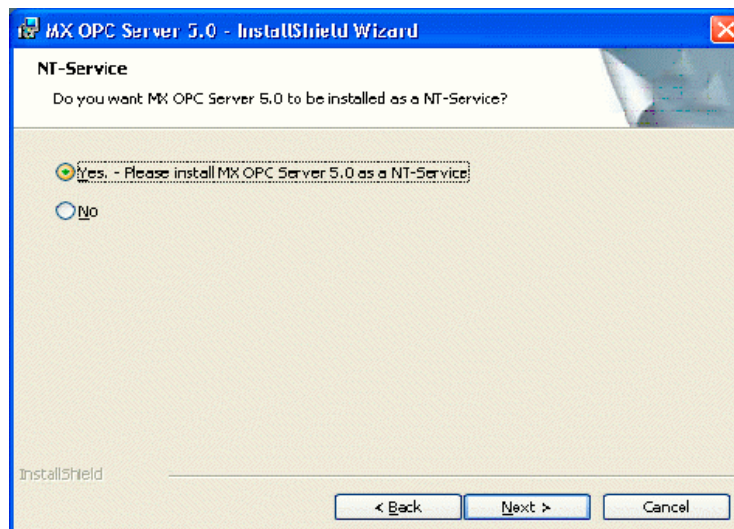
Entering Contact Information

- 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.



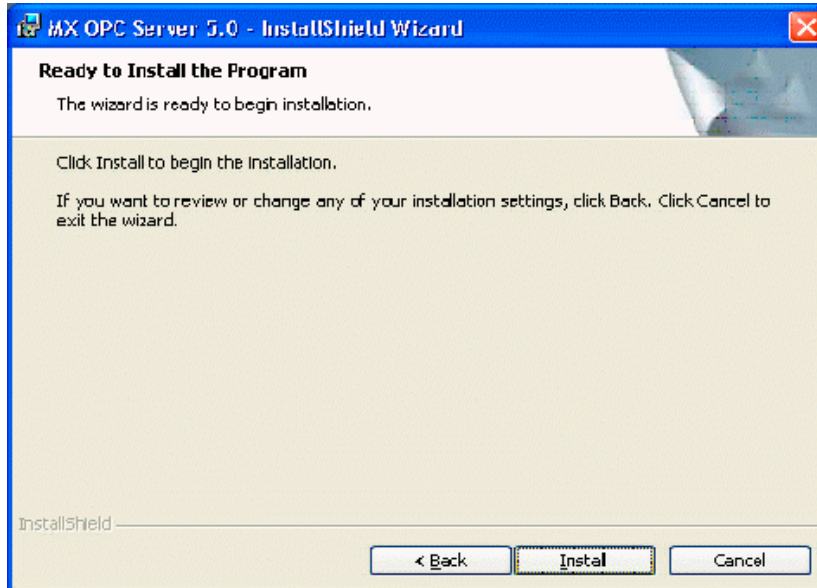
Choosing the Destination Location

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



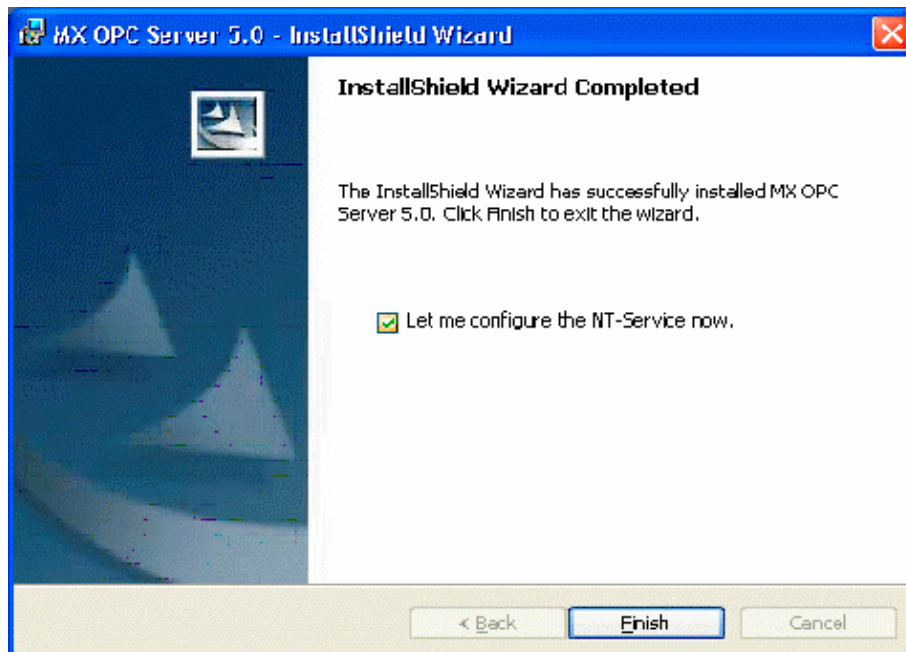
Option to Run MX OPC Server as an NT Service

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



Starting File Installation

- After all files and components have been installed, click the **Finish** button to complete the installation, as shown in the figure below. You may be prompted to restart your computer before using the product. Choose whether to configure the NT Service now or later, and then click the **Finish** button.



Installation Complete

1.3 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

1.3.1 Starting the MX OPC Configurator

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

1. From the Windows **Start** menu, select **Programs > Melssoft Application > MX OPC Server 5.0 > 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, data blocks, 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 blocks:** A data block is an addressable portion of a device. The data block contains the properties and methods that manage the behavior of a data block. Data blocks in the server's local memory correspond to data areas in the MX Runtime. When you add new data blocks to the server's local memory, you also add new data areas to the MX Runtime. Data blocks are not visible to the OPC client.

Note: Data block configuration is optional (for advanced users).

- **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).

1.3.2 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
 - Configuring data blocks
 - Setting default values for driver, channels, devices, data blocks and tags
 - Configuring tags

- 5 Check the I/O Driver Status:**
 - Using monitoring mode

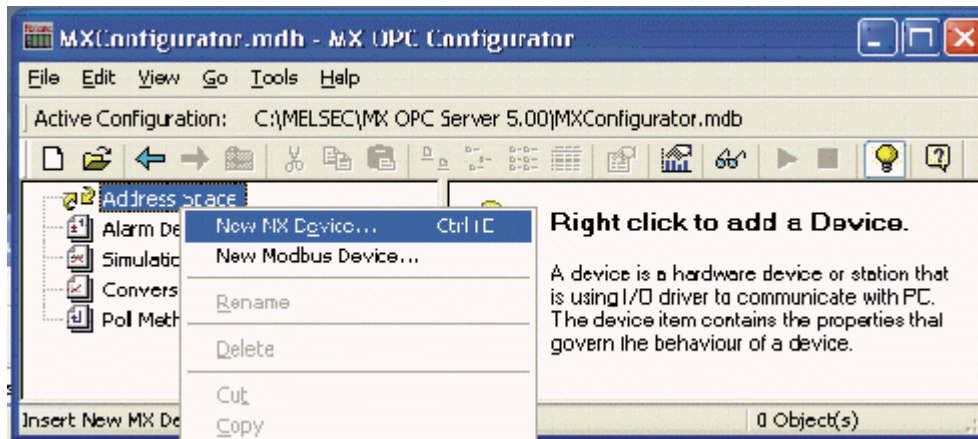
1.3.3 Creating Communication Channels

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

Serial Communication Channel

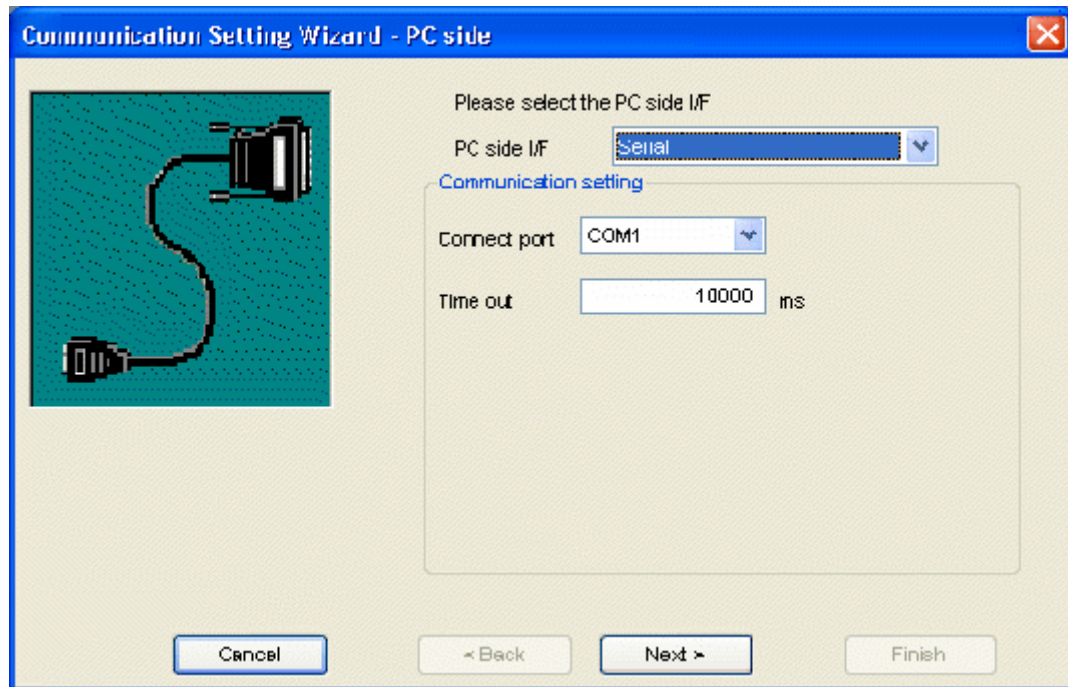
To create a new Serial communication path:

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.



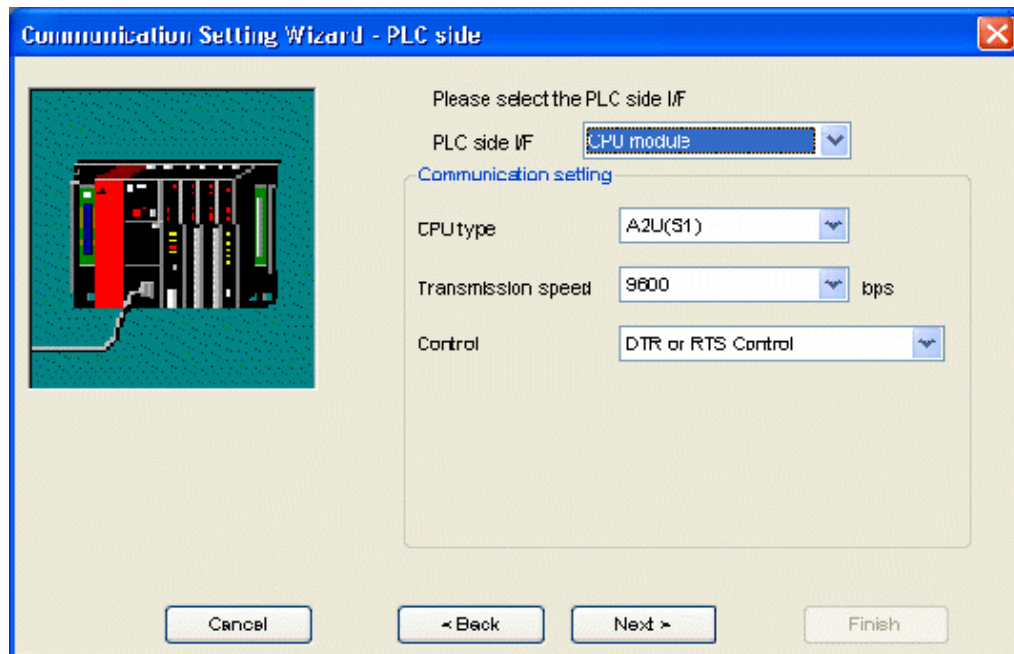
Adding a New Device

2. This opens 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. Click **Next** to continue.



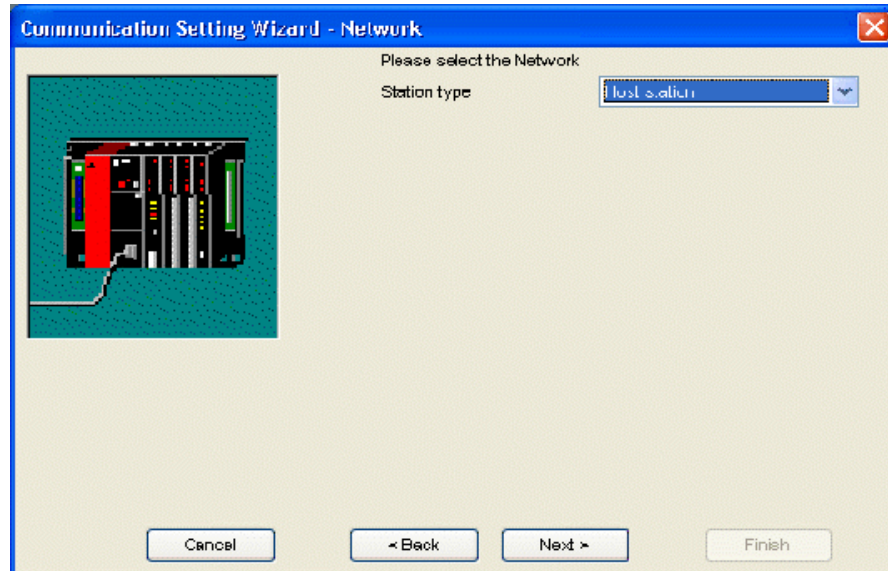
Setting up the Connection for the PC Side

- 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 a communication type from the drop-down list as the connection type to use for the selected channel. Click **Next** to continue.



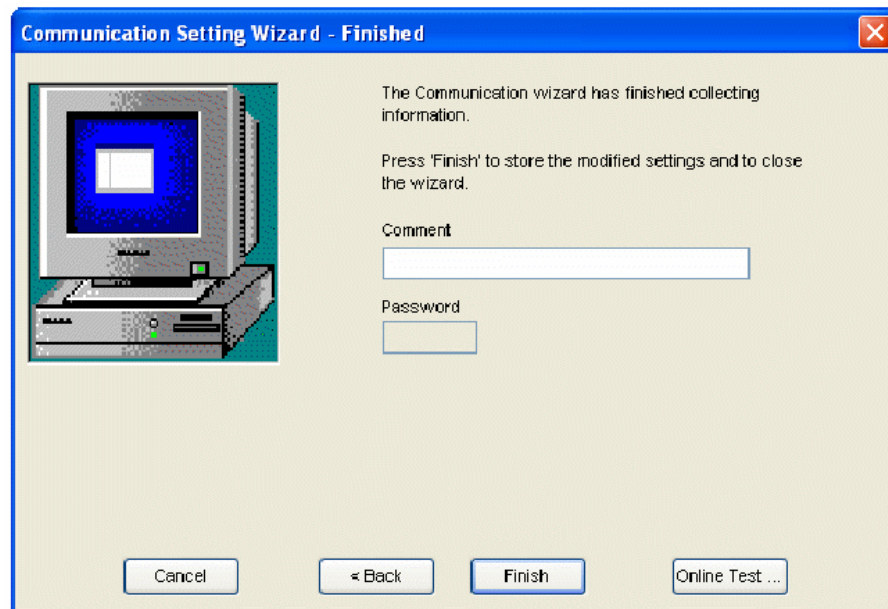
Setting up the Connection for the PLC Side

4. 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.



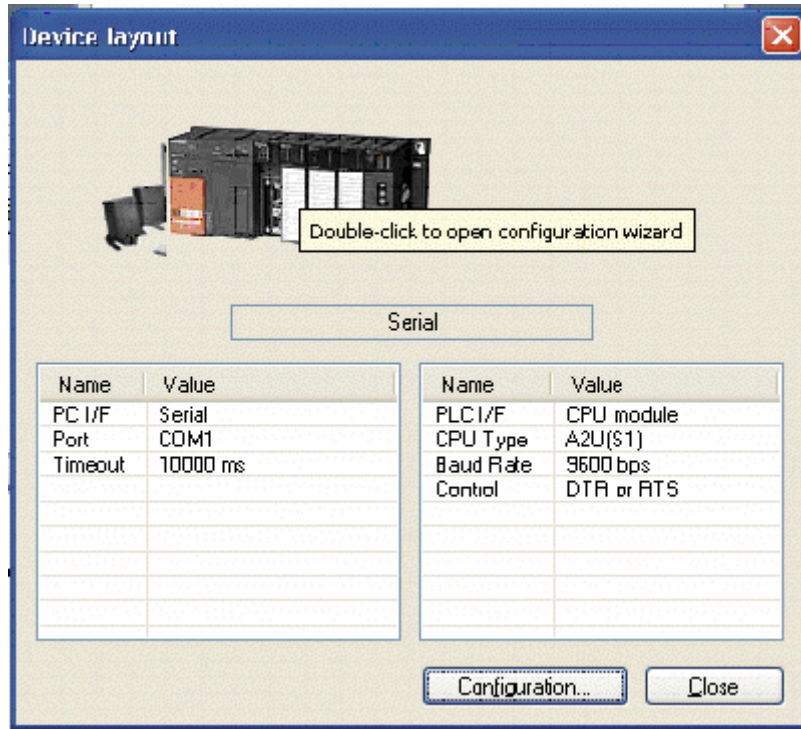
Configuring Network Settings

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



Completing Channel Setup

- The **Device Layout** dialog box appears, as shown in the figure below. The left-hand side of the dialog lists the configuration properties for the PC side. You can double-click the image of the PC (or double-click the configuration properties list) to edit the PC side 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 **Configuration** button to return to the Communication Setting Wizard. Click the **Close** button to configure the device properties.



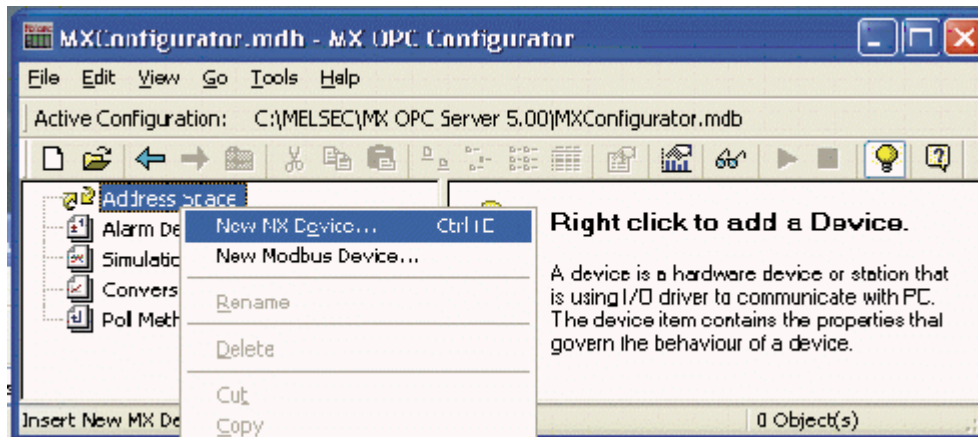
Device Layout

Note: For complete information about channel configuration properties and the Communication Setting Wizard, see **Chapter 3**.

Ethernet Communication Channel

To create a new Ethernet communication path:

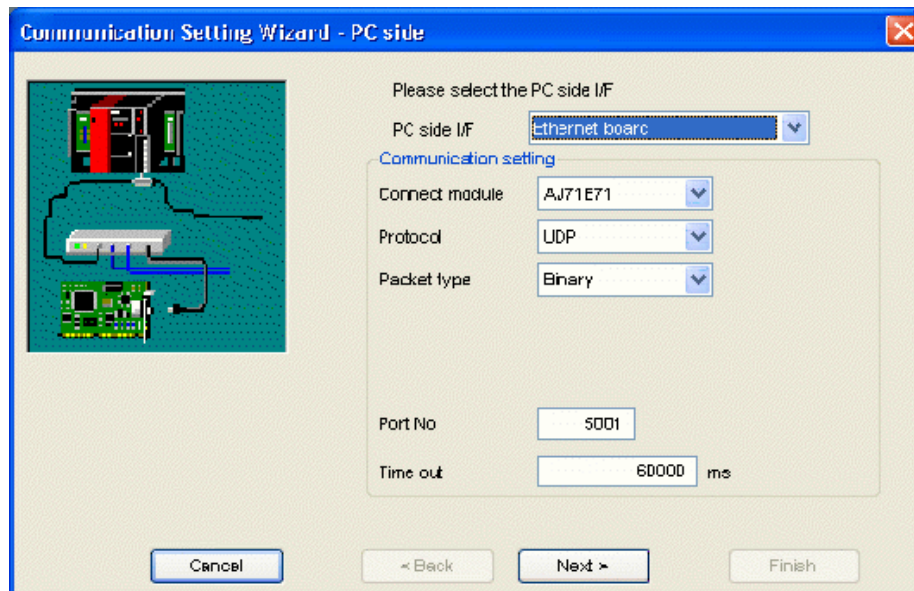
- Right-click the **Address Space** tree control of the Configurator screen and select **New MX Device** from the pop-up menu, as shown in the figure below.



Adding a New Device

2. This opens 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.

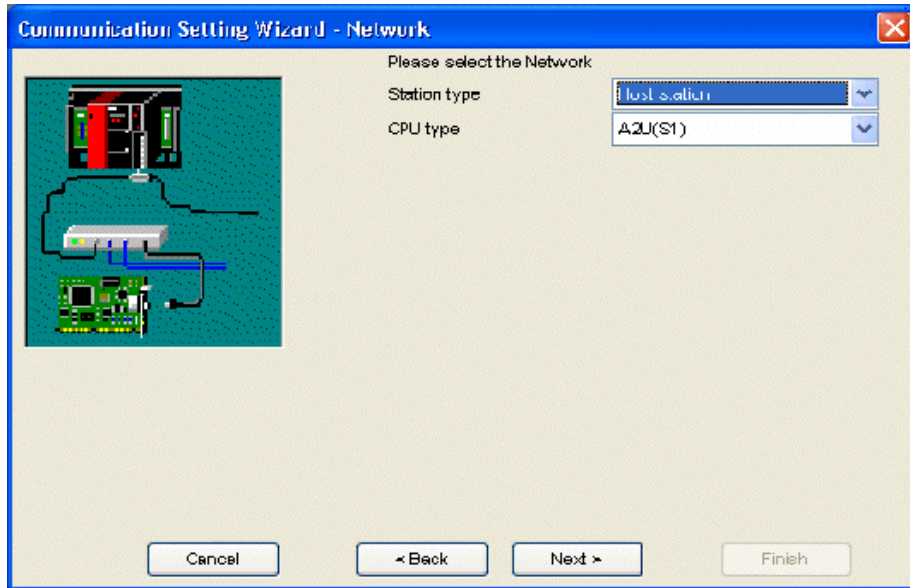
Note: For a list of supported devices, please see **Chapter 3**.



Setting up the Connection for the PC Side

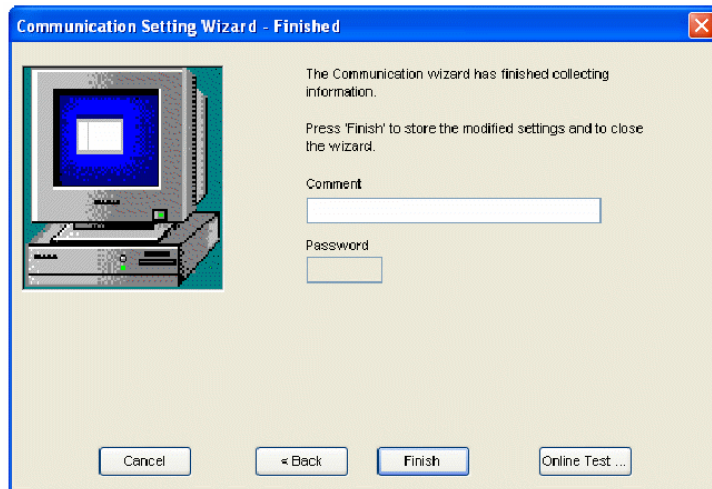
3. Now you must specify the communication type to use on the PLC (device) side, as shown in the figure below. In the **PLC Side I/F** field, select **Ethernet module** from the drop-down list as the connection type to use for the selected channel. Specify a **Host name** and a **Port number**. Click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

4. 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.



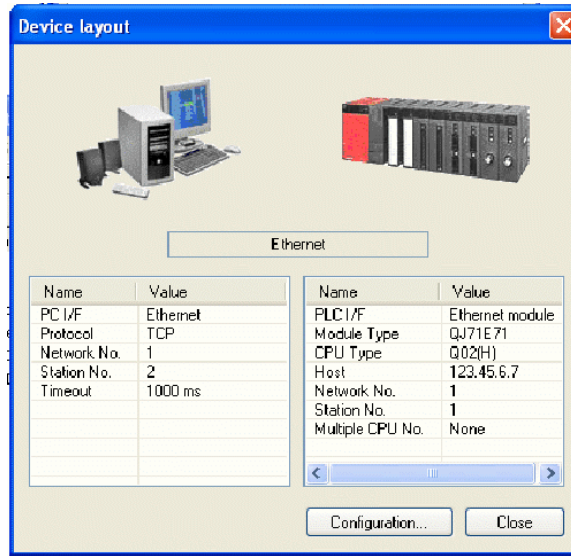
Configuring Network Settings

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



Completing Channel Setup

6. The **Device Layout** dialog box appears, as shown in the figure below. The left-hand side of the dialog lists the configuration properties for the PC side. You can double-click the image of the PC (or double-click the configuration properties list) to edit the PC side 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 **Configuration** button to return to the Communication Setting Wizard. Click the **Close** button to configure the device properties.



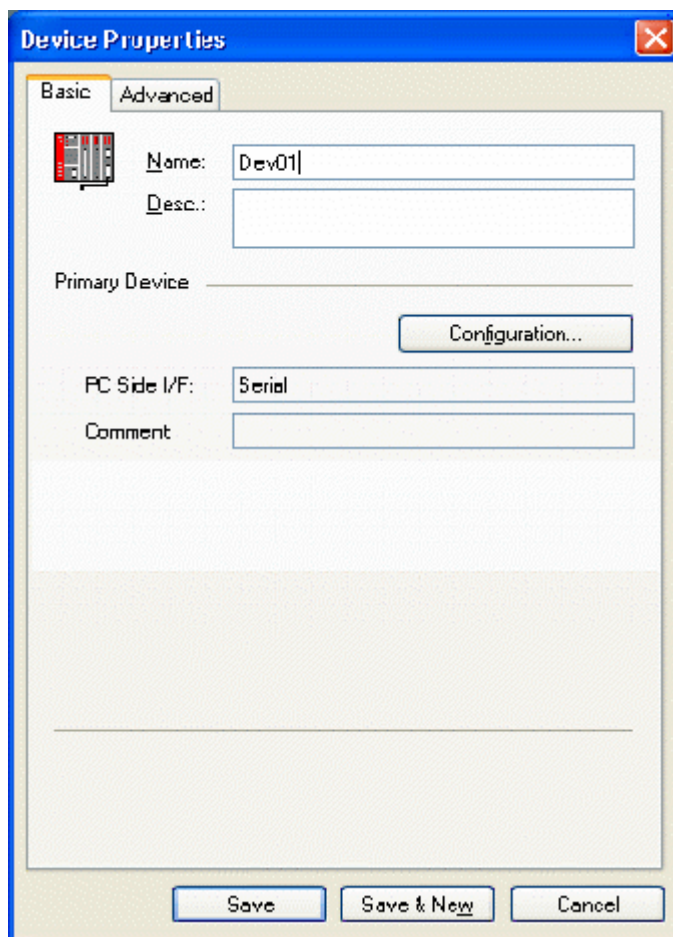
Device Layout

Note: For complete information about channel configuration properties and the Communication Setting Wizard, see **Chapter 3**.

1.3.4 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:

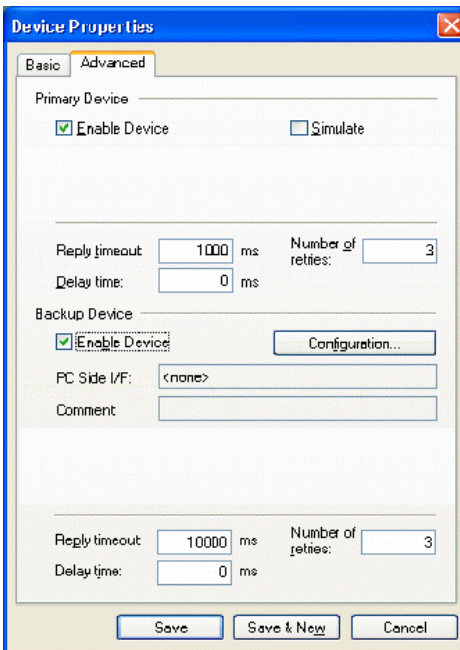
1. The **Basic** tab of the **Device Properties** dialog box appears, as shown in the figure below. The communication type (e.g. Serial, Ethernet, etc.) is shown in the **PC Side I/F** field, as shown in the figure below.



Configuring Device Properties

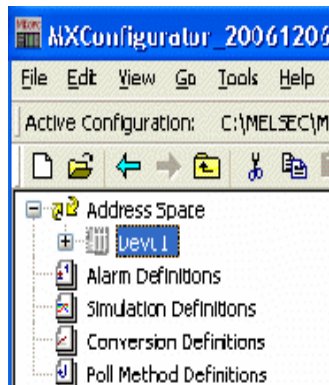
2. In the **Name** field, type a name for the new device, and type a description for the device (optional). To specify which connection type and which physical connection (COM) port to use for the selected device, click the **Configuration** button to launch the Communication Setting Wizard for the Primary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

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



Device Properties: Advanced Tab

- Check the **Enable Device** check box to activate the device. **Note:** For complete information about device configuration properties, see **Chapter 3**.
- 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 Chapter 3.

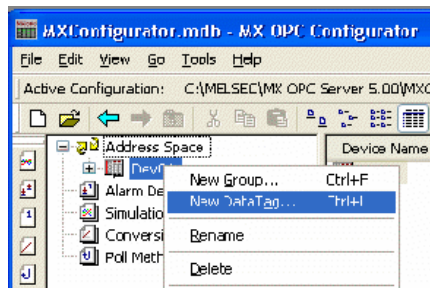
1.3.5 Creating Data Tags

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 **Chapter 3** for details.

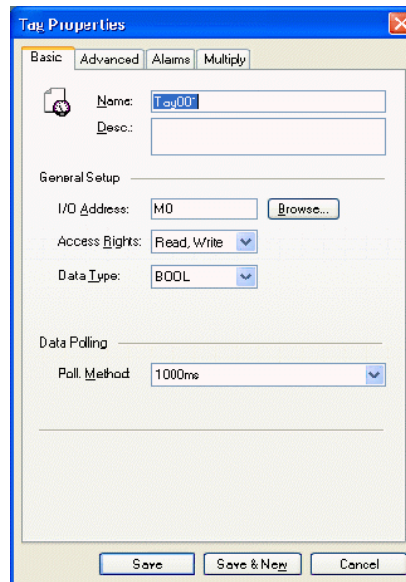
To create a new tag:

1. Right-click on the **Dev01** 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 Data Tag

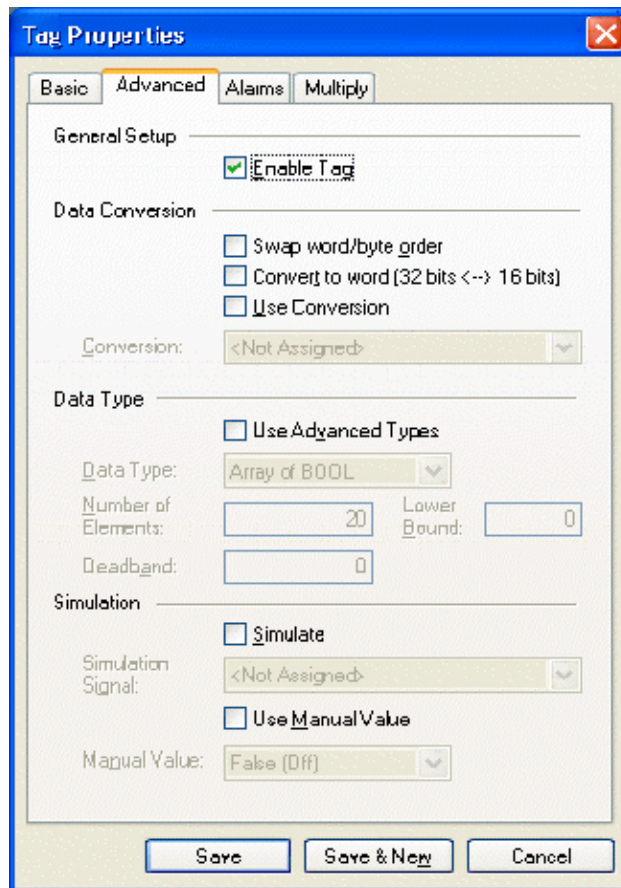
2. The **Basic** tab of the **Data Tag Properties** dialog box appears, as shown in the figure below.



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**.

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



Data Tag Properties: Advanced Tab

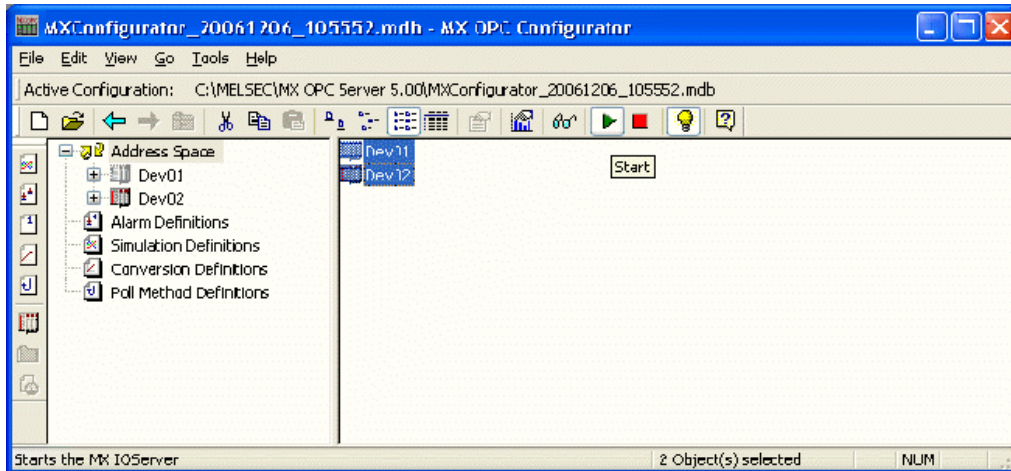
- Set the **Data Conversion** and **Data Type** parameters for the tag.
- In the **Alarms** tab of the **Tag Properties** dialog box, you can check the **Generate Alarms** check box to make the server generate a limit alarm and/or a digital alarm based on the data item value.
- When you have finished 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 **Chapter 3**.

Note: By default, the Configurator will automatically arrange your data tags into blocks so that they can be read from the device in the most efficient way. If you would prefer to define the data blocks manually, you can do this by selecting **Options** from the **Tools** menu and checking the **Advanced Mode (Show Data Blocks)** check box on the **General** tab of the **Options** dialog box.

1.3.6 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

1.3.7 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.

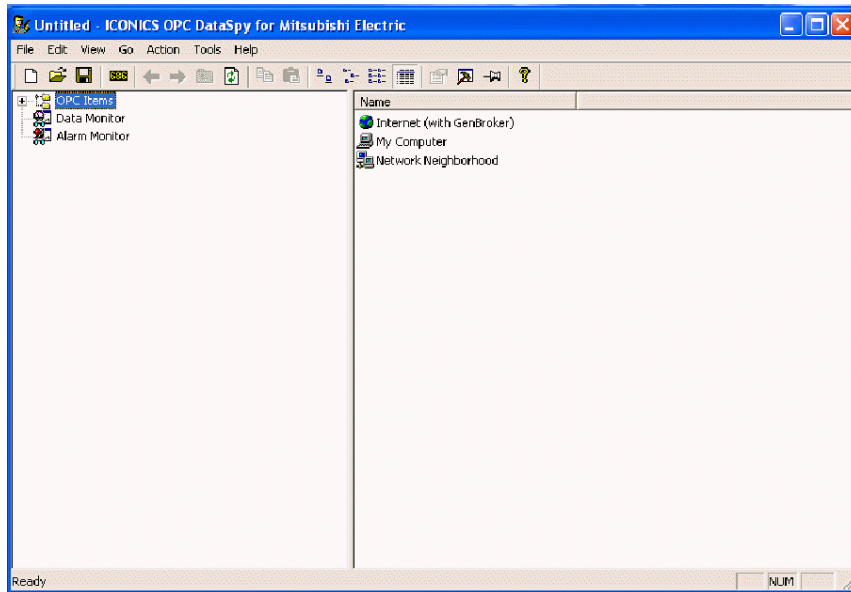
The OPC DataSpy user interface consists of the following elements:

- 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 **Programs > Melseft Application > MX OPC Server 5.00 > 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.5"** section.

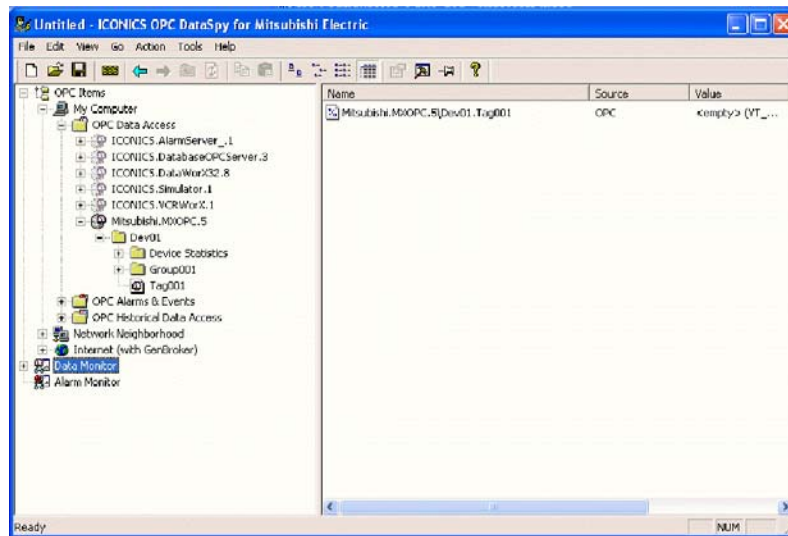
Users will then see all configured devices listed underneath **"Mitsubishi.MXOPC.5"** and can browse groups and data tags accordingly.

3. Right click on any selected tag listed beneath **"Mitsubishi.MXOPC.5"** 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 single array elements or reading a single bit within a word, you may still be able to use these features by changing the name of the item that you add.

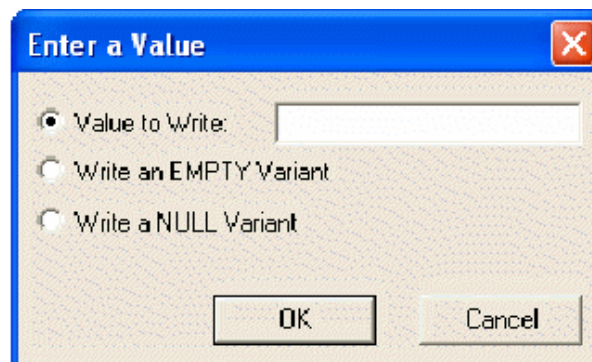
A client can address a bit within a word tag using the item syntax "tagname.bitNr" (e.g. tagname.0 for bit 0).

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 (,).



Starting MX OPC Server Configuration **2**

2.1 Starting the Configurator

To start the MX OPC Configurator:

1. From the Windows **Start** menu, select **Programs > Melsoft Application > MX OPC Server 5.00 > 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.

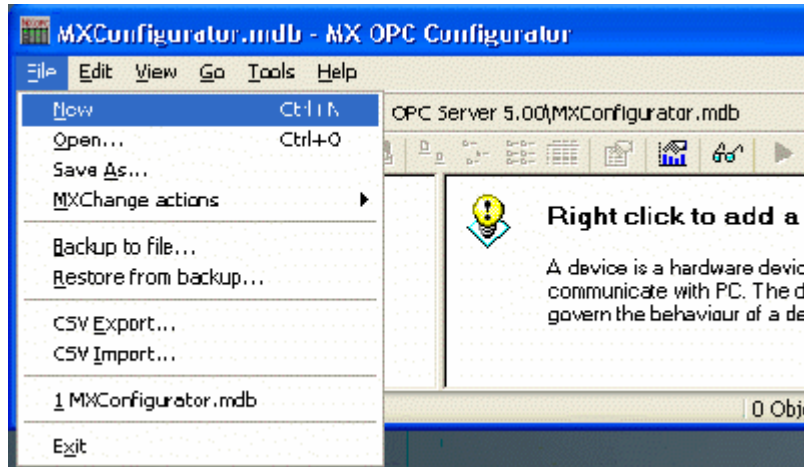


Configurator Screen

2.2 Creating Configuration Databases

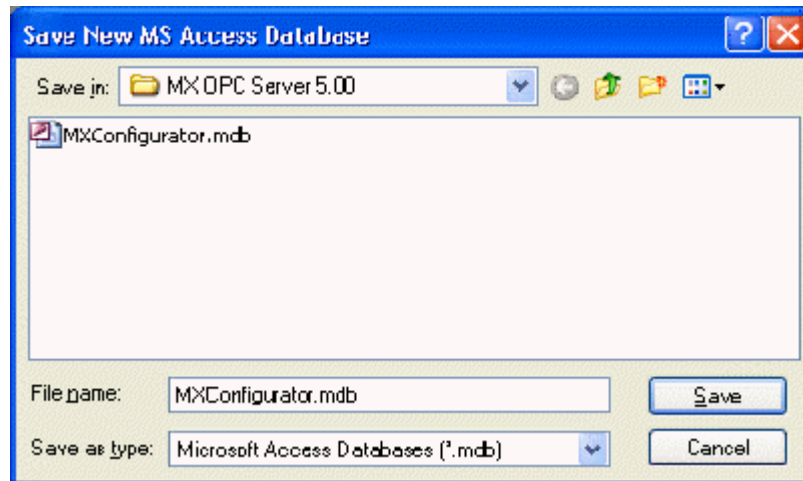
The Configurator uses Microsoft Access configuration databases. To create a new configuration database in the Configurator:

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



Creating a Configuration Database

1. 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.

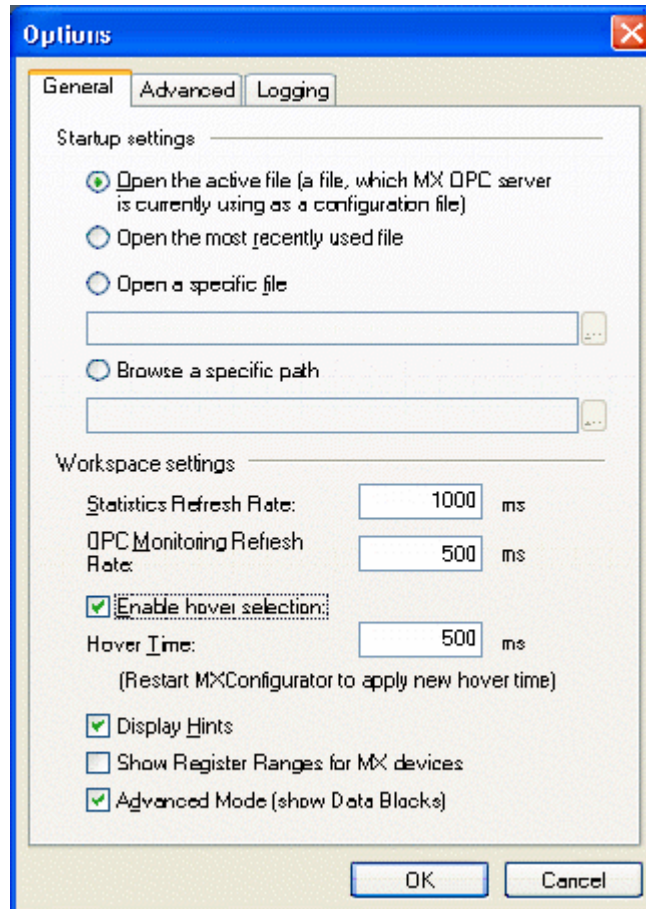


Saving the New Configuration Database

2.2.1 Configuration Modes

You can use the configurator in either basic mode or advanced mode. By default, the configurator runs in **basic configuration mode**. In basic mode, data blocks are hidden in the **Address Space** configuration. To switch to **advanced configuration mode**, go to the **Tools** menu and select **Options**. Go to the **General** tab and check the **Advanced Mode (show Data Blocks)** check box, as shown in the figure below. When this check box is checked, data blocks will be visible and editable throughout the configuration.

Note: Data block configuration is optional (for advanced users).



Options Dialog Box: General Tab

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
- Edit
- View
- Go
- Tools
- Help

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.

2.4.1.1 New

New - Creates a new configuration database.

2.4.1.2 Open

Open - Opens an existing configuration database Microsoft Access (.mdb) file.

2.4.1.3 Save As

Save As - Saves the current database under a different name as a Microsoft Access (.mdb) or Microsoft Data Link (.udl) file.

2.4.1.4 MXChange Actions

MXChange Actions - Establishes a connection with the MXChange Server, allowing you to login to or log out of the server and change your security password.

2.4.1.5 Backup to File

Backup to File - Saves the present configuration to a new or existing Microsoft Access Database (.mdb).

2.4.1.6 Restore from Backup

Restore from Backup - Loads prior configuration from Microsoft Access Database (.mdb).

2.4.1.7 CSV Export

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.

2.4.1.8 CSV Import

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.

2.4.1.9 Exit

Exit - Closes the application. The current configuration database is automatically saved.

2.4.2 MXChange Actions

The **MXChange Actions** command on the **File** menu allows you to establish a connection to the MXChange Server. NOTE: Modbus devices are not imported or exported within MXChange.

[Login](#)

[Logout](#)

[Change Password](#)

[Import](#)

[Export](#)

[Update](#)

2.4.2.1 Login

Logging Into the MXChange Server

To connect to the MXChange Server:

Select **MXChange Actions > Login** from the **File** menu.

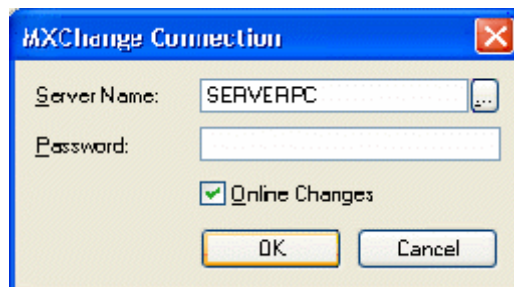
This opens the **MXChange Connection** dialog box, as shown in the figure below, which is used to connect to the MXChange Server. You can browse for the MXChange Server or type in the **Server Name**:

Server Name: Name of a computer you want to connect to using MXChange API. You can use click the ... button to browse the MXChange Servers on the Local Area Network (LAN). It will take a few seconds to get the information.

Password: Enter the administrator password to access the MXChange Server.

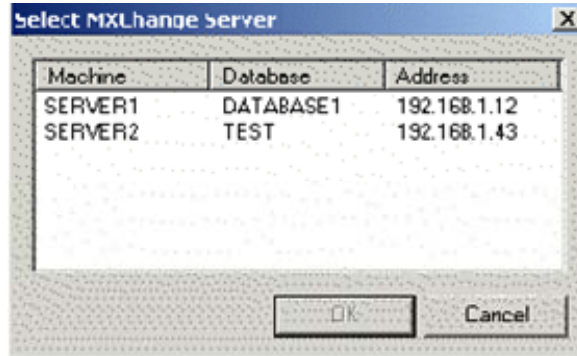
Note: The user is always the administrator (admin).

Online Changes: Allows you to see MXChange Server database changes as soon as they happen. If you check the check box then you will see notification dialogs when something is changed in the MXChange database on the MXChange Server.



Logging Into the MXChange Server

1. If you click the ... button, the **Select MXChange Server** dialog will appear after a few seconds, as shown below. This allows you to browse the network for a server node. All MXChange servers on the network will be displayed with the name of the active database and the IP address of the server. These items are ready to select and then connect with the previous dialog. Click **OK**.



Selecting an MXChange Server

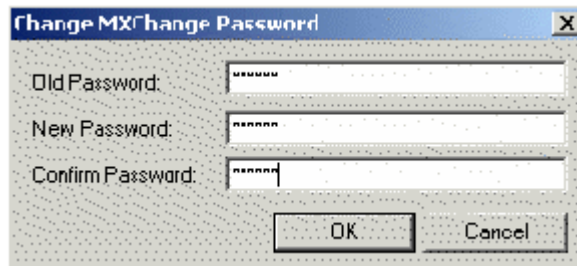
2. On the **MXChange Connection** dialog box, click **OK**. The configurator tries to connect to the MXChange Server and login as administrator with the provided password. If the login fails, you will be informed by an error dialog.

2.4.2.2 Change Password

Changing the Password for the MXChange Server

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 logged in 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

2.4.2.3 Logout

Logging out of the MXChange Server

To disconnect from the MXChange Server, select **MXChange Actions > Logout** from the **File** menu. (**Note:** This menu command is available only if you are currently logged in to the MXChange Server.)

2.4.2.4 Export

Exporting Configuration Data to the MXChange Server

You can export configuration data from your configuration database to the MXChange Server. If you are not connected to the MXChange Server before you use the export command, then you will be asked to log in to the server. You can export data only if you have a device created. The export function exports all devices as separate PLC projects in the MXChange Server. To export data to the server:

1. Select **MXChange Actions > Export** from the **File** menu.
2. The export function will either reuse an existing OPC project node or create a new OPC project node in the MXChange database (if one does not already exist). The name of the OPC project node is derived from the name of the MX OPC Configurator's currently edited database.

Note: OPC project nodes and device nodes that already exist in the MXChange database will be reused.

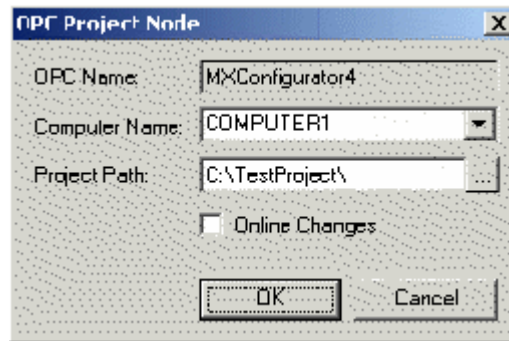
3. The **OPC Project Node** dialog box appears, as shown in the figure below. (Note: This dialog will also appear if there is no OPC project node in the MXChange Server with the same name as the currently edited configurator database.) Specify the following properties for the node, and then click **OK**:

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 MXChange Server database changes using the MXChange callback function. If you check the check box then 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).



OPC Project Node Properties

- The **PLC Project Node** dialog box appears, as shown in the figure below. Specify the following properties for the node, and then click **OK**:
Note: If the project does not already exist, you will be prompted to create it, as shown in the figure below.

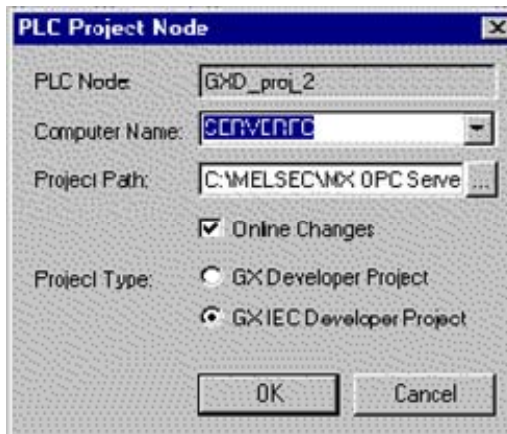
PLC Node: Name of the project node in the MXChange database.

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 MXChange Server database changes using the MXChange callback function. If you check the check box then 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 what type of PLC project will be used. There are two developer systems normally used for creating PLC Project nodes: **GX Developer** and **GX IEC Developer**.



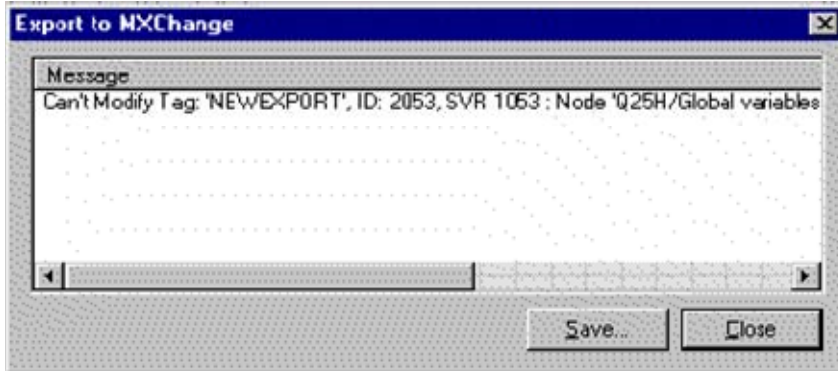
PLC Project Node Properties

- When you click **OK** on the previous dialog, then the export will begin. If everything goes well, the following message box will appear. The MX OPC Server tags will be automatically created in the MXChange Server database.



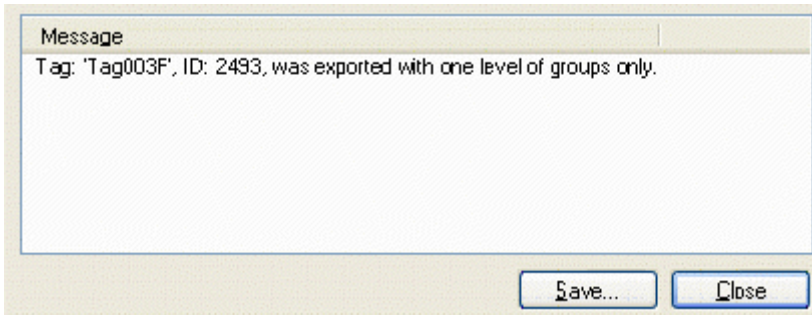
Export to MXChange Server Successful

6. 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

Note: In the MX OPC Configurator, data tags can be logically organized into **groups** (folders). When exporting configuration data from your configuration database to the MXChange Server, the MX OPC Configurator can also export tags that are included in groups. The group name is encoded into the description field during the tag export. The syntax of groups is "\$_GroupName_\$Description". The group name is delimited by special characters: "\$_" at the beginning of the group name, and "\$_" at the end of the group name. Only one level of groups is supported in the export. If a tag is nested under more than one level of groups, it is exported with the uppermost (top-level) group only, and a warning message is generated, as shown in the figure below.



MXChange Export Warning

Note: In the MX OPC Configurator, data tags can be logically organized into **groups** (folders). When importing configuration data into your configuration database from the MXChange Server, the MX OPC Configurator can also import tags that are included in groups. A tag configured in MXChange must have the group name defined in its description field using the following syntax: "\$_GroupName_\$Description". The group name is delimited by special characters: "\$_" at the beginning of the group name, and "_\$" at the end of the group name.

In the GX IEC Developer, the groups can be specified in the **Comment** field in the Global Variable List of tags. There are two simple rules for encoding the group name into the comment: 1) Group name is encapsulated by the \$_ and _\$ delimiter characters (e.g. \$_Group01_\$). 2) Description/comment cannot begin with \$_ characters and end with _\$ characters; otherwise it could be recognized as a group during the import from MXChange.

Examples

Let's suppose the configurator has the following MX OPC Server address space:
Device01 -

```

|- Tag001
|- Tag002
|- Group01 -
|   |- Tag003
|   |- Tag004
|   |- Group02-
|       |- Tag005
|
|- Group03 -
|       |- Tag006

```

All six tags will be exported into the MXChange database in the following way:

```

Tag001: comment ""      -> device level
Tag002: comment ""      -> device level
Tag003: comment "$_Group01_$" -> Group01 under device level
Tag004: comment "$_Group01_$" -> Group01 under device level
Tag005: comment "$_Group01_$" -> Group01 under device level -> warning
message will be displayed about the unsupported level of groups
Tag006: comment "$_Group03_$" -> Group03 under device level

```

In case the description is used in the tag definition in MX OPC Server, or you want to mix the group definition and tag comment in GID, the following encoding is used:

Group Name	Comment/Description	Encoded Comment in GX IEC Developer
-	"hello"	"hello"
-	"hello_\$"	"hello_\$"
-	"\$_hello"	"\$_hello"
Group01	""	"\$_Group01_\$"
Group01	"hello_\$"	"\$_Group01_\$hello_\$"
Group01	"\$_hello"	"\$_Group01_\$\$_hello"

Notification of Changed Tags in MXChange Server

Whenever a tag is modified in the MXChange Server, a notification will appear as shown in the figure below. Click the **OK** button.



Notification of Changed Tags in MXChange Server

2.4.2.6 Update

Updating the MXChange Server Database

Selecting **MXChange Actions > Update** from the **File** menu synchronizes the data between the MX OPC Configurator and the MXChange Server.

2.4.2.7 MXChange Select Node

If you click the ... button, the Select MXChange Server dialog will appear after a few seconds. This allows you to browse the network for a server node. All MXChange servers on the network will be displayed with the name of the active database and the IP address of the server. These items are ready to select and then connect with the previous dialog. Click OK.

2.4.2.8 MXChange OPC Project Node

The OPC Project Node dialog box will appear after logging in to MXChange and selecting Import/Export/Update if there is not already a node with the same name. 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 MXChange Server database changes using the MXChange callback function. If you check the check box then 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).

2.4.2.9 MXChange PLC Project Node

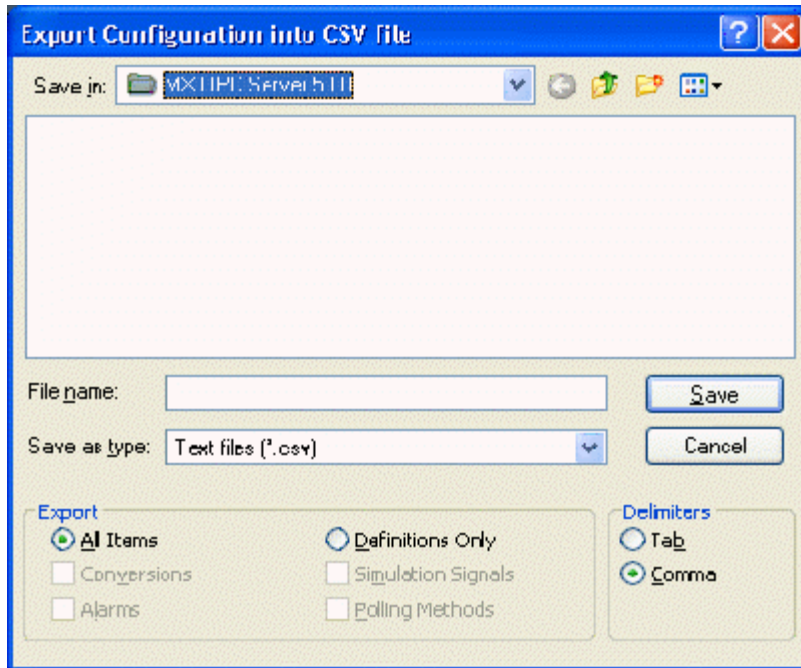
When the PLC Project Node dialog box appears, you will be able to specify the following properties for the node, and then click OK. Note: If the project does not already exist, you will be prompted to create it.

- **PLC Node:** Name of the project node in the MXChange database.
- **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 MXChange Server database changes using the MXChange callback function. If you check the check box then 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 what type of PLC object will be used. There are two developer systems normally used for creating PLC Project nodes: GX Developer and GX IEC Developer.

2.4.3 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

2.4.4 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:

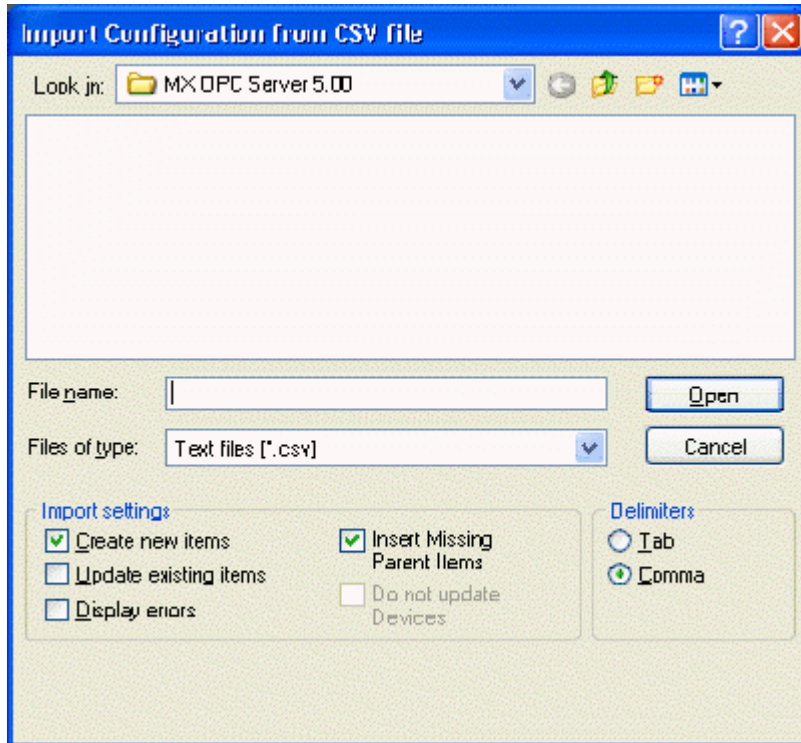
Create new items: When the import file contains items that are not yet in the configuration database, then it creates them. Otherwise it skips these items.

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.

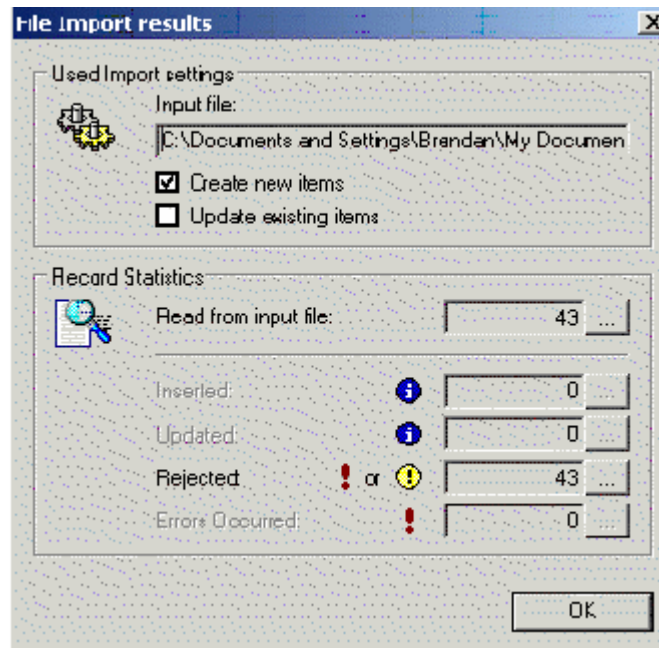
Display errors. When this item is checked, the Configurator shows a dialog box if an error occurs, and then asks you if you want to proceed with the import. When it is not checked, it skips all items where an error occurred.

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.



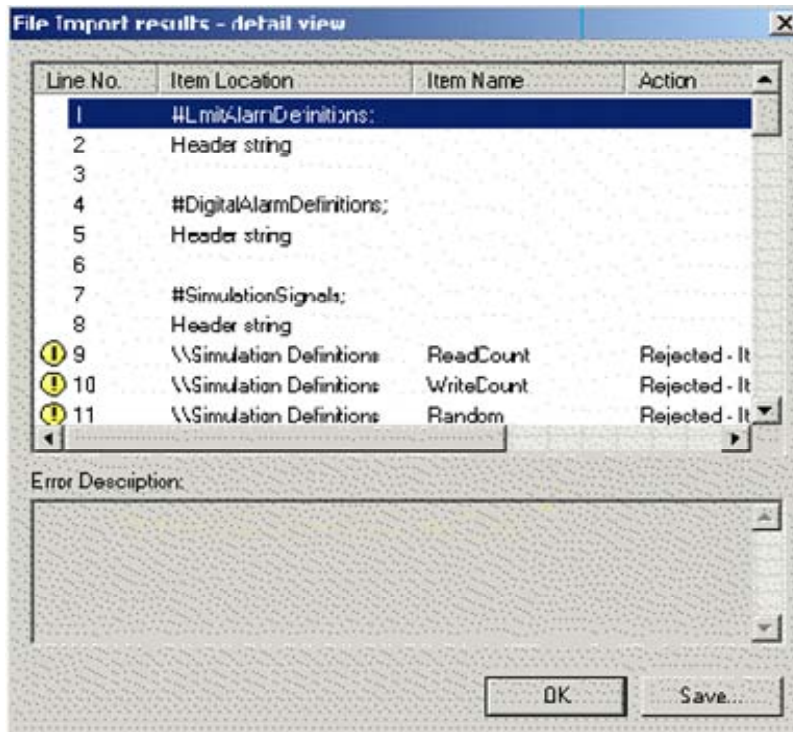
Importing Configuration Data

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.



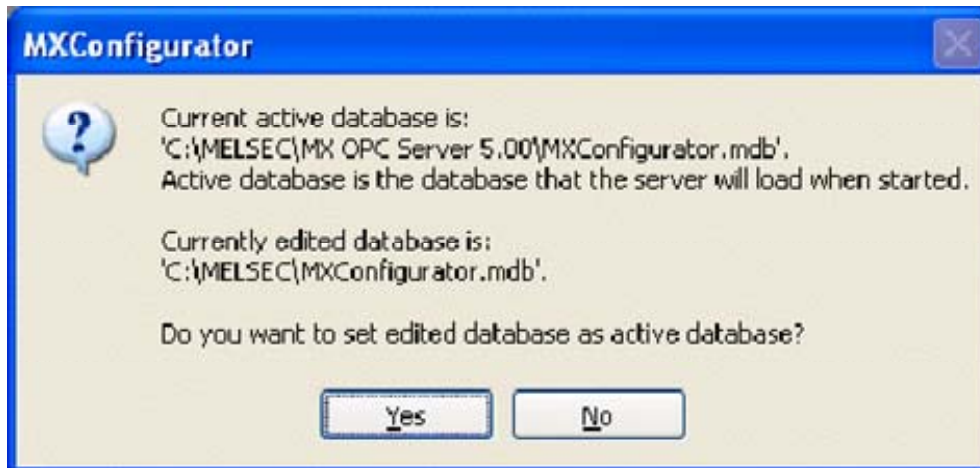
File Import Results Details

2.4.5 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

2.4.6 Edit Menu

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

Edit Menu Commands

Command	Shortcut Keys	Function
New MX Device	CTRL+E	Creates a new MX item.
New Modbus Device		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.

2.4.6.8 Rename

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 12 alphanumeric characters, including underscores (_) and hyphens (-).

2.4.7 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.

2.4.8 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 Address Space tree control.
Alarm Definitions	F3	Moves the cursor to the Alarm Definitions tree control.
Simulation Definitions	F4	Moves the cursor to the Simulation Definitions tree control.
Conversion Definitions	SHIFT+F3	Moves the cursor to the Conversion Definitions tree control.
Poll Method Definitions	SHIFT+F4	Moves the cursor to the Poll Method Definitions 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.

2.4.9 Tools Menu

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

Tools Menu Commands

Command	Function
Options	Launches the Options dialog box.
Compact/Repair MS Access Database	Opens the Compact/Repair MS Access Database dialog box.

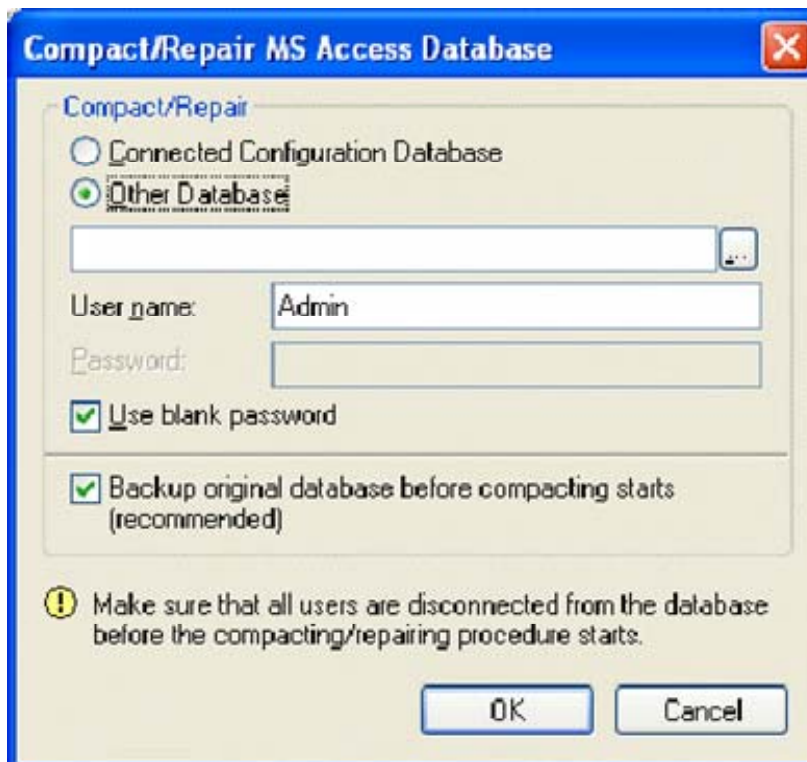
2.4.9.2 Compact/Repair MS Access Database

Compacting and Repairing Microsoft Access Databases

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

2.4.10 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.

2.5 Options

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

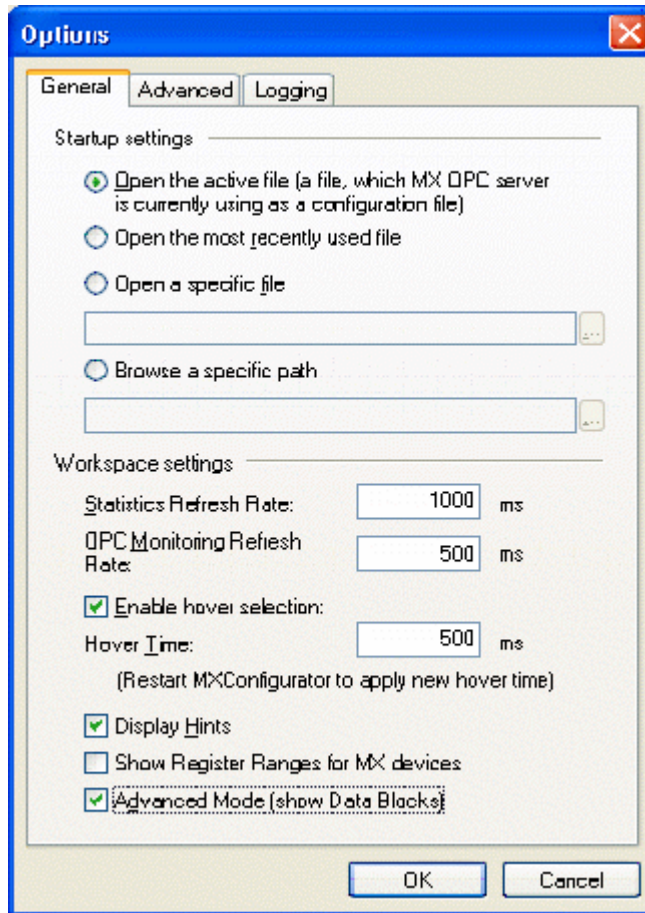
General

Advanced

Logging

2.5.1 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.

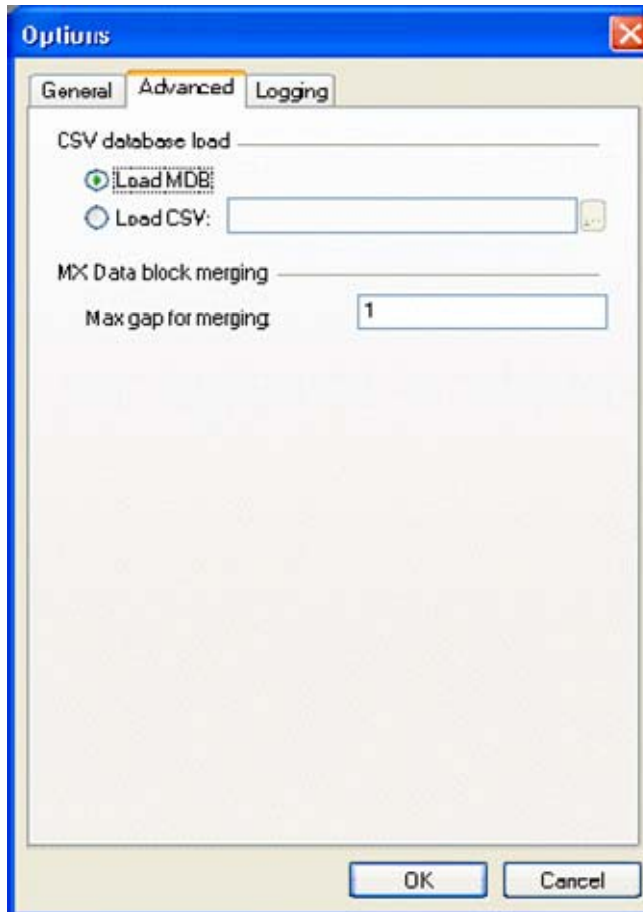
Advanced Mode (Show Data Blocks): By default, data blocks are hidden in the **Address Space** configuration. When this check box is checked, data blocks will be visible throughout the configuration. Only user configured data blocks are visible in the configurator even with this switch on.

Note: Data block configuration is optional (for advanced users).

2.5.2 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

Configure the following settings:

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.

MX Data block merging

Max gap for merging: The maximum gap where data blocks can merge together. For example if this value is 1, it means that a block made of consecutive addresses D20-D50 can be in joined with a block D52-D80.

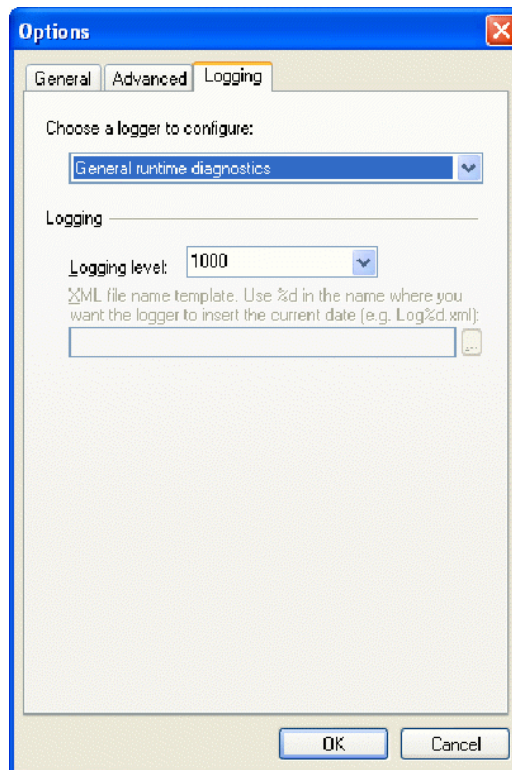
2.5.3 Logging Tab

The new version of the MX OPC Configurator allows the logging of various diagnostic information to the disk.

The logger is a simple logging tools build 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, that is only when the first message is being logged, the file is created on the disc.

To enable the logging, the **Logging** option page allows the selection of the log file for each of the logger categories, as seen on the next screenshot.



The sole purpose of the “Choose a logger to configure” combo box is to specify, which of the three loggers are configured by the rest of the controls. Its behavior is similar to the behavior of a tab control. At any given time, any combination of loggers can be active, each logging to a different file.

There are three logger categories; each of them is set up separately:

- **General Runtime Diagnostics** – Logs runtime events
- **Protocol & Wire** – Logs communication with the hardware.
- **User Changes Logging** – Logs configuration changes made to the database, like adding, deleting and modifying items.

The **General runtime diagnostics** logging is enabled by setting the Logging level lower than 1000 (Off). The other two loggers have a simple instead of the Logging level definition, checking the checkbox starts the logger, unselecting stops it. All the logging categories support specifying a logging file.

The meaning of the individual controls is as follows:

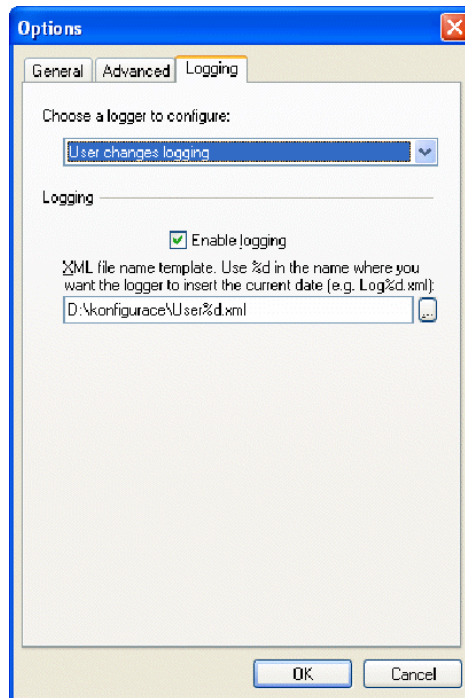
- **Choose a logger to configure** - This setting specifies, which logger category is configured by the remaining controls.
- **Logging level** - Specifies the logging verbosity. Setting to 1000 disables all logging. Visible only for **General runtime diagnostics**.

Other logging levels are: 800 – Error, 700 – Warning and 600 – Normal.

Beneath **Logging Level**, users are able to enter the XML file the logger will log into. This is enabled only when the **Logging Level** is set to less than 1000 or the **Enable logging** checkbox is checked.

- ... - Opens the standard save dialog for specifying the log file. The result from this dialog is inserted into the open text field. Enabled only when the **Logging Level** is set to less than 1000 or the **Enable logging** checkbox is checked.
- **Enable logging** - Enables / disables logging. Visible only for **User changes logging**.

Note: for all categories, the file setting specifies only the file template, not the actual file the logger will use. The logger will add the date in the form `yyyymmdd` to the file name in the place where the two characters `%d` are placed. For example, when the user enters the name "Log%d.xml", the date will be added right before the extension. If `%d` is used in the file name, the logger will log its messages to a different file each day. The output file is in XML format.



Address Space

3

3.1 Configuring the Address Space

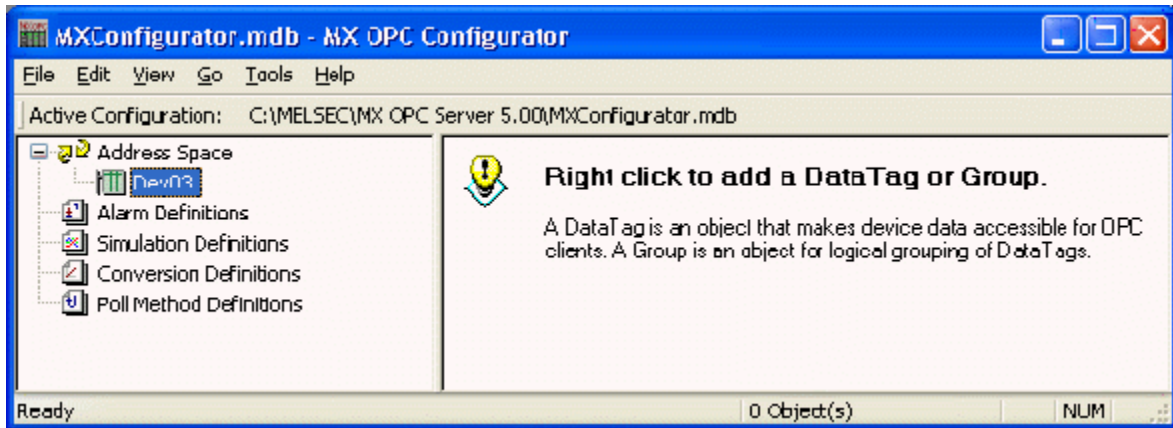
The MX I/O Server is the I/O driver core. The I/O Server contains objects and interfaces that:

- 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, data blocks, 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 blocks:** A data block is an addressable portion of a device. The data block contains the properties and methods that manage the behavior of a data block. Data blocks in the server's local memory correspond to data areas in the MX Runtime. When you add new data blocks to the server's local memory, you also add new data areas to the MX Runtime. Data blocks are not visible to the OPC client.
Note: Data block configuration is optional (for advanced users).
- **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).

3.1.1 Tree View



Address Space Tree Control

3.1.2 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.

3.2 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.

As of version 5.00, 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

3.2.1 Basic Device Properties

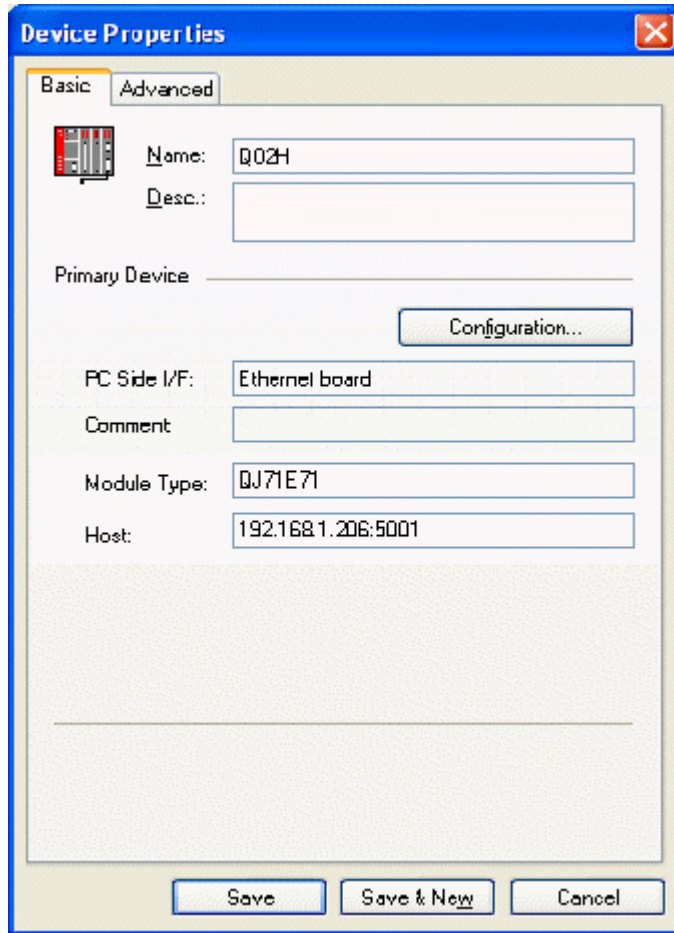
In the **Basic** tab of the **Device Properties** dialog box, shown below, configure the following settings:

The screenshot shows the 'Device Properties' dialog box with the 'Basic' tab selected. The 'Name' field is set to 'Dev01'. The 'Desc.' field is empty. The 'Primary Device' section has a 'Configuration...' button. The 'PC Side I/F' field is set to 'Serial'. The 'Comment' field is empty. The bottom buttons are 'Save', 'Save & New', and 'Cancel'.

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 on the left screenshot.

These two controls are grayed out and do appear only when I/F is configured as **Ethernet board**.



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 12 alphanumeric characters, including underscores (_) and hyphens (-).
- **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.
- **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.

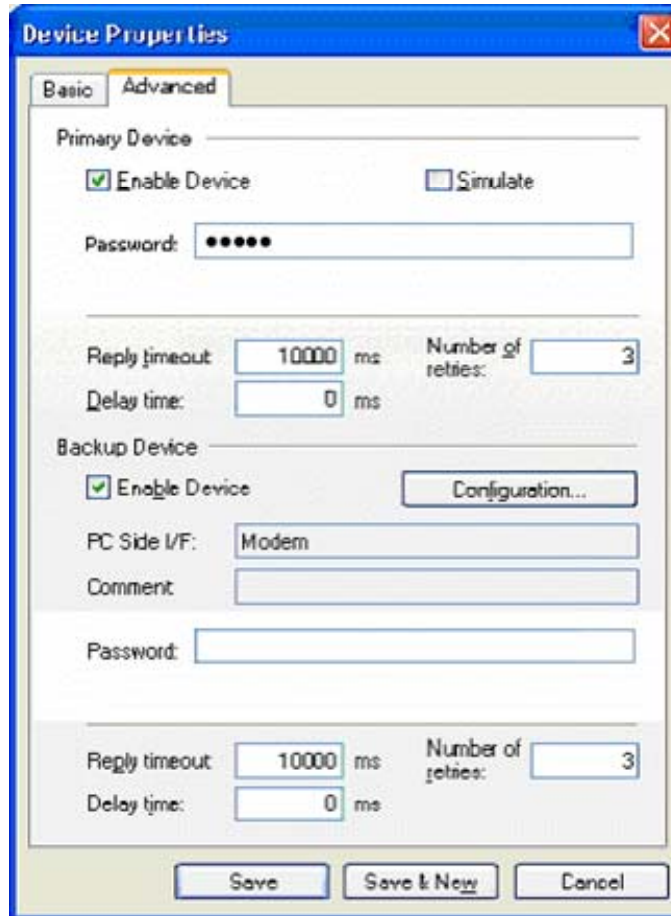
Communication Path	Description
Serial	Communication over a serial link can be either direct CPU or to a serial communications card, e.g. AJ71C24.
MELSECNET/10	Communication over MELSECNET/10 is via a special function module.
MELSECNET/H	Communication over MELSECNET/H is via a special function module.
CC-Link board	Communication is to an Ethernet card, e.g. AJ71E71.
Ethernet board	Communications via Ethernet using TCP/IP or UDP/IP protocols.
CPU board	Communications via a slot PLC
Q Series Bus	Communication over Q Series bus is via a back plane.
GX Simulator	Communication over GX Simulator is via GX Developer.
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 COM63	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.

3.2.2 Advanced Device Properties

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



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 or any of its data blocks. You may want to disable one or more devices in the following situations:
 - o You are swapping hardware for repair or maintenance and do not want to display errors.
 - o You do not need to collect data from all your devices and you want to reduce the communications load.
 - o You want to isolate a device for debugging.
- **Simulate:** Simulates polling of the channel by the driver when the check box is checked.
- The previous screenshot shows the **Advanced** page of the MX Device **Device Properties** sheet configured for a serial PC Side I/F.

As the screenshot above shows, when the device is configured as a Modem device, the user can also specify a password on the Advanced page.

Backup Device

- **Enable Device:** Enables the backup device.
- **Comment:** Displays any comments entered in the Communication Setting Wizard.
- **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.
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 is to an Ethernet card, e.g. AJ71E71.
Ethernet board	Communications via Ethernet using TCP/IP or UDP/IP protocols.
CPU board	Communications via a slot PLC
Q Series Bus	Communication over Q Series bus is via a back plane.
GX Simulator	Communication over GX Simulator is via GX Developer.
Modem	Communication over MODEM is via a special function module.
USB	Communication over a serial link can be direct to the CPU USB port.
COM1 to COM2 (if available)	Computer Ports
COM3 to COM63	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 block 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 the data block on 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 multiple data blocks. 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 the data block and all subsequent data blocks 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 blocks on the device and marks them 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 the data block as failed. If the driver has messages for other data blocks 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 are other data blocks defined for the device. If there are other data blocks, the driver fails all data blocks 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 a data block 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.

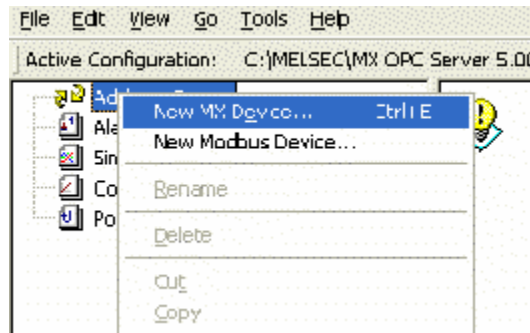
Example

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

3.2.3 Adding a New MX Device

To add a new device:

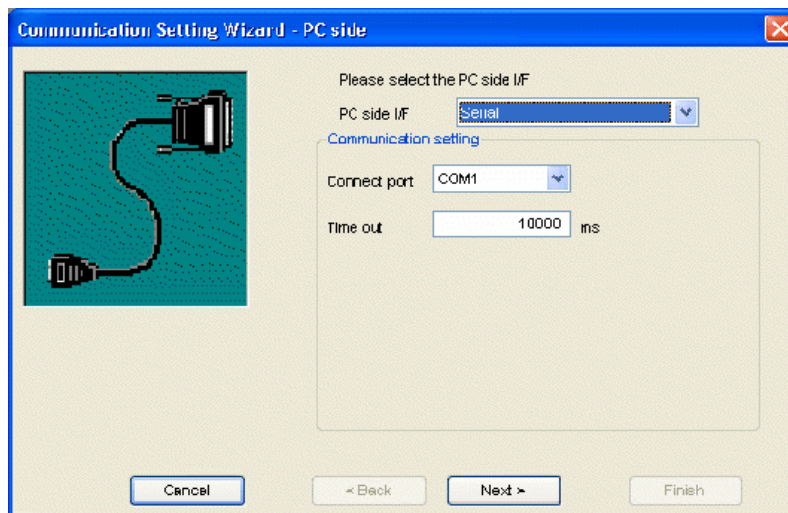
1. Right-click the **Address Space** tree control of the Configurator screen and select **New MX Device** from the pop-up menu, as shown in the figure below.



Adding a New MX Device

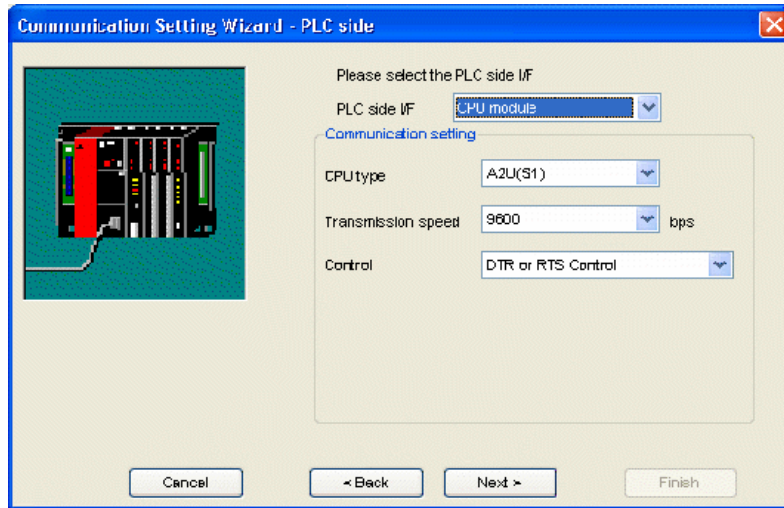
2. This opens the **Communication Setting Wizard**, as shown in the figure below. Specify the communication type to use on the PC side, as shown in the figure below. In the **PC Side I/F** field, select a connection type to use for the selected channel from the drop-down list. Click **Next** to continue.

Note: For more information, see the "Communication Setting Wizard" section.



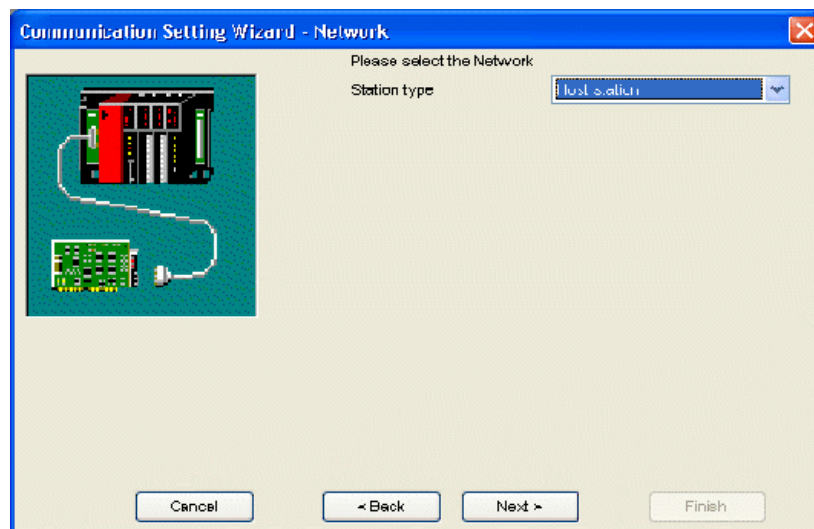
Setting up the Connection for the PC Side

3. 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.



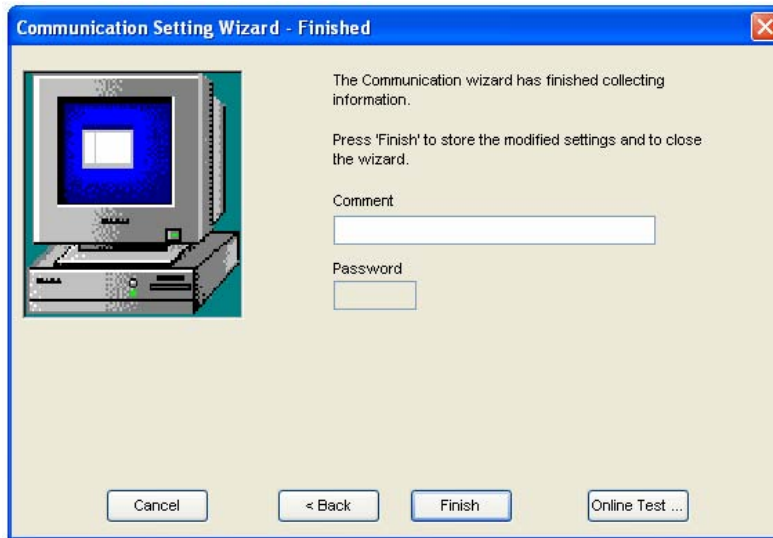
Setting up the Connection for the PLC Side

4. 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.



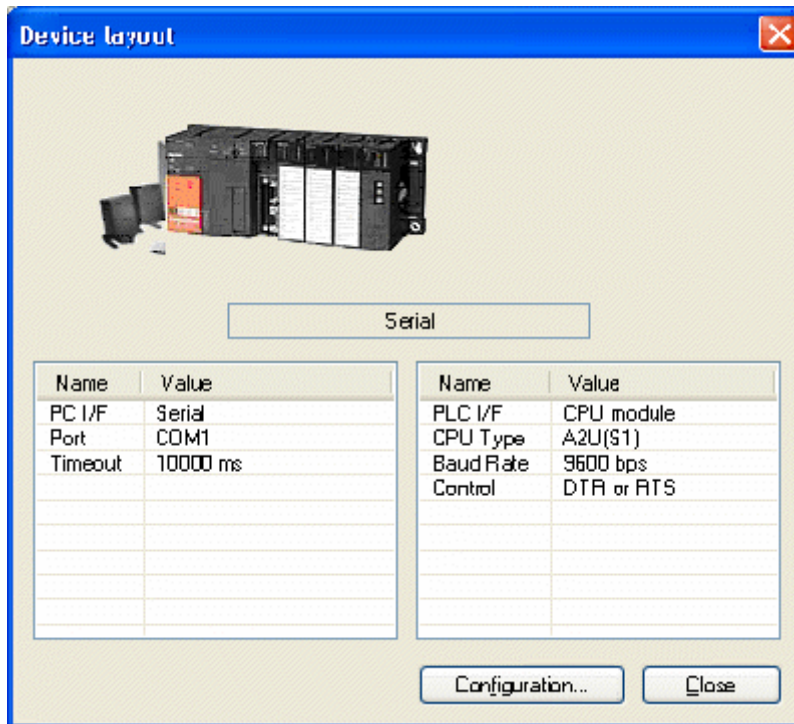
Configuring Network Settings

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



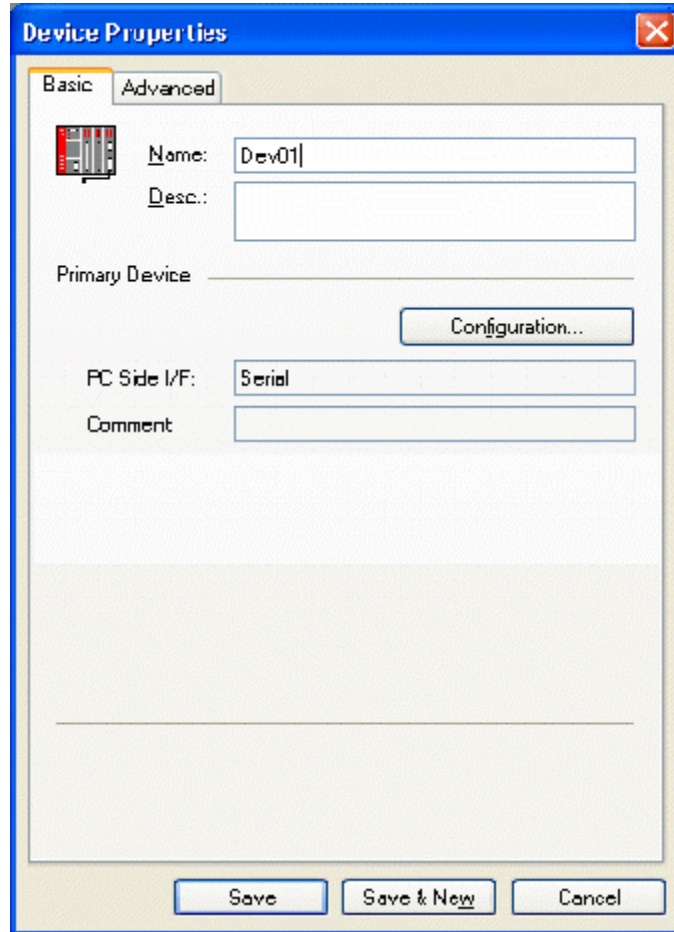
Completing Channel Setup

6. The **Device Layout** dialog box appears, as shown in the figure below. The left-hand side of the dialog lists the configuration properties for the PC side. You can double-click the image of the PC (or double-click the configuration properties list) to edit the PC side 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 **Configuration** button to return to the Communication Setting Wizard. Click the **Close** button when you are finished.



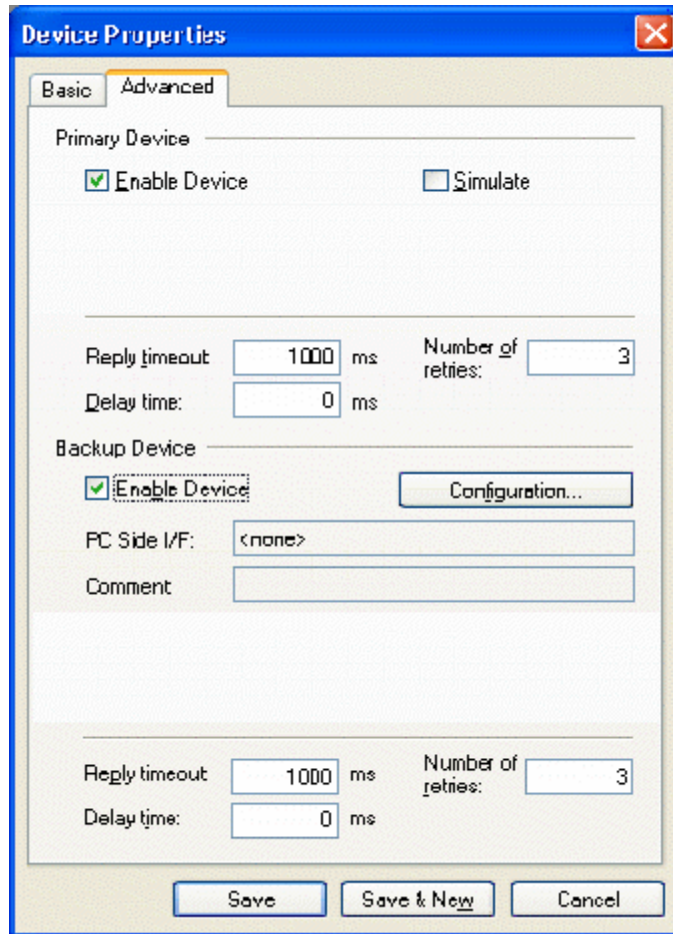
Device Layout

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



Configuring Device Properties

8. 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.
9. Click on the **Advanced** tab to set the advanced parameters for the **Primary** and **Backup** devices, as shown in the figure below.



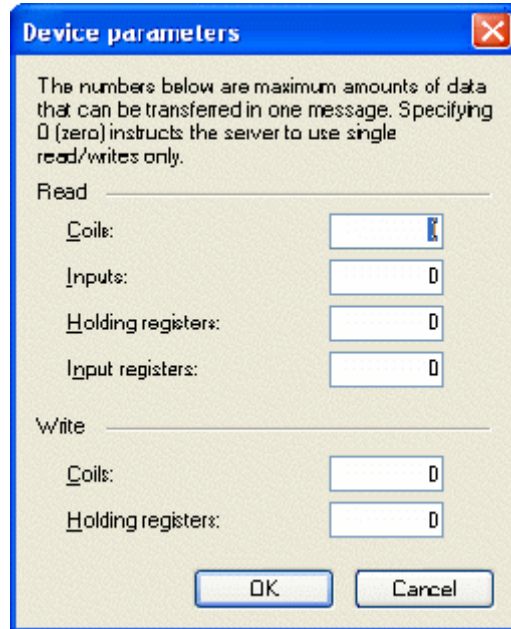
Device Properties: Advanced Tab

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

3.2.4 Basic Modbus Properties

Device Parameters

The **Modbus Device Parameters** dialog lets the user specify custom device parameters. The meaning of 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.



Modbus Device Parameters Dialog

READ

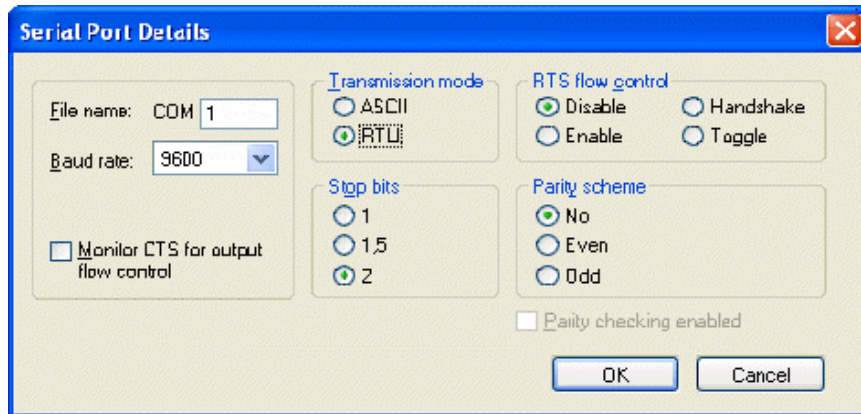
1. **Coils:** Enter the maximum number of coil bits sent in one message.
2. **Input:** Enter the maximum number of input bits sent in one message.
3. **Holding registers:** Enter the maximum number of holding registers sent in one message.
4. **Input registers:** Enter the maximum number of input registers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

Serial Port Details

The **Serial Ports Details** dialog allows the user to configure the serial port for the current device. The following screenshot shows this dialog. When the Parity checking enabled check box is not checked, it will force the server to ignore the parity bits in the message.



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".

3.2.5 Advanced Modbus Device Properties

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

The image shows a screenshot of the 'Device properties' dialog box, specifically the 'Advanced' tab. The dialog has a blue title bar with the text 'Device properties' and a close button. Below the title bar are two tabs: 'Basic' and 'Advanced', with 'Advanced' selected. The main area contains several sections:

- Polling rate (ms):** A label followed by a text box containing the value '1000'.
- Timeouts (ms):** A label followed by three text boxes: 'Reply timeout' (1000), 'Number of retries' (3), and 'Delay time' (5000).
- Optimizations:** A label followed by a paragraph of text: 'The numbers below indicate how much unused data can be transferred in one message to merge together addresses that are close but not adjacent.' Below this are two text boxes: 'Bits' (160) and 'Words' (5).

At the bottom of the dialog are three buttons: 'Save', 'Save & New', and 'Cancel'.

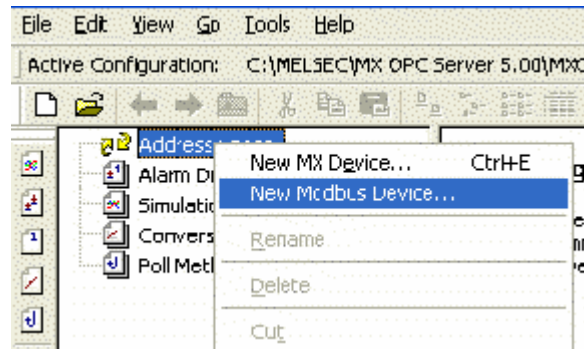
2. Enter the **Reply Timeout**.
3. Enter the **Number of retries** to set the number of consecutive read/write attempts that time out before the OPC server will suspend communication with the device.
4. Enter the **Delay time** to set the amount of time that the OPC server will wait before attempting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

3.2.6 Adding a New Modbus Device

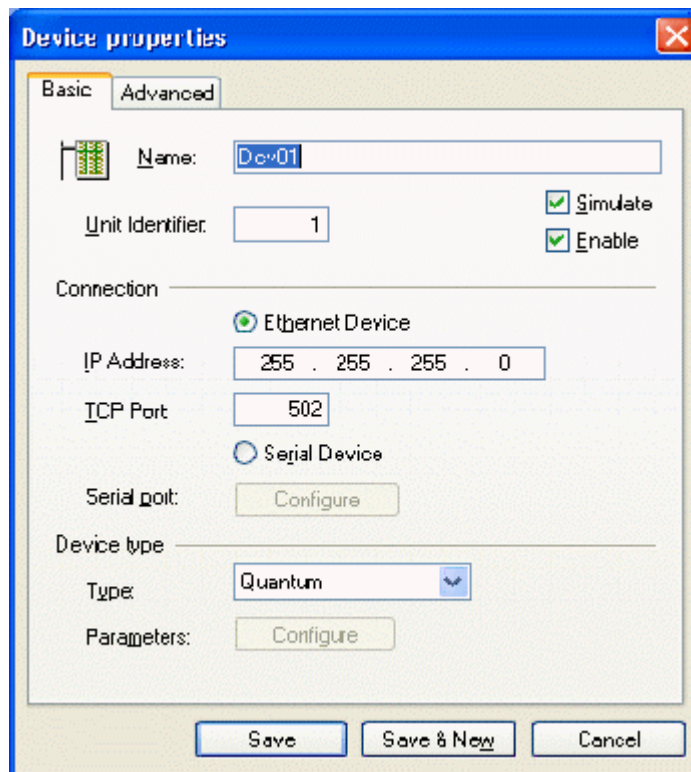
To add a new device:

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.

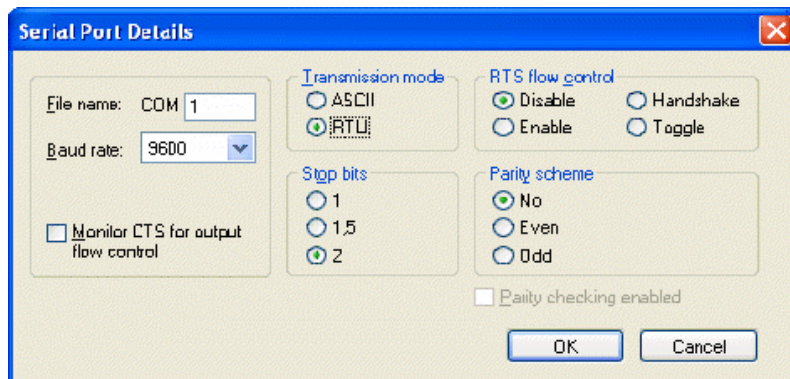


3. Enter the **Unit Identifier** which provides a unique ID for this device.
4. Select the **Simulate** (in order to simulate all Data Items on this device) and/or **Enable** (activate device) check boxes if required.
5. The radio button within the Connection section helps signify if the device is an **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).
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. The default Device type is set at Quantum.
10. **Parameters:** Opens a dialog for configuring the parameters of this modbus device.

Serial Port Details

The **Serial Ports Details** dialog allows the user to configure the serial port for the current device. The following screenshot shows this dialog. When the Parity checking enabled check box is not checked, it will force the server to ignore the parity bits in the message.



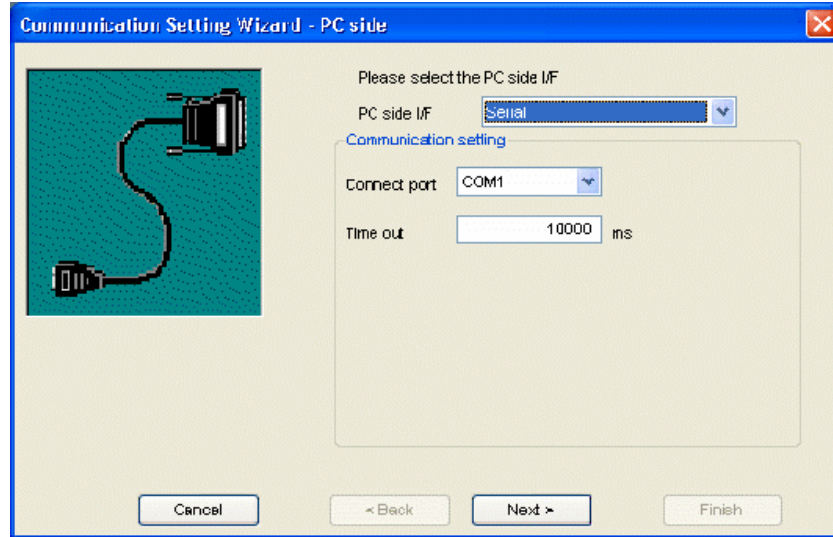
1. Enter the port name in the **File name** entry field.
2. Select the **Baud rate** from the pulldown menu.
3. The **Monitor CTS for output flow control** checkbox allows users to select whether or not the CTS signal for output flow control should be monitored.
4. Select either ASCII or RTU under **Transmission 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”.

3.2.7 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 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.
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 is to an Ethernet card, e.g. AJ71E71.
Ethernet board	Communications via Ethernet using TCP/IP or UDP/IP protocols.
CPU board	Communications via a slot PLC
Q Series Bus	Communication over Q Series bus is via a back plane.
GX Simulator	Communication over GX Simulator is via GX Developer.
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 COM63	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.

Board Number: Specifies the module (1-4) of the selected board. See the "Supported Devices and Device Ranges" section for more information.

Connect Module: Select a unit type from the drop-down list. See the "Supported Devices and Device Ranges" section for more information.

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):

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).

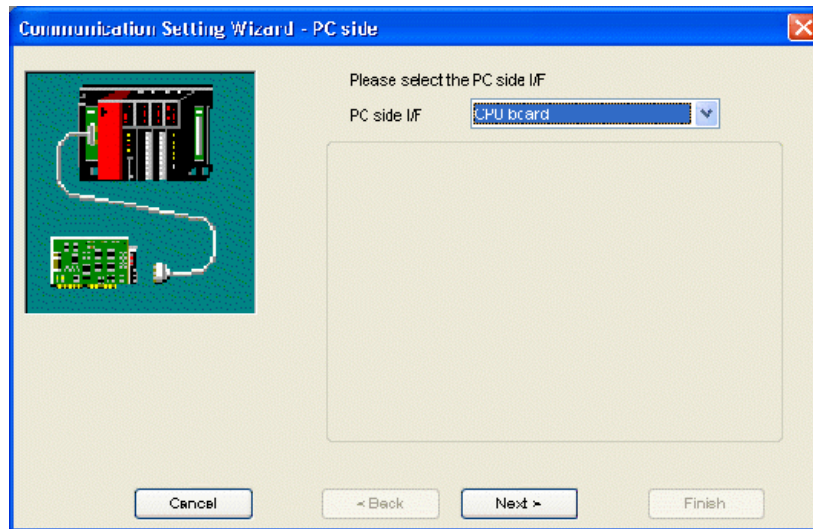
Notes:

- 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. See the "Supported Devices and Device Ranges" section for more information about CPUs.

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. See the "Supported Devices and Device Ranges" section for more information.

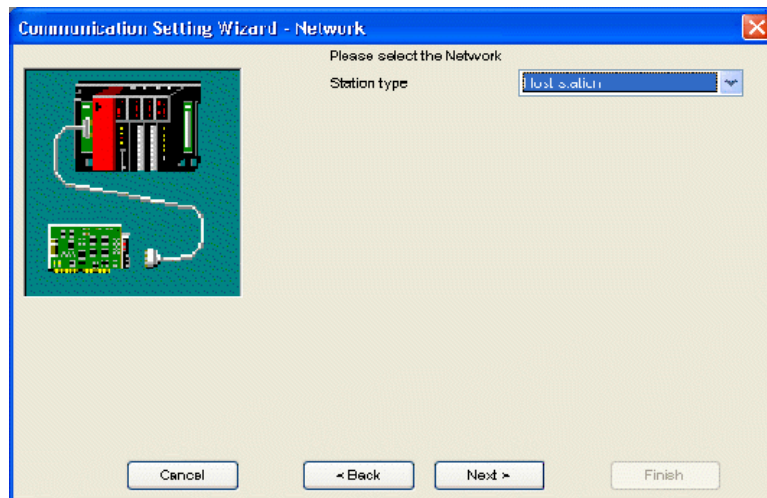
Port Number: This is the port address to be used when connecting to a Melsec Ethernet module. Valid range is 256-65535. The Ethernet must be configured to accept connections through this port.

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

Mode: Specify a mode for the selected communication module.

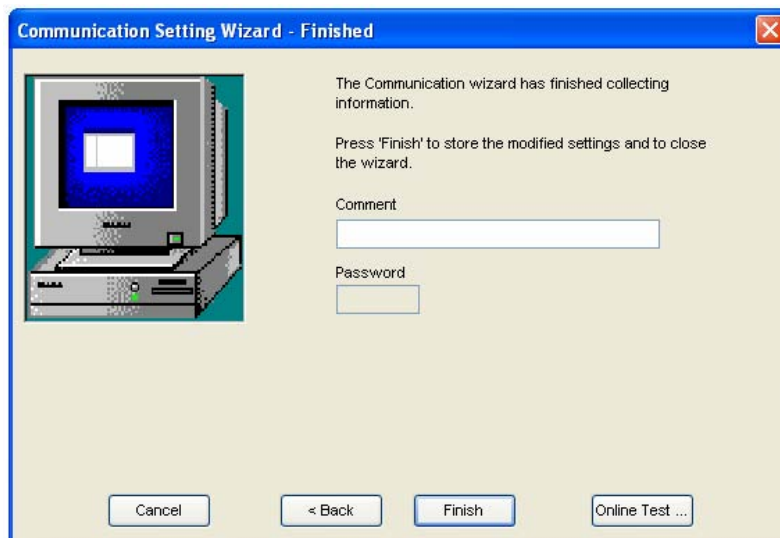
Specifying the Network Station Type

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. See "Supported PLC Devices and Communication Paths to Station Type 'Other Station'" for more information.



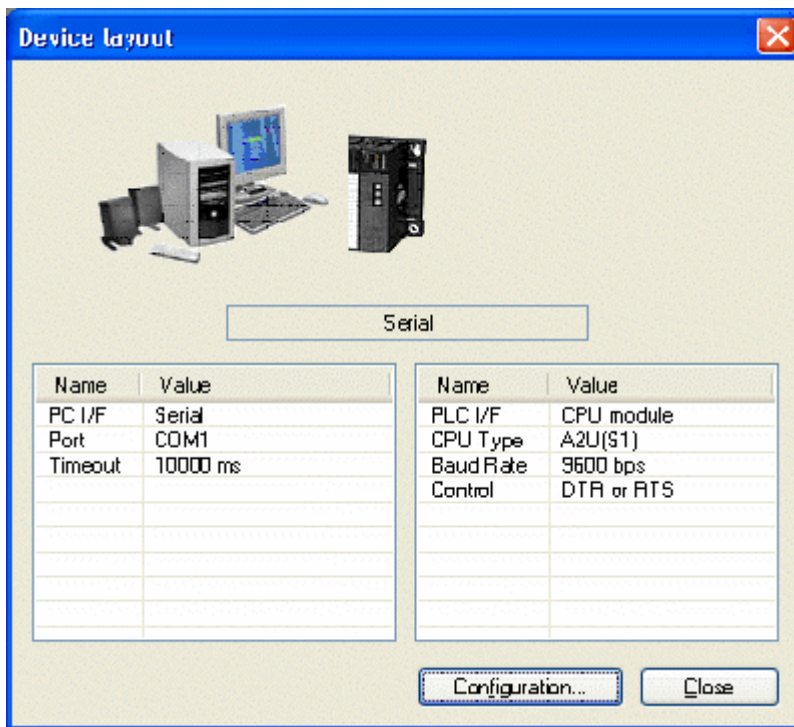
Configuring Network Settings

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



Completing Channel Setup

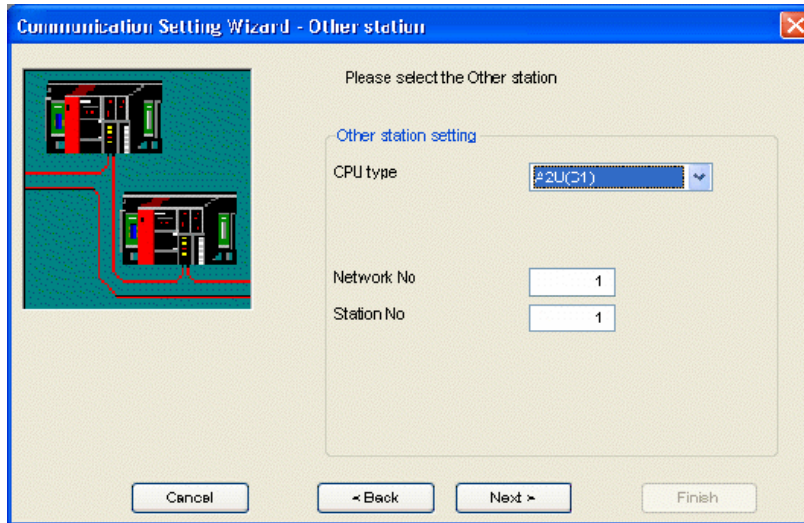
The **Device Layout** dialog box appears, as shown in the figure below. The left-hand side of the dialog lists the configuration properties for the PC side. You can double-click the image of the PC (or double-click the configuration properties list) to edit the PC side configuration properties. The right-hand 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 **Configuration** button to return to the Communication Setting Wizard. Click the **Close** button when you are finished.



Device Layout

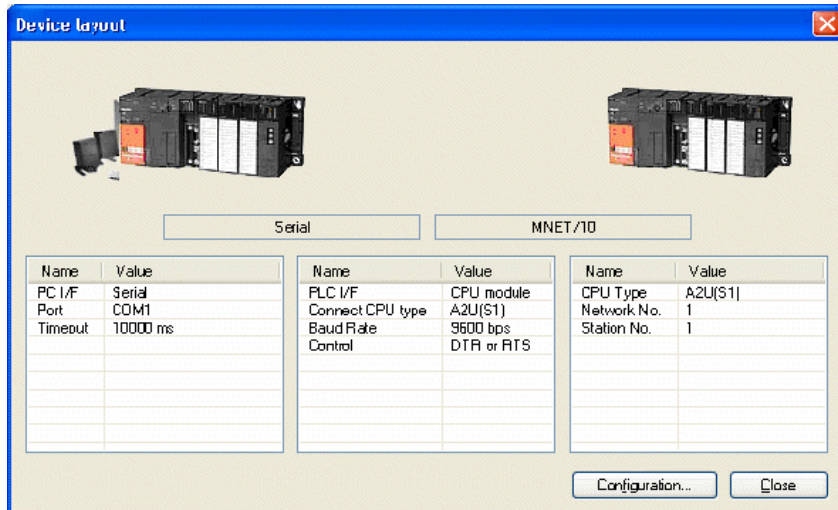
Properties for Station Type "Other Station"

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 **CPU Type** from the respective drop-down list, and specify a Network Number (if applicable) and a Station Number. Click the **Next** button when you are finished. Then click the **Finish** button to complete the communication channel setup.



Properties for "Other" Station

The **Device Layout** dialog box appears, as shown in the figure below. The left-hand side of the dialog lists the configuration properties for the PC side. You can double-click the image of the PC (or double-click the configuration properties list) to edit the PC side configuration properties. The middle 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. The right-hand side of the dialog lists the configuration properties for the selected network station ("other station") type. You can double-click the image of the station device (or double-click the configuration properties list) to edit the station configuration properties. You can also click the **Configuration** button to return to the Communication Setting Wizard. Click the **Close** button when you are finished.



Device Layout With Other Station

3.2.8 Supported Devices and Device Ranges

MX OPC server 5.0 supports the following CPUs, arranged in a PLC-family hierarchy.

Note: See the MX Component Operating Manual for more detailed information.

Communication Paths for PLC				
Communication (PC side)	Board (PC side)	Communication (PLC side)	Module (PLC side)	CPU Supported
1. Serial	COM1 to COM63 (RS-232C)	CPU Module	-	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH, Q02(H)-A, Q06H-A
				Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR
				A0J2H, A1FX, A1S(S1), A1SJ, A1SH, A1SJH, A1N, A2C, A2CJ, A2N(S1), A2S(S1), A2SH(S1), A3N, A2A(S1), A3A, A2U(S1), A2US(S1), A2AS(S1), A2AS(S1), A2AS-S30, A2AS- S60, A2USH-S1, A3U, A4U, A171SH, A172SH, A173UH(S1), A273UH(S3)
				FX0(S), FX0N, FX1, FX1S, FX1N(C), FXU/FX2C, FX2N(C)
		Computer Link C24 module	AJ71C24 AJ71UC24	Q02(H)-A, Q06H-A
				Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR
				A0J2H, A1FX, A1S(S1), A1SJ, A1SH, A1SJH, A1N, A2C, A2CJ, A2N(S1), A2S(S1), A2SH(S1), A3N, A2A(S1), A3A, A2US(S1), A2AS(S1), A2AS(S1), A2AS-S30, A2AS- S60, A2USH-S1, A3U, A4U, A171SH, A172SH, A173UH(S1), A273UH(S3)
				Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR
		G4 Module	Mode A	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH
				Q02(H)-A, Q06H-A A0J2H, A1FX, A1S(S1), A1SJ, A1SH, A1SJH, A1N, A2C, A2CJ, A2N(S1), A2S(S1), A2SH(S1), A3N, A2A(S1), A3A,

Communication Paths for PLC				
Communication (PC side)	Board (PC side)	Communication (PLC side)	Module (PLC side)	CPU Supported
				A2US(S1), A2AS(S1), A2AS(S1), A2AS-S30, A2AS-S60, A2USH-S1, A3U, A4U, A171SH, A172SH, A173UH(S1), A273UH(S3)
			Mode QnA	Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR
			Mode Q	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH,
2. USB	-	CPU Module	-	Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH,
3. MELSECNET /10	MELSECNET/10 board	MELSECNET/10	Modules 1-4	-
4. MELSECNET /H	MELSECNET /H board	MELSECNET /H	Modules 1-4	-
5. CC-Link	CC-Link board	CC-Link	Modules 1-4	-
6. ETHERNET	ETHERNET board	ETHERNET module	AJ71E71	Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR Protocol: UDP/TCP - Binary - ASCII A0J2H, A1FX, A1S(S1), A1SJ, A1SH, A1SJH, A1N, A2C, A2CJ, A2N(S1), A2S(S1), A2SH(S1), A3N, A2A(S1), A3A, A2US(S1), A2AS(S1), A2AS(S1), A2AS-S30, A2AS-S60, A2USH-S1, A3U, A4U, A171SH, A172SH, A173UH(S1), A273UH(S3)
			AJ71QE71	Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR Protocol: UDP/TCP - Binary - ASCII
			QJ71QE71	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH Protocol: UDP/TCP - Binary - ASCII
			GOT	-
7. CPU Board	CPU Board	-	-	A80BDE-A2USH-S1
8. Q Series Bus	Q Back Plane	-	Multiple CPU	Q00J, Q00, Q01, Q02(H),

Communication Paths for PLC				
Communication (PC side)	Board (PC side)	Communication (PLC side)	Module (PLC side)	CPU Supported
			No. 1 .. No. 3	Q06H, Q12H, Q25H, Q12PH, Q25PH
9. GX Simulator	-	-	GX Developer (offline debugging)	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH, Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR A0J2H, A1FX, A1S(S1), A1SJ, A1SH, A1SJH, A1N, A2C, A2CJ, A2N(S1), A2S(S1), A2SH(S1), A3N, A2A(S1), A3A, A2US(S1), A2AS(S1), A2AS-S30, A2AS-S60, A2USH-S1, A3U, A4U, A171SH, A172SH, A173UH(S1), A273UH(S3) FX0(S), FX0N, FX1, FX1S, FX1N(C), FXU/FX2C, FX2N(C)
10. MODEM	CPU board	Serial communication Q6TEL	-	Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR
		Serial communication A6TEL	-	A0J2H, A1FX, A1S(S1), A1SJ, A1SH, A1SJH, A1N, A2C, A2CJ, A2N(S1), A2S(S1), A2SH(S1), A3N, A2A(S1), A3A, A2US(S1), A2AS(S1), A2AS-S30, A2AS-S60, A2USH-S1, A3U, A4U, A171SH, A172SH, A173UH(S1), A273UH(S3)
		Serial communication FXCPU	FXCPU	FX1S, FX1N(C), FX2N(C)
		Serial communication	AJ71QC24N	Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR
		Serial communication	QJ71C24	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH

3.2.9 Supported PLC Devices and Communication Paths to Station Type "Other Station"

The **Other Station** is used to access the PLC through the following networks:

- MELSECNET/10(H)
- MELSECNET II
- ETHERNET
- C24

Communication Paths for PLC					
Communication (PC side)	Board (PC side)	Communication (PLC side)	Module (PLC side)	Other station	CPU Supported
1. Serial	COM1 to COM63 (RS-232C)	1.1 CPU module	-	1.1.1 MELSECNET/10(H) 1.1.2 MELSECNET(II)	Q02(H)-A, Q06H-A A0J2H, A1FX, A1S(S1), A1SJ, A1SH, A1SJH, A1N, A2C, A2CJ, A2N(S1), A2S(S1), A2SH(S1), A3N, A2A(S1), A3A, A2US(S1), A2AS(S1), A2AS-S30, A2AS-S60, A2USH-S1, A3U, A4U, A171SH, A172SH, A173UH(S1), A273UH(S3)
		1.2 C24 module	1.2.1 AJ71C24 1.2.2 AJ71UC24	1.2.1.1 MELSECNET/10(H) 1.2.1.2 MELSECNET(II) 1.2.1.3 C24	
			1.2.3 AJ71QC24	1.2.3.1 MELSECNET/10(H) 1.2.3.2 MELSECNET(II) 1.2.3.3 C24 1.2.3.4 ETHERNET 1.2.3.5 CC-Link	Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR

Communication Paths for PLC					
Communication (PC side)	Board (PC side)	Communication (PLC side)	Module (PLC side)	Other station	CPU Supported
			1.2.4 QJ71C24	1.2.3.1 MELSECNET/10(H)	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH, Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR A0J2H, A1FX, A1S(S1), A1SJ, A1SH, A1SJH, A1N, A2C, A2CJ, A2N(S1), A2S(S1), A2SH(S1), A3N, A2A(S1), A3A, A2US(S1), A2AS(S1), A2AS-S30, A2AS-S60, A2USH-S1, A3U, A4U, A171SH, A172SH, A173UH(S1), A273UH(S3)
				1.2.3.2 C24 1.2.3.3 ETHERNET	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH, Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR

Communication Paths for PLC					
Communication (PC side)	Board (PC side)	Communication (PLC side)	Module (PLC side)	Other station	CPU Supported
				1.2.3.4 CC-Link	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH, Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR A0J2H, A1FX, A1S(S1), A1SJ, A1SH, A1SJH, A1N, A2C, A2CJ, A2N(S1), A2S(S1), A2SH(S1), A3N, A2A(S1), A3A, A2US(S1), A2AS(S1), A2AS-S30, A2AS-S60, A2USH-S1, A3U, A4U, A171SH, A172SH, A173UH(S1), A273UH(S3)
		1.3 G4 Module	1.3.1 Mode A	-	-
			1.3.2 Mode QnA	1.3.2.1 MELSECNET/10(H) 1.3.2.2 MELSECNET(II) 1.3.2.3 C24 1.3.2.4 ETHERNET 1.3.1.5 CC-Link	Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR

Communication Paths for PLC					
Communication (PC side)	Board (PC side)	Communication (PLC side)	Module (PLC side)	Other station	CPU Supported
			1.3.3 Mode Q	1.3.3.1 MELSECNET/10(H)	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH, Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR A0J2H, A1FX, A1S(S1), A1SJ, A1SH, A1SJH, A1N, A2C, A2CJ, A2N(S1), A2S(S1), A2SH(S1), A3N, A2A(S1), A3A, A2US(S1), A2AS(S1), A2AS-S30, A2AS-S60, A2USH-S1, A3U, A4U, A171SH, A172SH, A173UH(S3) A273UH(S3)
				1.3.3.4 ETHERNET	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH, Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR
				1.3.3.3 C24	Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR

Communication Paths for PLC					
Communication (PC side)	Board (PC side)	Communication (PLC side)	Module (PLC side)	Other station	CPU Supported
2. USB	-	2.1 CPU Module	-	2.1.1 MELSECNET/10(H)	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH, Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR A0J2H, A1FX, A1S(S1), A1SJ, A1SH, A1SJH, A1N, A2C, A2CJ, A2N(S1), A2S(S1), A2SH(S1), A3N, A2A(S1), A3A, A2US(S1), A2AS(S1), A2AS-S30, A2AS-S60, A2USH-S1, A3U, A4U, A171SH, A172SH, A173UH(S3), A273UH(S3), A173UH(S1), A273UH(S3)
				2.1.2 ETHERNET	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH, Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1,

Communication Paths for PLC					
Communication (PC side)	Board (PC side)	Communication (PLC side)	Module (PLC side)	Other station	CPU Supported
					Q2AS(H)S1, Q3A, Q4A, Q4AR
				2.1.3 C24	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH, Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR
				2.1.4 CC-Link	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH, Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR A0J2H, A1FX, A1S(S1), A1SJ, A1SH, A1SJH, A1N, A2C, A2CJ, A2N(S1), A2S(S1), A2SH(S1), A3N, A2A(S1), A3A, A2US(S1), A2AS(S1), A2AS-S30, A2AS-S60, A2USH-S1, A3U, A4U,

Communication Paths for PLC					
Communication (PC side)	Board (PC side)	Communication (PLC side)	Module (PLC side)	Other station	CPU Supported
					A171SH, A172SH, A173UH(S1) A273UH(S3)
3. MELSECNET/10 (H)	3.1 MELSECNE T/10 (H) board	MELSECNET/10 (H)	3.1.1 Other station(Sing le) Modules 1-4	-	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH, Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR A0J2H, A1FX, A1S(S1), A1SJ, A1SH, A1SJH, A1N, A2C, A2CJ, A2N(S1), A2S(S1), A2SH(S1), A3N, A2A(S1), A3A, A2US(S1), A2AS(S1), A2AS-S30, A2AS-S60, A2USH-S1, A3U, A4U, A171SH, A172SH, A173UH(S1), A273UH(S3)

Communication Paths for PLC					
Communication (PC side)	Board (PC side)	Communication (PLC side)	Module (PLC side)	Other station	CPU Supported
				3.2.1 MELSECNET/10(H)	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH, Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR A0J2H, A1FX, A1S(S1), A1SJ, A1SH, A1SJH, A1N, A2C, A2CJ, A2N(S1), A2S(S1), A2SH(S1), A3N, A2A(S1), A3A, A2US(S1), A2AS(S1), A2AS-S30, A2AS-S60, A2USH-S1, A3U, A4U, A171SH, A172SH, A173UH(S1), A273UH(S3)
				2.2.2 ETHERNET	Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR
				2.2.3 C24	Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR
4. CC-Link	4.1 CC-Link board	MELSECNET/10 (H)	4.1.1 Other station(Sing le) Modules 1-4	-	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH,

Communication Paths for PLC					
Communication (PC side)	Board (PC side)	Communication (PLC side)	Module (PLC side)	Other station	CPU Supported
					Q25PH, Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR A0J2H, A1FX, A1S(S1), A1SJ, A1SH, A1SJH, A1N, A2C, A2CJ, A2N(S1), A2S(S1), A2SH(S1), A3N, A2A(S1), A3A, A2US(S1), A2AS(S1), A2AS-S30, A2AS-S60, A2USH-S1, A3U, A4U, A171SH, A172SH, A173UH(S1), A273UH(S3)
				4.2.1 MELSECNET/10(H) 4.2.2 ETHERNET	Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR
5. ETHERNET	5.1 ETHERNET board	ETHERNET module	5.2 AJ71E71 Protocol: UDP/TCP - Binary - ASCII	5.2.1 MELSECNET/10(H) 5.2.2 MELSECNET(II)	Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR A0J2H, A1FX, A1S(S1), A1SJ, A1SH, A1SJH, A1N, A2C, A2CJ, A2N(S1), A2S(S1), A2SH(S1), A3N, A2A(S1),

Communication Paths for PLC					
Communication (PC side)	Board (PC side)	Communication (PLC side)	Module (PLC side)	Other station	CPU Supported
					A3A, A2US(S1), A2AS(S1), A2AS-S30, A2AS-S60, A2USH-S1, A3U, A4U, A171SH, A172SH, A173UH(S1), A273UH(S3)
			5.3 AJ71QE71 Protocol: UDP/TCP - Binary - ASCII	5.3.1 MELSECNET/10(H) 5.3.2 MELSECNET(II) 5.3.3 C24	Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR
			5.4 QJ71QE71 Protocol: UDP/TCP - Binary - ASCII	5.4.1 MELSECNET/10(H)	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH, Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR A0J2H, A1FX, A1S(S1), A1SJ, A1SH, A1SJH, A1N, A2C, A2CJ, A2N(S1), A2S(S1), A2SH(S1), A3N, A2A(S1), A3A, A2US(S1), A2AS(S1), A2AS-S30, A2AS-S60, A2USH-S1, A3U, A4U,

Communication Paths for PLC					
Communication (PC side)	Board (PC side)	Communication (PLC side)	Module (PLC side)	Other station	CPU Supported
					A171SH, A172SH, A173UH(S1), A273UH(S3)
				5.4.2 ETHERNET	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH, Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR
				5.4.3 C24 5.4.4 CC-Link	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH, Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR A0J2H, A1FX, A1S(S1), A1SJ, A1SH, A1SJH, A1N, A2C, A2CJ, A2N(S1), A2S(S1), A2SH(S1), A3N, A2A(S1), A3A, A2US(S1), A2AS(S1), A2AS-S30, A2AS-S60, A2USH-S1, A3U, A4U, A171SH, A172SH, A173UH(S1),

Communication Paths for PLC					
Communication (PC side)	Board (PC side)	Communication (PLC side)	Module (PLC side)	Other station	CPU Supported
					A273UH(S3)
			GOT	-	-
6. CPU Board	6.1 CPU board	Serial communication	-	6.2.1 MELSECNET/10(H) 6.2.2 MELSECNET(II)	Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR A0J2H, A1FX, A1S(S1), A1SJ, A1SH, A1SJH, A1N, A2C, A2CJ, A2N(S1), A2S(S1), A2SH(S1), A3N, A2A(S1), A3A, A2US(S1), A2AS(S1), A2AS-S30, A2AS-S60, A2USH-S1, A3U, A4U, A171SH, A172SH, A173UH(S1), A273UH(S3)
7. MODEM		7.1 Serial communication Q6TEL	-	7.1.1 MELSECNET/10(H) 7.1.2. MELSECNET(II) 7.1.3 C24 7.1.4 ETHERNET	Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR
		7.2 Serial communication A6TEL	-	7.2.1 MELSECNET/10(H) 7.2.2. MELSECNET(II)	Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR A0J2H, A1FX,

Communication Paths for PLC					
Communication (PC side)	Board (PC side)	Communication (PLC side)	Module (PLC side)	Other station	CPU Supported
					A1S(S1), A1SJ, A1SH, A1SJH, A1N, A2C, A2CJ, A2N(S1), A2S(S1), A2SH(S1), A3N, A2A(S1), A3A, A2US(S1), A2AS(S1), A2AS-S30, A2AS-S60, A2USH-S1, A3U, A4U, A171SH, A172SH, A173UH(S1), A273UH(S3)
		7.3 Serial communication FXCPU	-	-	-
		7.4 Serial communication	AJ71QC24N	7.4.1 MELSECNET/10(H) 7.4.2. MELSECNET(II) 7.4.3 C24 7.4.4 ETHERNET	Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR

Communication Paths for PLC					
Communication (PC side)	Board (PC side)	Communication (PLC side)	Module (PLC side)	Other station	CPU Supported
		7.5 Serial communication	QJ71C24N	7.5.1 MELSECNET/10(H) 7.5.2 CC-Link	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH, Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR A0J2H, A1FX, A1S(S1), A1SJ, A1SH, A1SJH, A1N, A2C, A2CJ, A2N(S1), A2S(S1), A2SH(S1), A3N, A2A(S1), A3A, A2US(S1), A2AS(S1), A2AS-S30, A2AS-S60, A2USH-S1, A3U, A4U, A171SH, A172SH, A173UH(S1), A273UHS3)
				7.5.3. ETHERNET 7.5.4 C24 7.5.5 CC-Link	Q00J, Q00, Q01, Q02(H), Q06H, Q12H, Q25H, Q12PH, Q25PH, Q02(H)-A, Q06H-A Q2A, Q2AS(H), Q2AS1, Q2AS(H)S1, Q3A, Q4A, Q4AR

3.3 Data Blocks

A data block is an addressable portion of a device. Data blocks in the server's local memory correspond to data areas in the MX Runtime. When you add new data blocks to the server's local memory, you also add new data areas to the MX Runtime. Data blocks are not visible to the OPC client.

Note: Data block configuration is optional (for advanced users). By default, the configurator runs in **basic configuration mode**. In basic mode, data blocks are hidden in the **Address Space** configuration. To switch to **advanced configuration mode**, go to the **Tools** menu and select **Options**. Go to the **General** tab and check the **Advanced Mode (Show Data Blocks)** check box. When this check box is checked, data blocks will be visible and editable throughout the configuration.

3.3.1 Basic Data Block Properties

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

The screenshot shows the 'Data Block Properties' dialog box with the 'Basic' tab selected. The 'Name' field contains 'Block00001'. The 'Desc.' field is empty. Under 'General Setup', 'Starting Addr.' is 'M0', 'Ending Addr.' is 'M255', and 'Polling Method' is '1000ms'. The 'Browse...' button is next to the 'Starting Addr.' field. At the bottom are 'Save', 'Save & New', and 'Cancel' buttons.

Data Block Properties: Basic Tab

- **Name:** Specifies the name of the selected data block. This name is mainly for your reference. It is helpful to use descriptive names you can recognize so that modifying or accessing information on the data block is easier. The name can be up to 12 alphanumeric characters, including underscores (_) and hyphens (-).

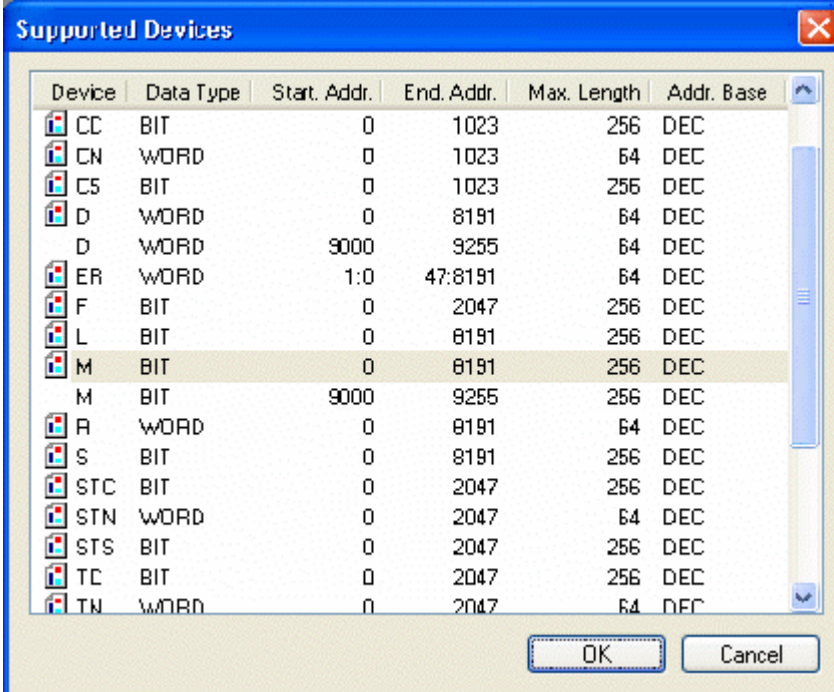
NOTE: Each data block name must be unique within the whole driver configuration.

- **Description:** Allows you to enter text about the selected data block. 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 data block. The description can be up to 40 alphanumeric characters and symbols.
- **Starting Address:** See "Setting Data Block Address Properties" below.
- **Ending Address:** See "Setting Data Block Address Properties" below.
- **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. See "Setting Data Block Timing Properties" below for more information.
- **Save:** Saves all changes specified in the properties dialog box. The data block appears under the selected device in the tree control of the configurator.
- **Save & New:** Saves all changes specified in the properties dialog box and immediately starts configuration of a new data block.
- **Cancel:** Closes the properties dialog box.

Setting Data Block Address Properties

The **Starting Address** and **Ending Address** define the location in the device that the data block represents. You must always enter a starting address. This is the point in the device from which the data block starts. You can then enter an ending address to complete the entire block.

In the **Basic** tab of the **Data Block Properties** dialog box, click the **Browse** button next to the **Starting Address** field. This displays a list of **Supported Devices**, as shown in the figure below. This list contains default data type, starting address, ending address, and maximum address length settings for each device type. To select a device, click on the device to highlight the row, and then click **OK**.



Device	Data Type	Start. Addr.	End. Addr.	Max. Length	Addr. Base
CC	BIT	0	1023	256	DEC
CN	WORD	0	1023	64	DEC
C5	BIT	0	1023	256	DEC
D	WORD	0	8191	64	DEC
D	WORD	9000	9255	64	DEC
ER	WORD	1:0	47:8191	64	DEC
F	BIT	0	2047	256	DEC
L	BIT	0	8191	256	DEC
M	BIT	0	8191	256	DEC
M	BIT	9000	9255	256	DEC
R	WORD	0	8191	64	DEC
S	BIT	0	8191	256	DEC
STC	BIT	0	2047	256	DEC
STN	WORD	0	2047	64	DEC
STS	BIT	0	2047	256	DEC
TC	BIT	0	2047	256	DEC
TN	WORD	0	2047	64	DEC

Supported Devices

Example

You want to create a data block named DATABLOCK-C that starts at address 5 and has a length of 8 of register D.

To do this, enter D5 in the **Starting Address** field and D12 in the **Ending Address** field.

DATABLOCK-C uses the following location in the device:

D Registers

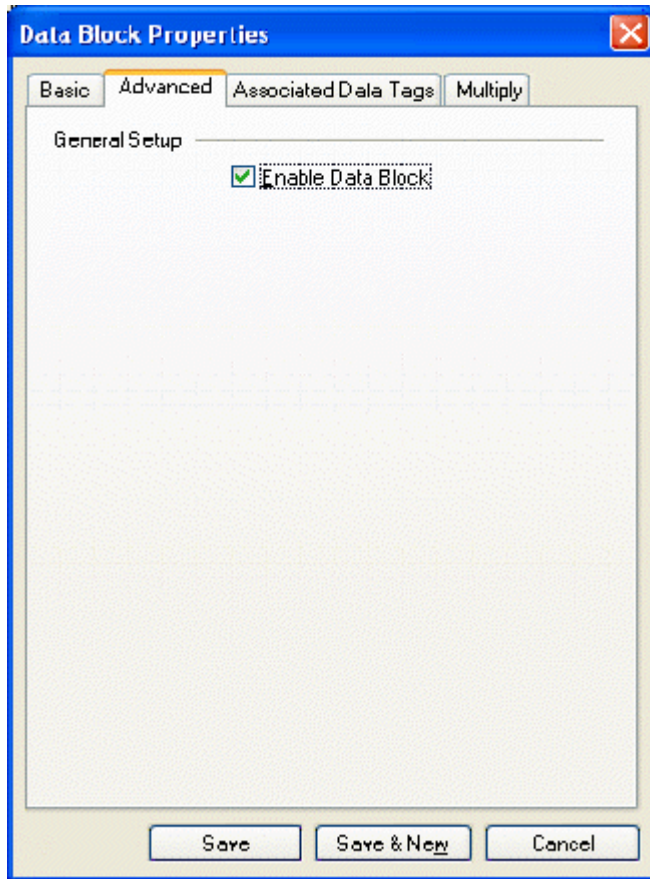
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	2
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0

Valid Data Block Registers

X, Y, L, M, SM, SB, TS, TC, TN, CS, CC, CN, STS, STC, STN, D, SD, SW, Z, V, R, ER, B, W, ZR, S, F

3.3.2 Advanced Data Block Properties

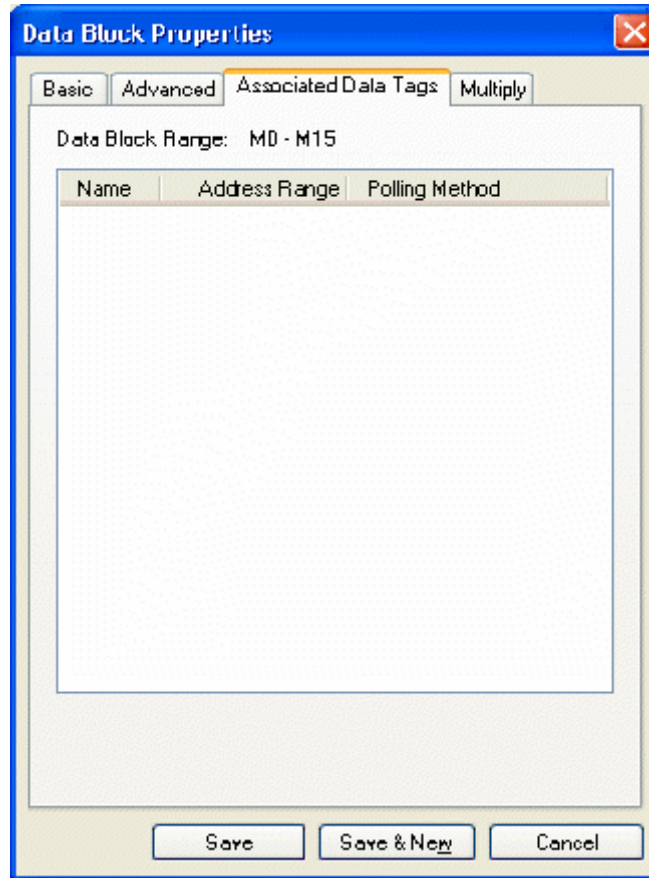
In the Advanced tab of the Data Block Properties dialog box, shown below, you can enable or disable the data block



Advanced Data Block Properties

3.3.3 Data Block Properties: Associated Data Tags

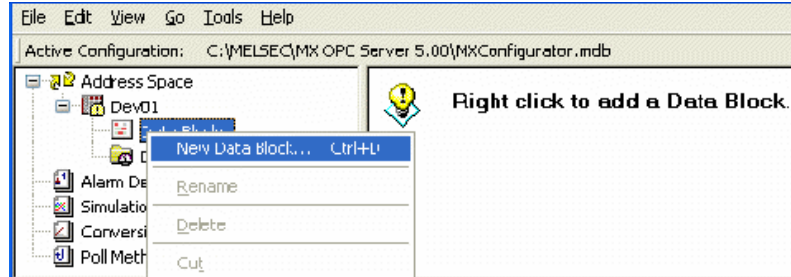
The **Associated Data Tags** tab of the **Data Block Properties** dialog box, shown below, lists all tags that are tied to each data block. It also lists the **Address Range** and the selected **Polling Method** for each tag.



3.3.4 Adding a New Data Block

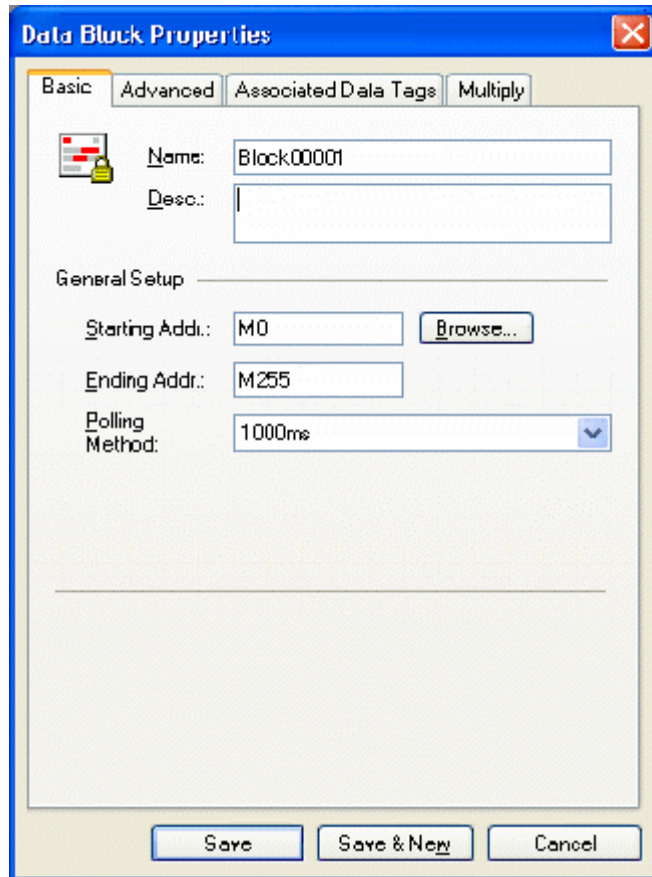
To add a new data block:

1. Right-click a device on the **Data Blocks** item under a device and select **New Data Block** from the pop-up menu, as shown in the figure below.



Adding a New Data Block

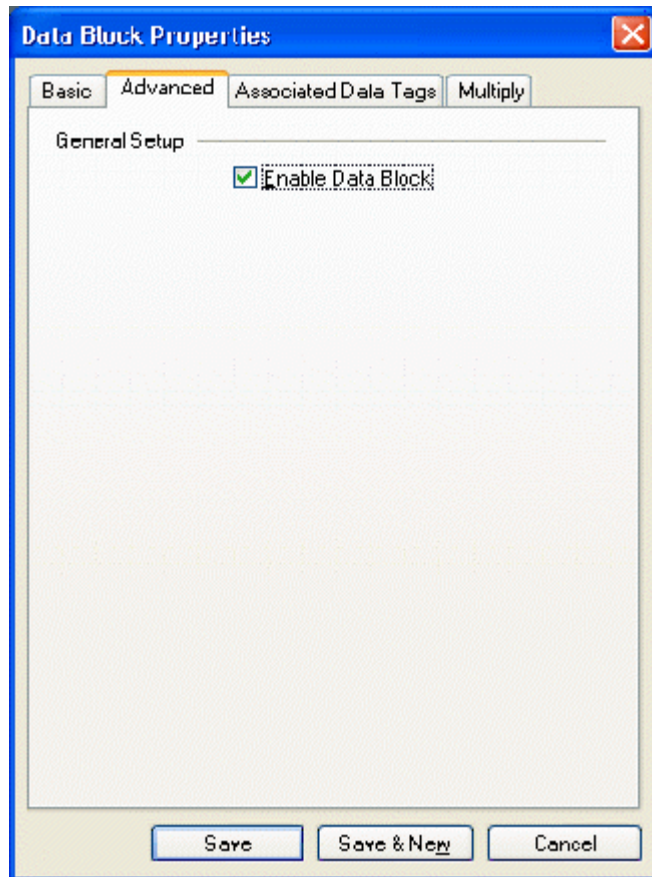
2. The **Basic** tab of the **Data Block Properties** dialog box appears, as shown in the figure below.



Configuring Data Block Properties

3. In the **Name** field, type a name for the new data block, and type a description for the data block (optional).
4. Set the parameters for the **I/O Address Setup**. Click the **Browse** button to see a list of **Supported Devices**.

5. Click on the **Advanced** tab, as shown in the figure below.



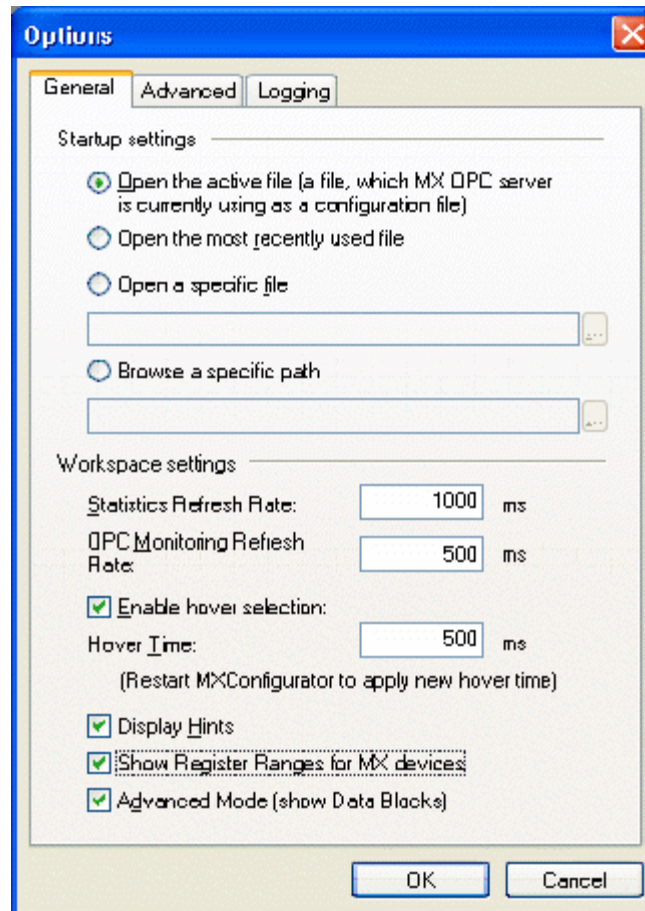
Data Block Properties: Advanced Tab

6. Check the **Enable Data Block** check box to activate the data block.
7. When you have finished configuring the data block properties, click the **Save** button. The new data block appears under the **Device** tree control.

3.3.5 Auto-Creation of Data Blocks

The configurator has an optional advanced configuration mode that allows you to view and edit data blocks. To switch to **advanced configuration mode**, go to the **Tools** menu and select **Options**. Go to the **General** tab and check the **Advanced Mode (Show Data Blocks)** check box, as shown in the figure below. When this check box is checked, data blocks will be visible and editable throughout the configuration.

The runtime automatically creates data blocks for tags, which have no data block defined in the configurator.



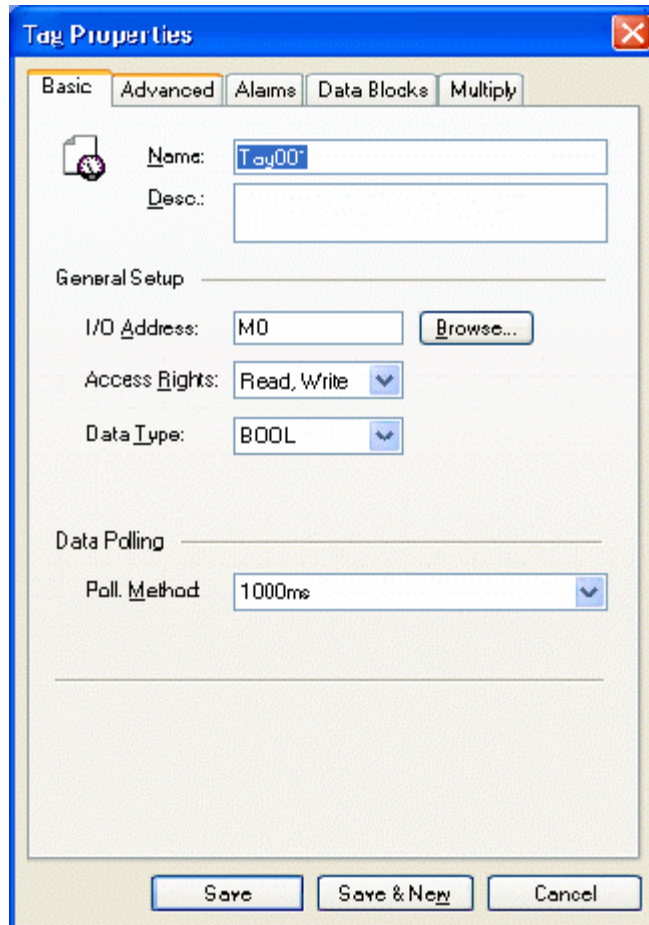
Options Dialog Box: General Tab

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 'Tag00'. 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 40 alphanumeric characters and symbols.
- **I/O Address:** Indicates the data block starting address.
- **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
STRING	Text stored internal as a 16-bit integer.
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.
ARRAY	Array except for STRING

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.

Device	Device	Data Type	Device Range	Size Bits
Link relay	B	BOOL	0-32687	1
Counter coil	CC	BOOL	0-23087	1
Counter Present value	CN	WORD, UINT, INT	0-23087	16
Counter Contact	CS	BOOL	0-23087	1
Data register	D	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	0-25983	16
Annunciator	F	BOOL	0-32687	1
Latch relay	L	BOOL	0-32687	1
Internal relay	M	BOOL	0-32767	1
File register	R	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	0-32767	16
Link special relay	SB	BOOL	0-2047	1
Special register	SD	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	0-2047	16
Special relay	SM	BOOL	0-2047	1
Retentive Timer Coil	SC	BOOL	0-32767	1
Retentive Timer Present value	SN	WORD, UINT, INT	0-32767	16
Retentive Timer Contact	SS	BOOL	0-32767	1

Device	Device	Data Type	Device Range	Size Bits
Link special register	SW	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	0-2047	16
Timer Coil	TC	BOOL	0-23087	1
Timer Present value	TN	WORD, UINT, INT	0-23087	16
Timer Contact	TS	BOOL	0-23087	1
Index register	V	WORD, UINT, INT	0-6	16
Link register	W	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	0-657F	16
Input relay	X	BOOL	0-8191	1
Output relay	Y	BOOL	0-8191	1
Index register	Z	WORD, UINT, INT	0-15	16
Step relay	S	BOOL	0-8191	1
Function input	FX	BOOL	0-15	1
Function output	FY	BOOL	0-15	1
Function register	FD	WORD, UINT, INT	0-4	16
Extended File register	ER	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	0-1042431	16
File register	ZR	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	0-1042432	16
Accumulator	A	DWORD, WORD, UDINT, UINT, DINT, INT, REAL	0-1	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.

3.4.1.1 Bit Access for Integers

For some devices, it is possible to set up a boolean data tag using an integer address and adding a suffix to the address, for example using the address D0, adding the suffix .1 (giving D0.1) will result in having a Boolean tag instead of an integer tag. It is not possible for any register type, though.

3.4.1.2 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 following tables define the valid default and extended address ranges for all the register types. Note that the register types available will depend on the device type selected. The default ranges correspond to the standard default parameter settings used by the hardware. The extended ranges, if used, cause a warning that these hardware parameters may need configuring. For more information see Section 8 of the ActiveX Communication Support Tool Operating Manual.

Table 1. Address Ranges for All ACTs Except for FX (Serial)

Device Name	Default Range	Extended Range
FX	0 – F	0 – F
FY	0 – F	0 – F
FD	0 – 4	0 – 4
SM	0 – 2047	0 – 2047
SD	0 – 2047	0 – 2047
X	0 – 1FFF	0 – 1FFF
Y	0 – 1FFF	0 – 1FFF
M	0 – 8191	0 – 32767
L	0 – 32687	0 – 32687
F	0 – 32687	0 – 32687
V	0 – 2047	0 – 32767
B	0 – 32687	0 – 32687
W	0 – 657F	0 – 657F
D	0 – 25983	0 – 25983
TS	0 – 23087	0 – 23087
TC	0 – 23087	0 – 23087
TN	0 – 23087	0 – 23087
STS	0 – 23087	0 – 23087
STC	0 – 23087	0 – 23087
STN	0 – 23087	0 – 23087
CC	0 – 23087	0 – 23087
CN	0 – 23087	0 – 23087
CS	0 – 23087	0 – 23087
SB	0 – 7FF	0 – 7FF

Device Name	Default Range	Extended Range
SW	0 – 7FF	0 – 7FF
S	0 – 8191	0 – 8191
Z	0 – 15	0 – 15
R	0 – 1042432	0 – 1042432
ZR	0 – 1042432	0 – 1042432
ER	(1 – 256) : (0 – 32767)	(1 – 256) : (0 – 32767)
A	0 – 1	0 – 1

Table 2. Address Ranges for FX (Serial)

Device Name	Range
M	0 – 3071, 8000 – 8255
S	0 – 999
TS	0 – 255
TC	0 – 255
TN	0 – 255
CC	0 – 255
CN	0 – 255
CS	0 – 255
D	0 – 7999, 8000 – 8255
V	0 – 7
Z	0 – 7
X	0 – 377
Y	0 – 377

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, Z, V, R, ER, B, W, ZR, S, F

Ending Address

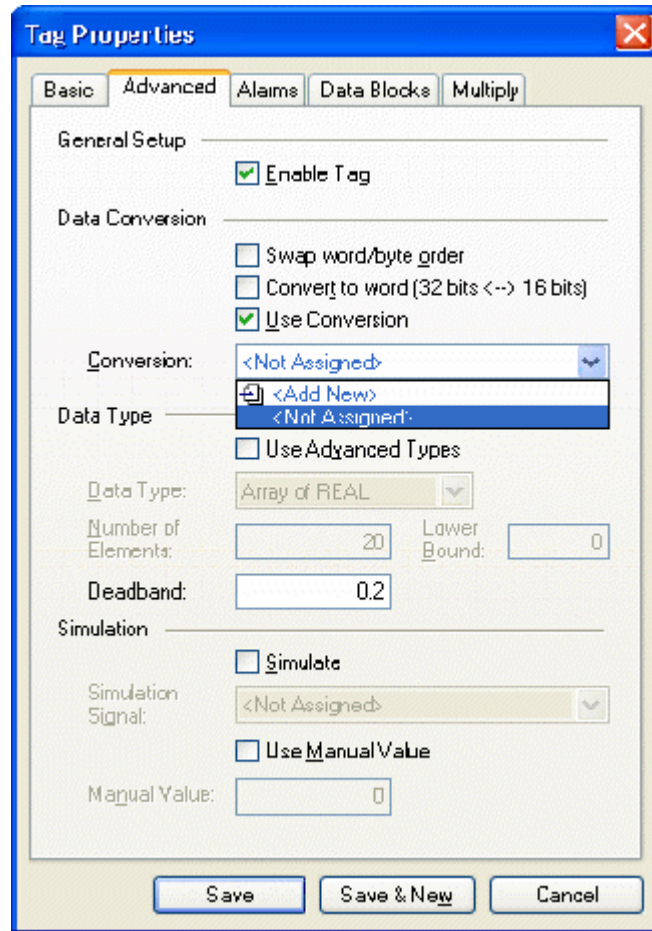
Specifies the location in the device where the selected data block ends.

Valid Data Block Registers

X, Y, L, M, SM, SB, TS, TC, TN, CS, CC, CN, STS, STC, STN, D, SD, SW, Z, V, R, ER, B, W, ZR, S, F

3.4.2 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.
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.
ARRAY	Array except for STRING

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.

Device	Device	Data Type	Device Range	Size Bits
Link relay	B	BOOL	0-32687	1
Counter coil	CC	BOOL	0-23087	1
Counter Present value	CN	WORD, UINT, INT	0-23087	16
Counter Contact	CS	BOOL	0-23087	1
Data register	D	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	0-25983	16
Annunciator	F	BOOL	0-32687	1
Latch relay	L	BOOL	0-32687	1
Internal relay	M	BOOL	0-32767	1
File register	R	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	0-32767	16
Link special relay	SB	BOOL	0-2047	1
Special register	SD	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	0-2047	16
Special relay	SM	BOOL	0-2047	1
Retentive Timer Coil	SC	BOOL	0-32767	1
Retentive Timer Present value	SN	WORD, UINT, INT	0-32767	16
Retentive Timer Contact	SS	BOOL	0-32767	1
Link special register	SW	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	0-2047	16
Timer Coil	TC	BOOL	0-23087	1
Timer Present value	TN	WORD, UINT, INT	0-23087	16
Timer Contact	TS	BOOL	0-23087	1
Index register	V	WORD, UINT, INT	0-6	16
Link register	W	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	0-657F	16
Input relay	X	BOOL	0-8191	1
Output relay	Y	BOOL	0-8191	1
Index register	Z	WORD, UINT, INT	0-15	16
Step relay	S	BOOL	0-8191	1
Function input	FX	BOOL	0-15	1
Function output	FY	BOOL	0-15	1
Function register	FD	WORD, UINT, INT	0-4	16
Extended File register	ER	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	0-1042431	16

Device	Device	Data Type	Device Range	Size Bits
File register	ZR	DWORD, WORD, UDINT, UINT, DINT, INT, REAL, STRING	0-1042432	16
Accumulator	A	DWORD, WORD, UDINT, UINT, DINT, INT, REAL	0-1	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

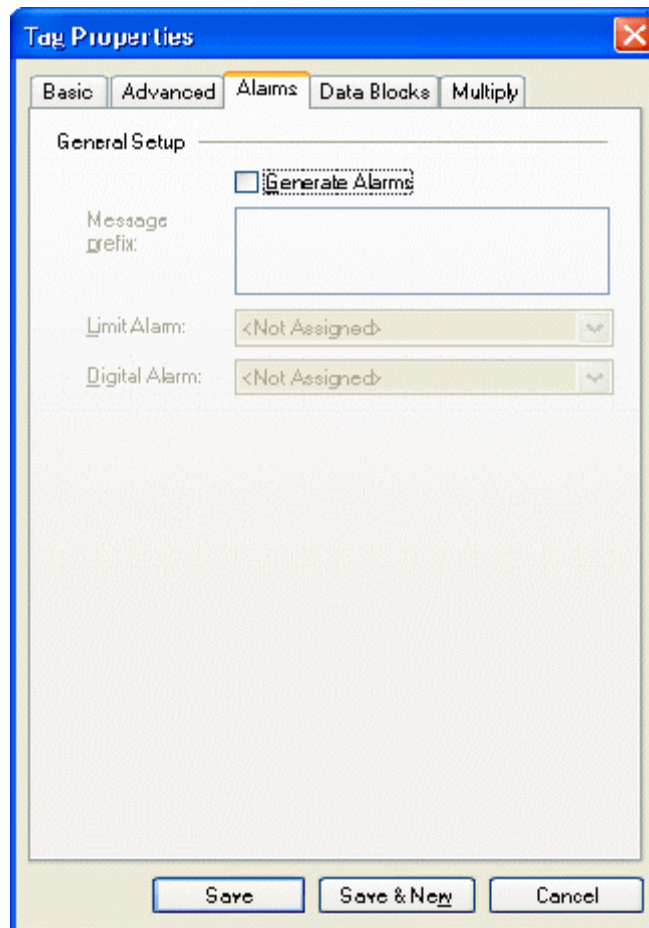
To test the client functionality, check the **Simulate** check box and then choose a **Simulation Signal** from the drop-down list.

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.

3.4.3 MX Data 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.

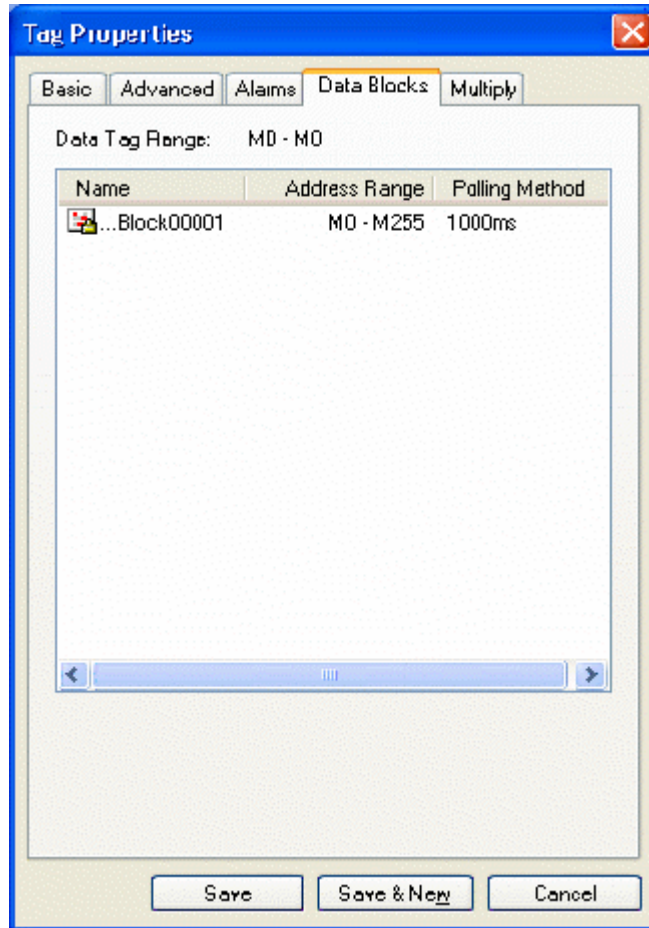
- **Digital Alarm:** Select a Digital Alarm Definition from the drop-down list.
- **Limit Alarm:** Select a Limit Alarm Definition from the drop-down list.
- **Save:** Saves all changes specified in the properties dialog box. The data tag appears in the list view for the selected device.
- **Save & New:**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

3.4.4 MX Data Tag Properties: Associated Data Blocks

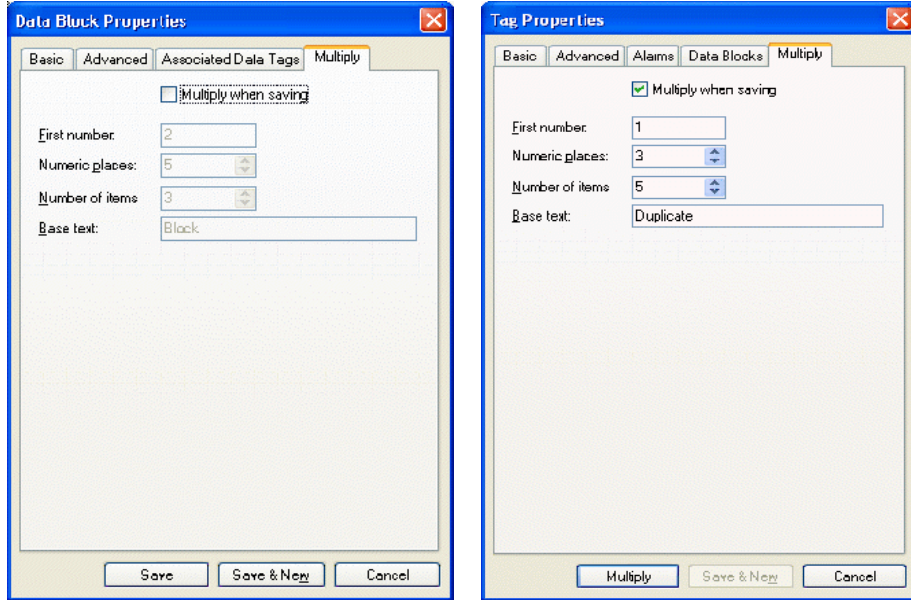
The **Data Blocks** tab of the **Tag Properties** dialog box, shown below, lists all data blocks that are tied to each data tag. It also lists the **Address Range** and the selected **Polling Method** for each data block.



MX Data Tag Properties: Data Blocks

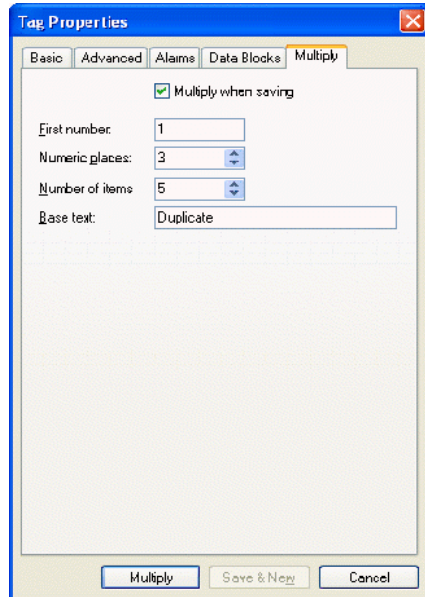
3.4.4.1 Multiply pages on Tag and Data Block dialogs

This feature is available from the MX Tag and Data Block 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 and Data Block property sheets.

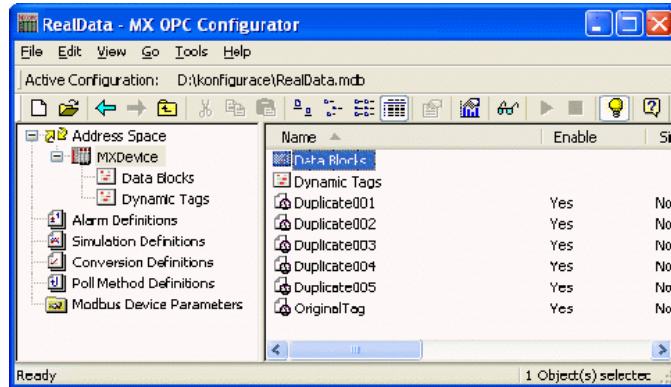


Example

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.



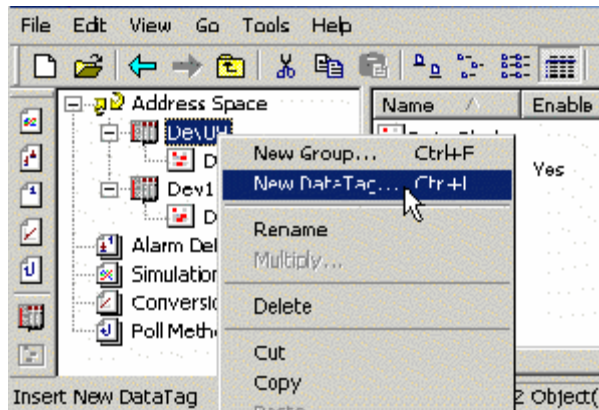
The **Multiply when saving** checkbox changes the text on the **Save** button to “Multiply” and disables the **Save & New** button. When the user now clicks the **Multiply** button, the tag is saved and multiplied.

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 semantics of the items on this property page and the multiply behavior is the same as in previous versions of MX OPC Configurator.

3.4.5 Adding a New MX Data Tag

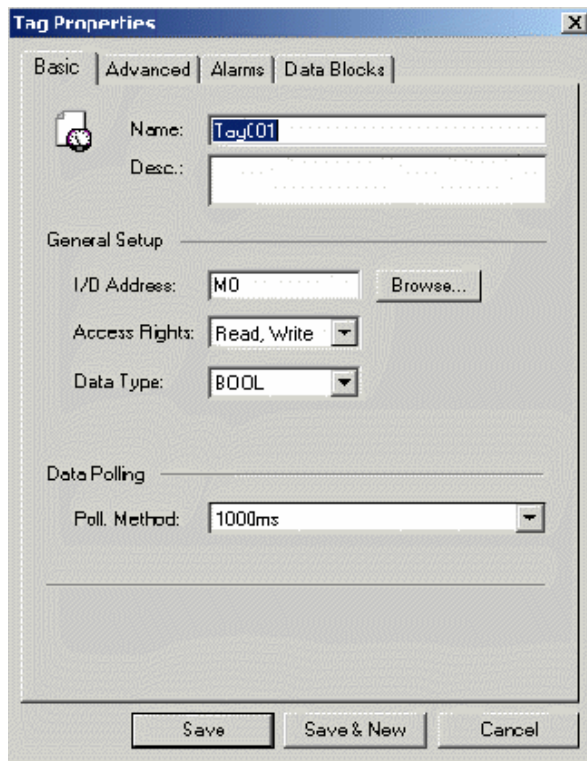
To add a new data tag:

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

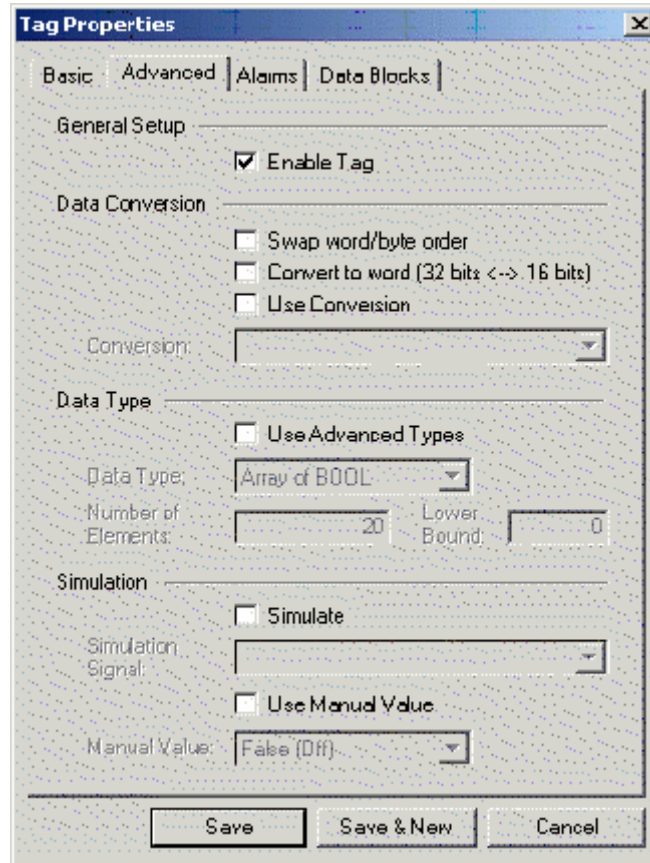
2. The **Basic** tab of the **Data Tag Properties** dialog box appears, as shown in the figure below.



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



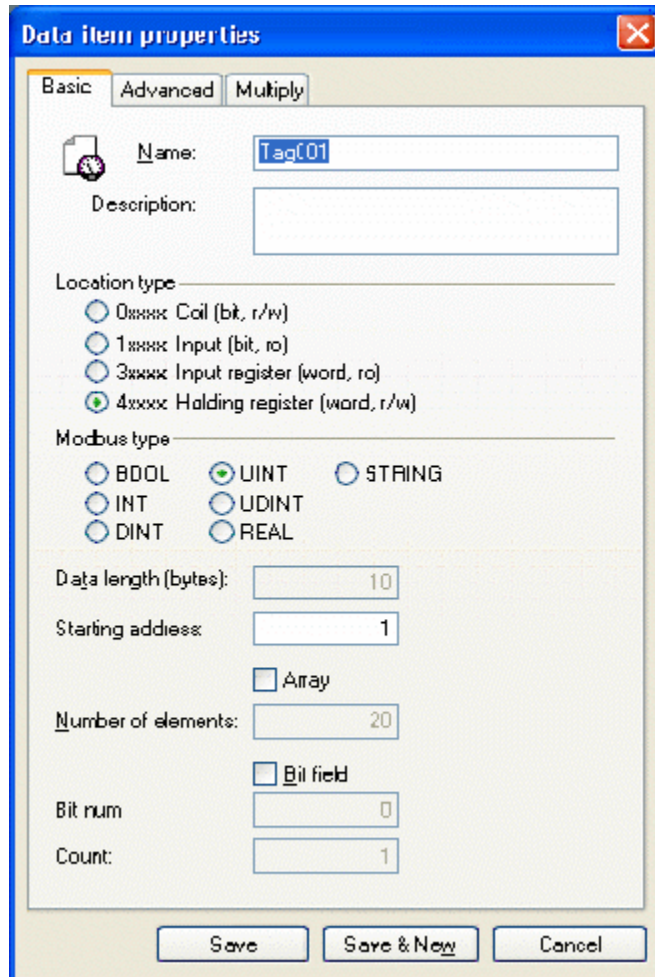
MX Data Tag Properties: Advanced Tab

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.

Note: By default, the Configurator will automatically arrange your data tags into blocks so that they can be read from the device in the most efficient way. If you would prefer to define the data blocks manually, you can do this by selecting **Options** from the **Tools** menu and checking the **Advanced Mode (Show Data Blocks)** check box on the **General** tab of the **Options** dialog box.

3.4.6 Basic Modbus Data Tag Properties

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



- **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 40 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:

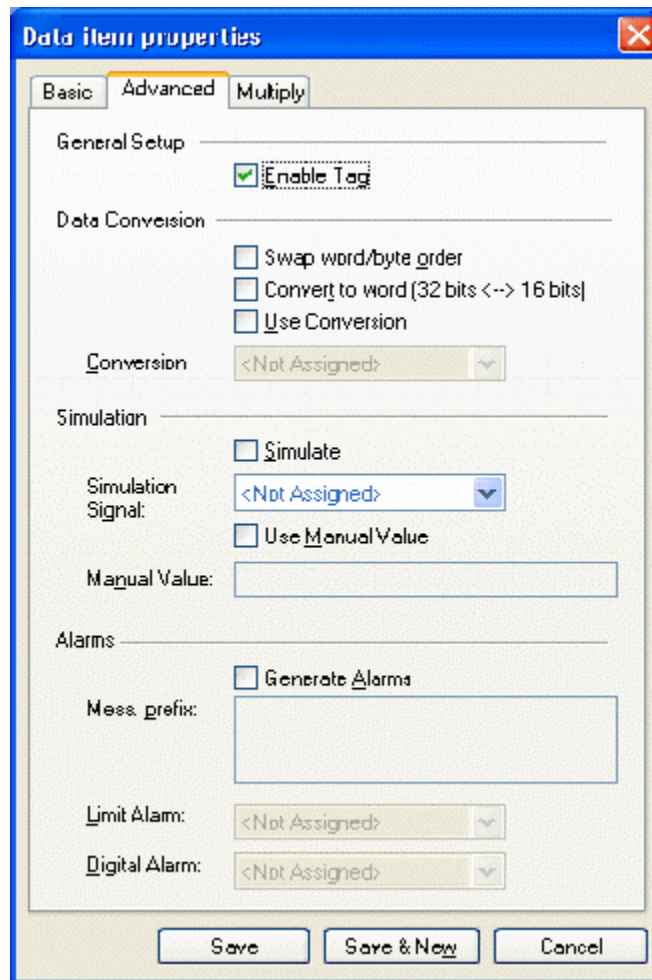
	Read Only	Read / Write
1 bit	Input	Coil
16 bit	Input Register	Holding Register

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.

3.4.7 Advanced Modbus Data Tag Properties

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



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.

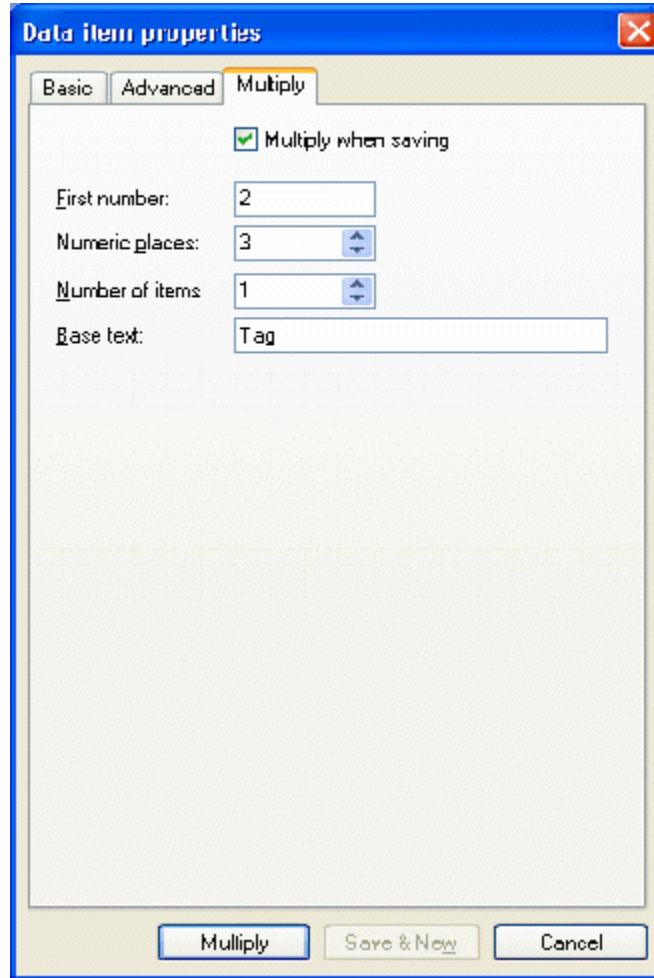
Alarms

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.

3.4.8 Multiply pages on Modbus Tags

This feature is available from the Modbus Tag and Data Block 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.

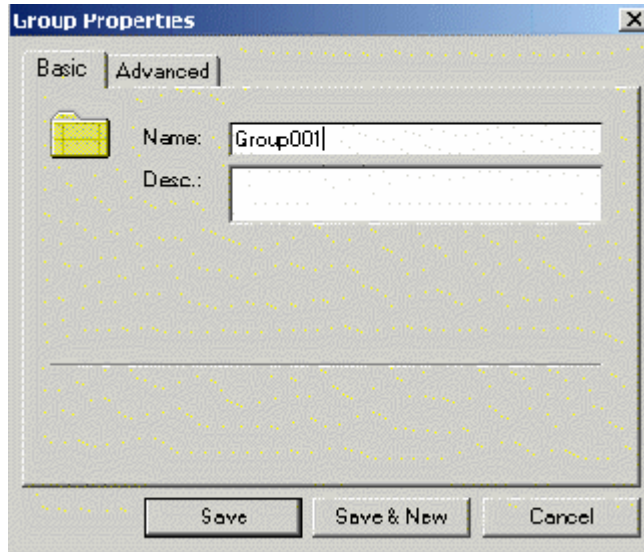
All the items on this property page (except the **Multiply when saving** checkbox) are enabled if and only if the **Multiply when saving** checkbox is checked. The semantics of the items on this property page and the multiply behavior is the same as in previous versions of MX OPC Configurator.

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

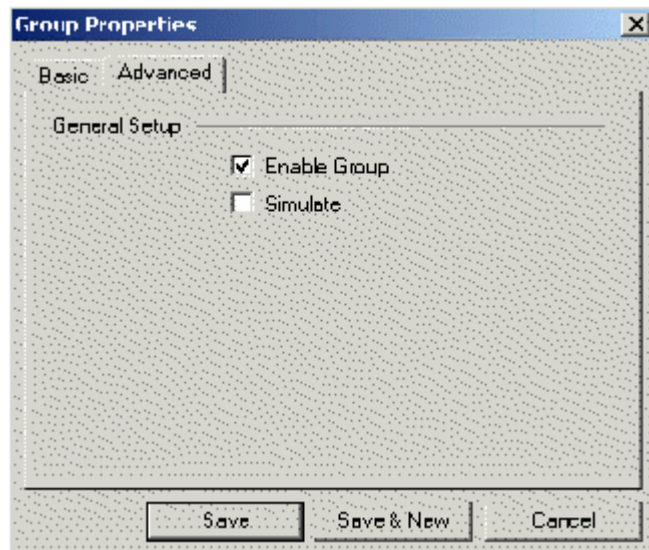
In the **Basic** tab of the **Group Properties** dialog box, shown below, type a name for the group and a description for the group (optional).



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.

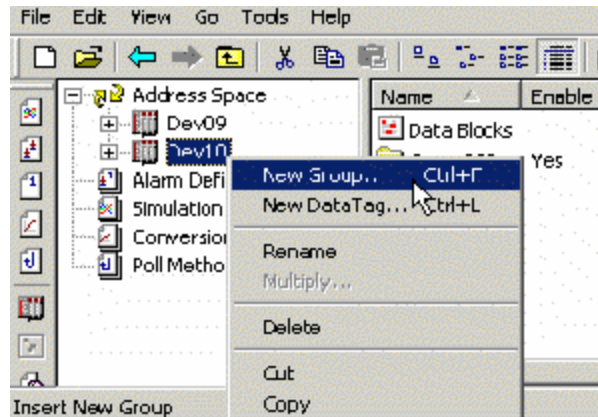


Group Properties: Advanced Tab

3.5.3 Adding a New Group

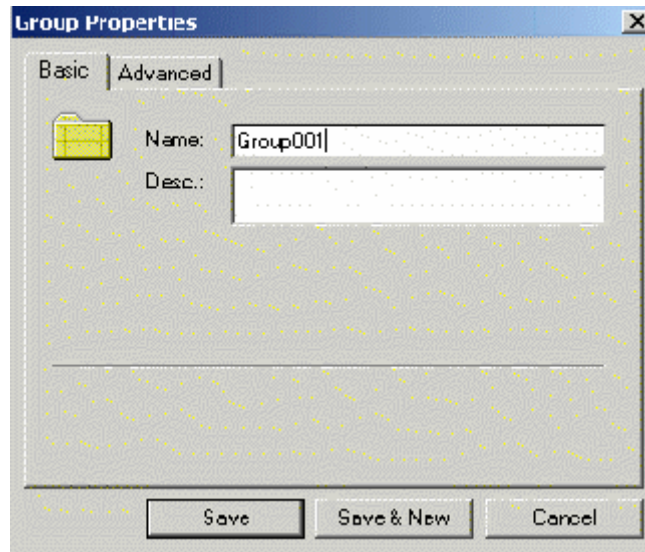
To add a new group:

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.



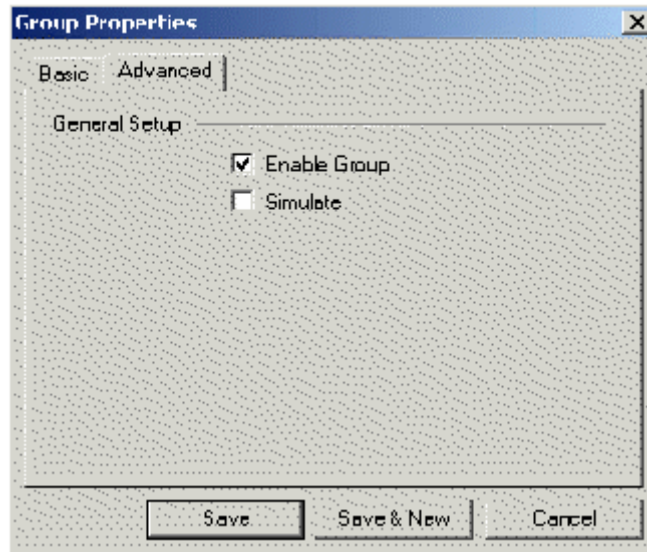
Adding a New Group

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

Alarm Definitions

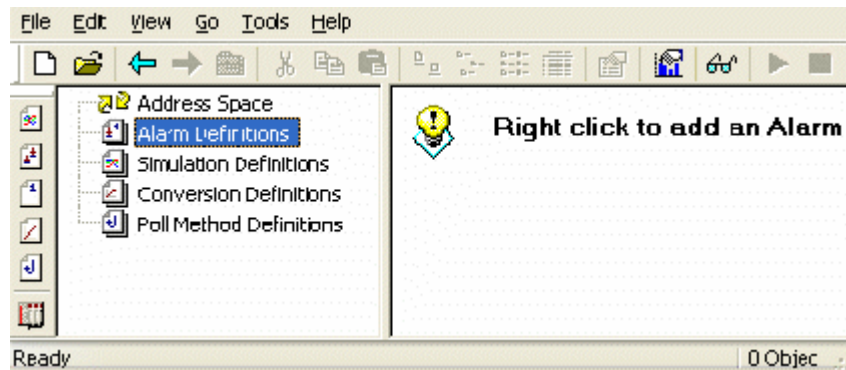
4

4.1 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

4.2 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.

4.2.1 Basic Limit Alarm Properties

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

The screenshot shows the 'Limit Alarm Properties' dialog box with the 'Basic' tab selected. The 'Name' field contains 'LimAlarm002'. Below it is a table for 'Limit Alarm Setup' with the following data:

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"/>

At the bottom of the dialog are three buttons: 'Save', 'Save & New', and 'Cancel'.

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.

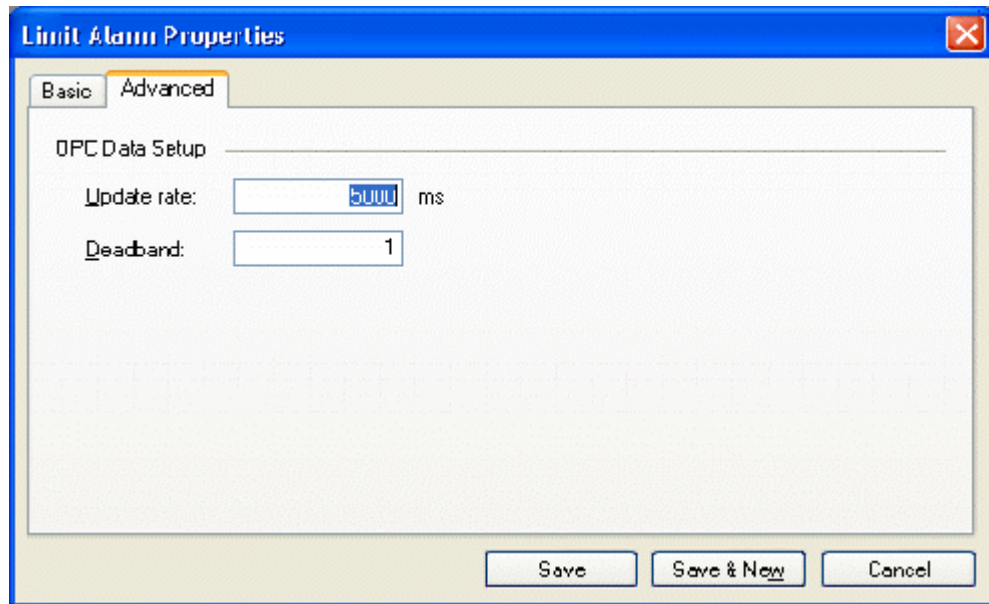
- **Severity** is the OPC-defined value for alarm Priority. The valid OPC severity range is 0 (lowest) to 1000 (highest).
- The **Requires Ack** field is used for OPC condition alarms to determine whether the alarm needs a user acknowledgement. If the **Requires Ack** field is checked, then the alarm 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.

4.2.2 Advanced Limit Alarm Properties

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



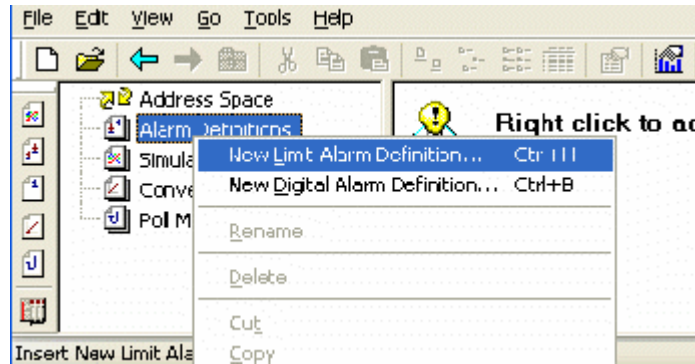
Limit Alarm Properties: Advanced Tab

- **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.

4.2.3 Creating a New Limit Alarm Definition

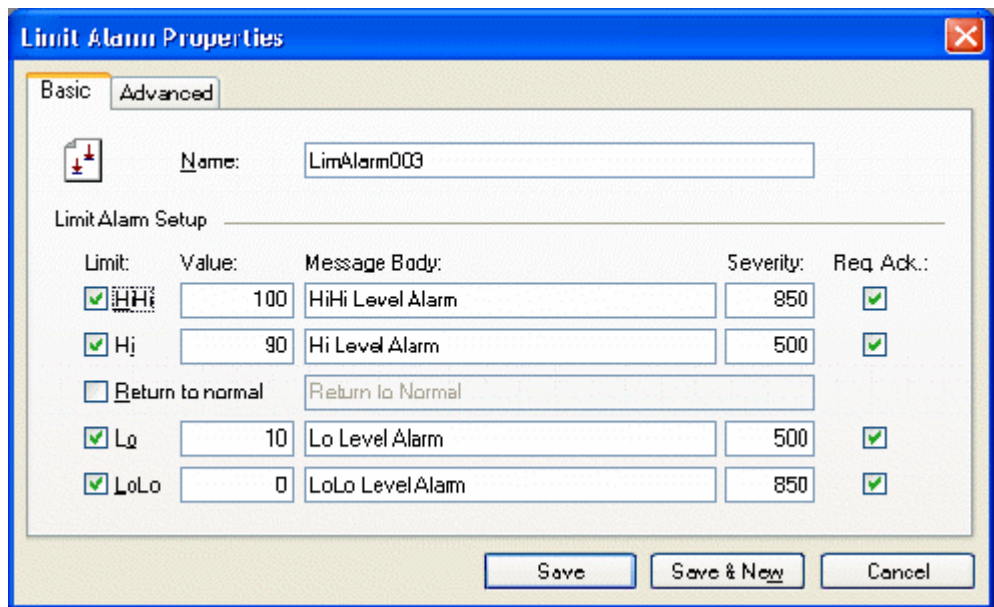
To create a new limit alarm definition:

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.



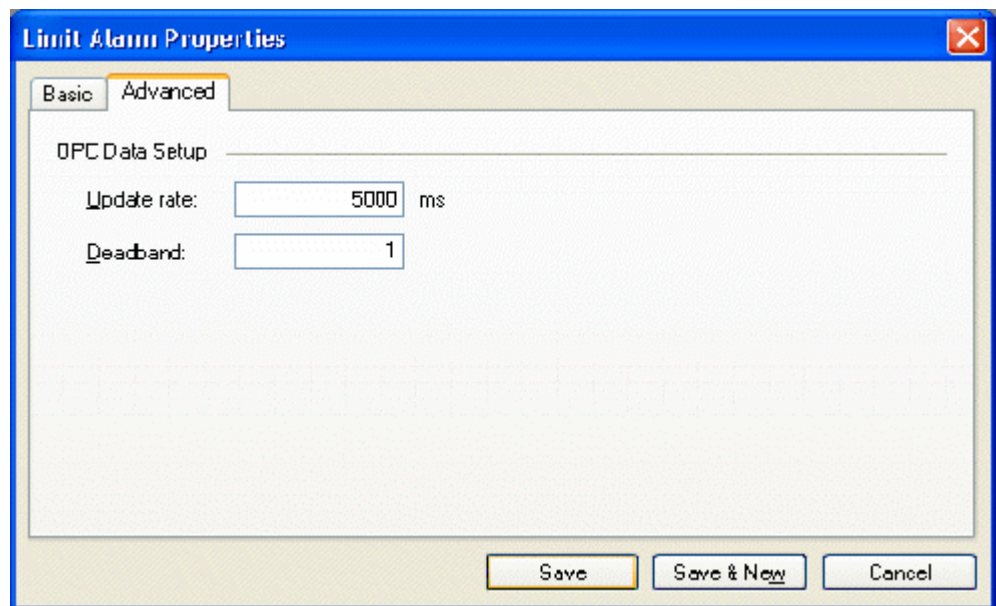
Creating a New Limit Alarm Definition

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.

4.3 Digital Alarm Definitions

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

4.3.1 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.

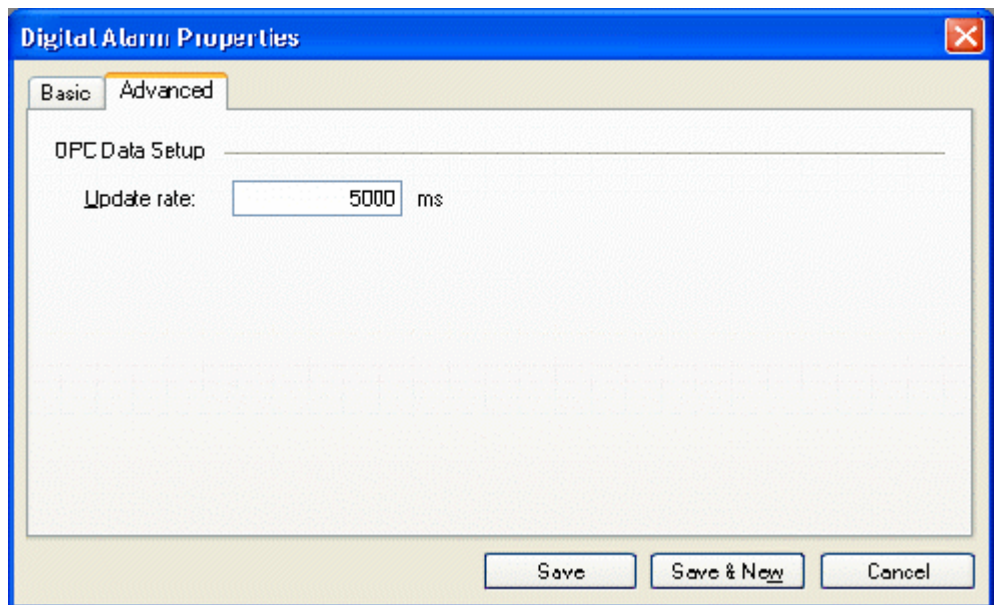
Digital Alarm Properties: Basic Tab

- **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.

4.3.2 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.

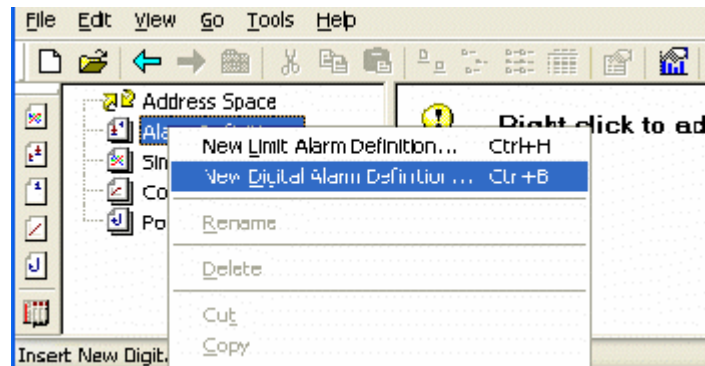


Digital Alarm Properties: Advanced Tab

4.3.3 Creating a New Digital Alarm Definition

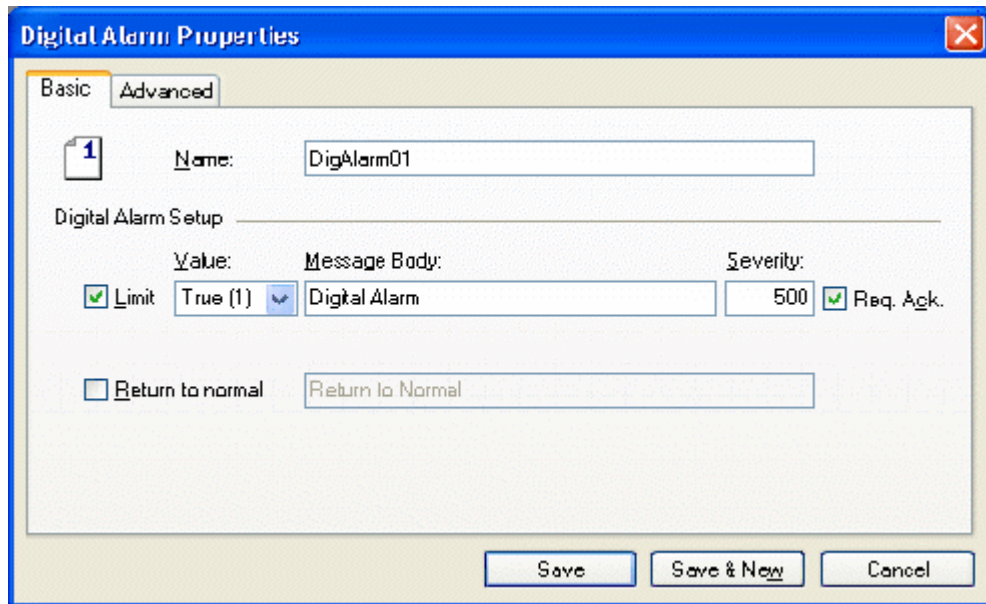
To create a new digital alarm definition:

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.



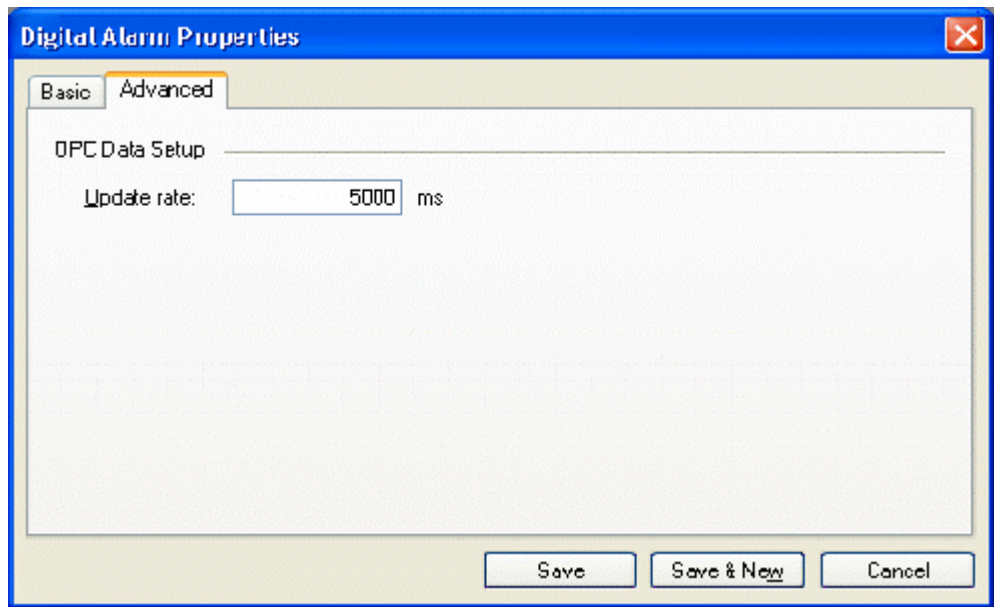
Creating a New Digital Alarm Definition

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



Digital Alarm Properties: Basic Tab

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



Digital Alarm Properties: Advanced Tab

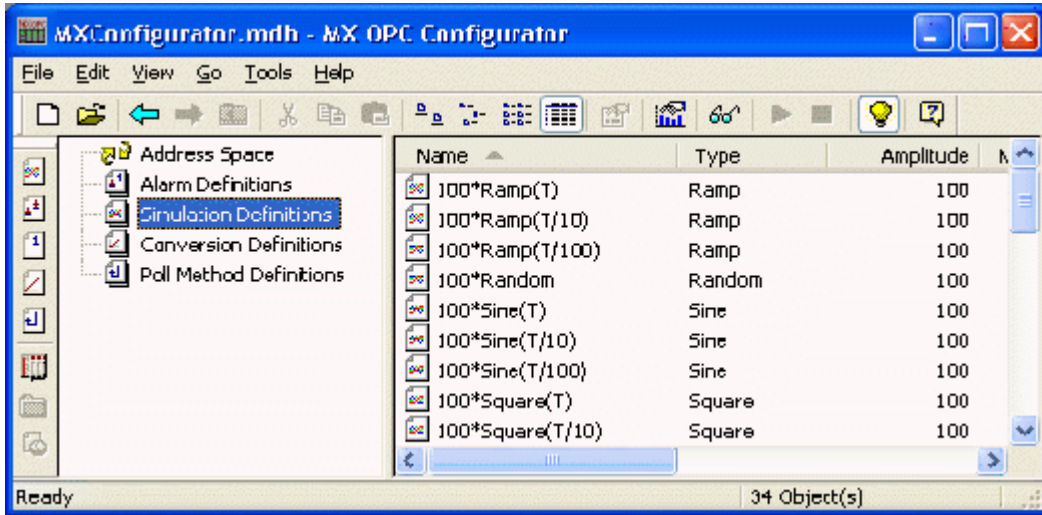
6. When you have finished configuring the alarm definition properties, click the **Save** button. The new configuration appears under the **Alarm Definitions** tree control.

Simulation Signals

5

5.1 Configuring Simulation Signals

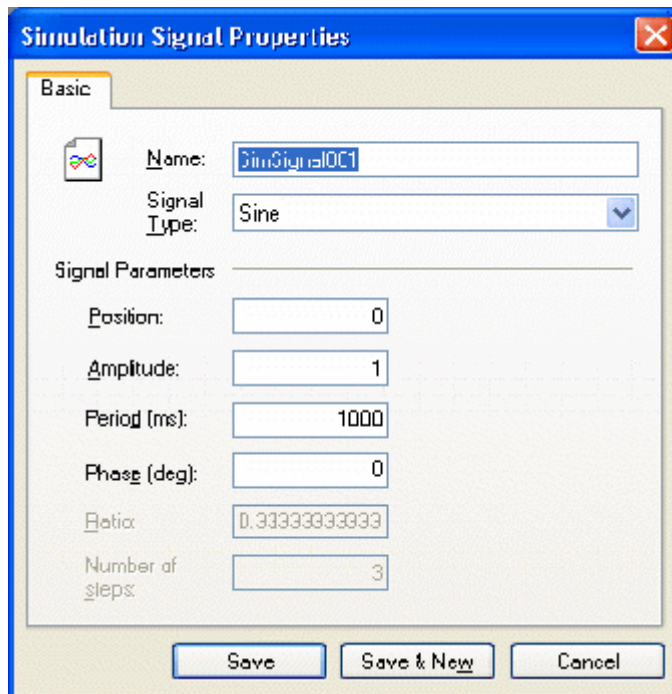
The Configurator offers a wide range of OPC data items in the **Simulation Definitions** tree control, as shown in the figure below.



Simulation Definitions Tree Control

5.1.1 Basic Simulation Signal Properties

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



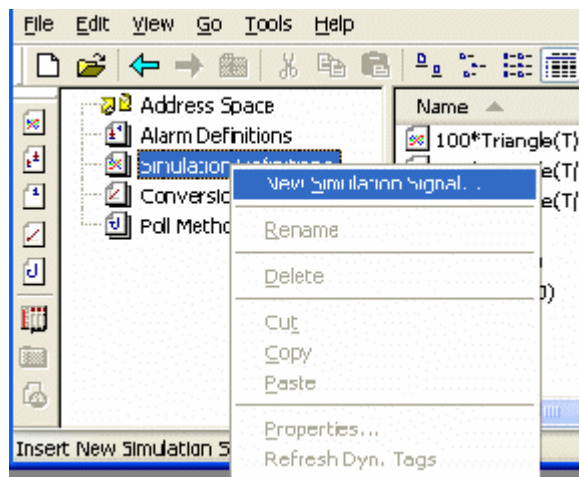
Simulation Signal Properties: Basic Tab

- **Name:** Specifies the name of the selected simulation signal. The name can be up to 12 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 every time when the item is read.
 - * **Write Count** increments when the item 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.

5.1.2 Creating a New Simulation Signal

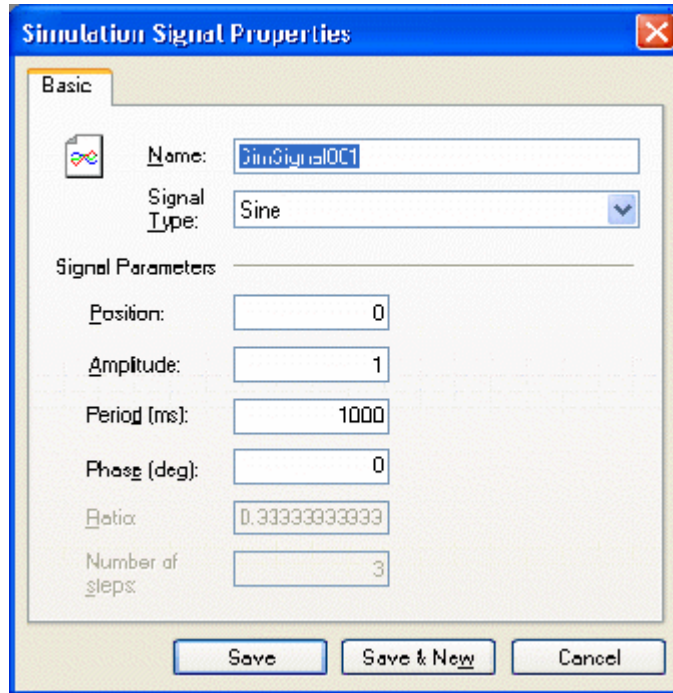
To create a new simulation signal:

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.

Runtime Operations

6

6.1 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, data blocks, 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 blocks:** A data block is an addressable portion of a device. The data block contains the properties and methods that manage the behavior of a data block. Data blocks in the server's local memory correspond to data areas in the MX Runtime. When you add new data blocks to the server's local memory, you also add new data areas to the MX Runtime. Data blocks are not visible to the OPC client.

Note: Data block configuration is optional (for advanced users).

- **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).

6.2 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.

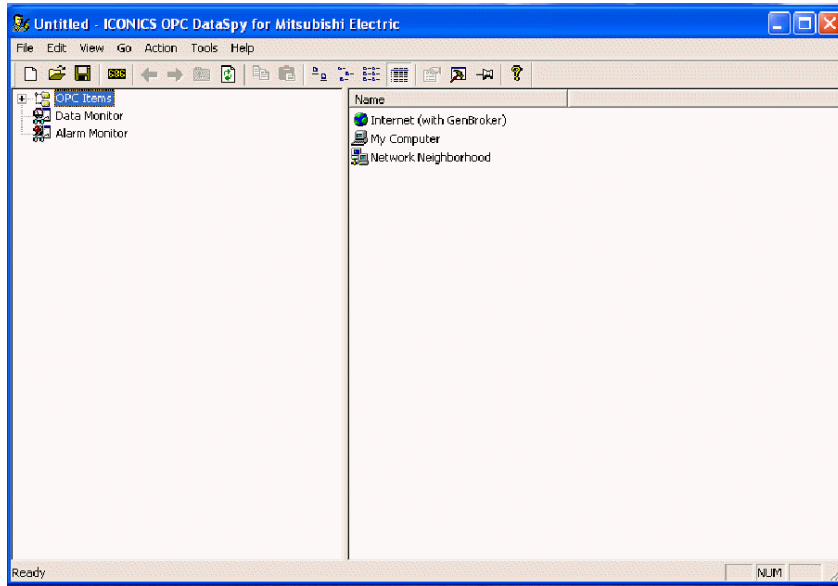
The OPC DataSpy user interface consists of the following elements:

- 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 **Programs > Melsoft Application > MX OPC Server 5.00 > 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).

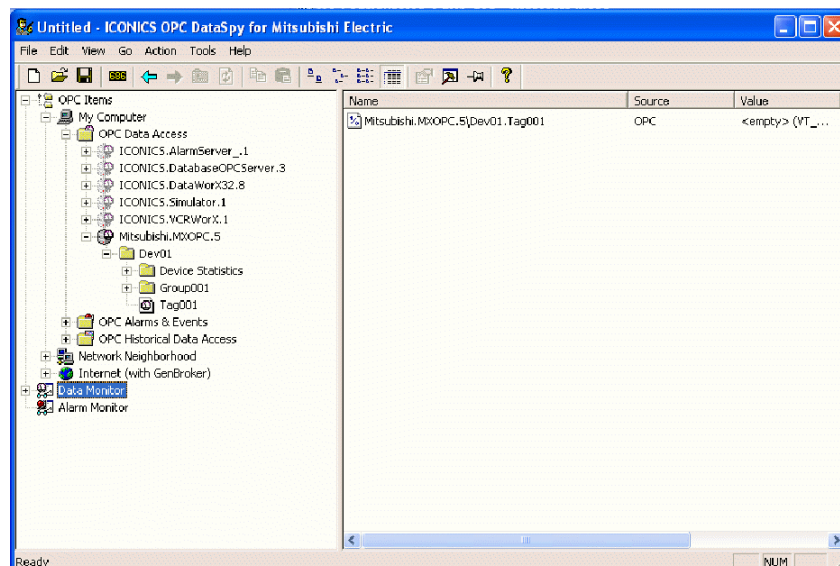


OPC DataSpy Screen

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.5**” section.

Users will then see all configured devices listed underneath “**Mitsubishi.MXOPC.5**” and can browse groups and data tags accordingly.

3. Right click on any selected tag listed beneath “**Mitsubishi.MXOPC.5**” 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.



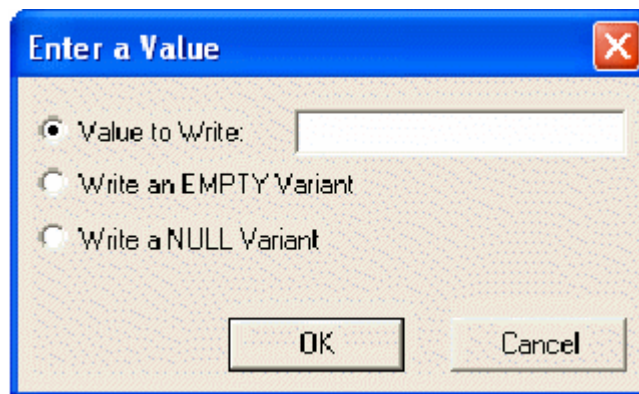
4. If your client application does not support reading single array elements or reading a single bit within a word, you may still be able to use these features by changing the name of the item that you add.

A client can address a bit within a word tag using the item syntax "tagname.bitNr" (e.g. tagname.0 for bit 0).

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

5. 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 (,).

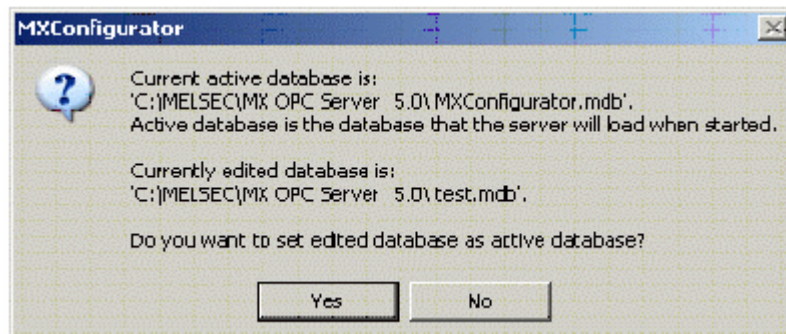


6.3 Activating the Database

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.

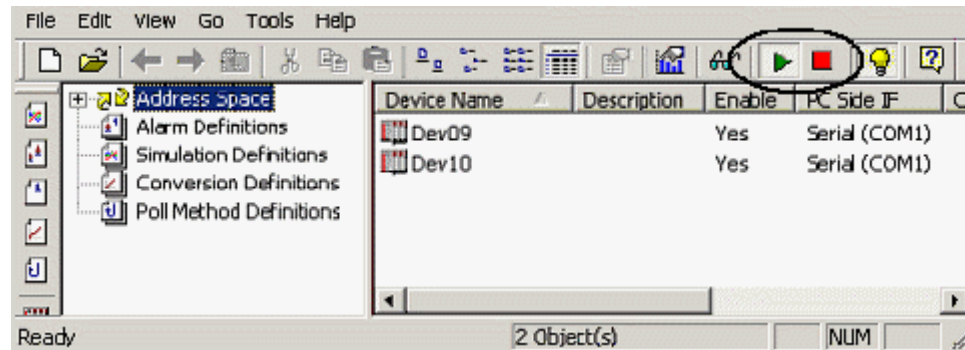


Activating the Database

6.4 Starting and Stopping the Driver

The I/O Server maintains the driver's device, data blocks, 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

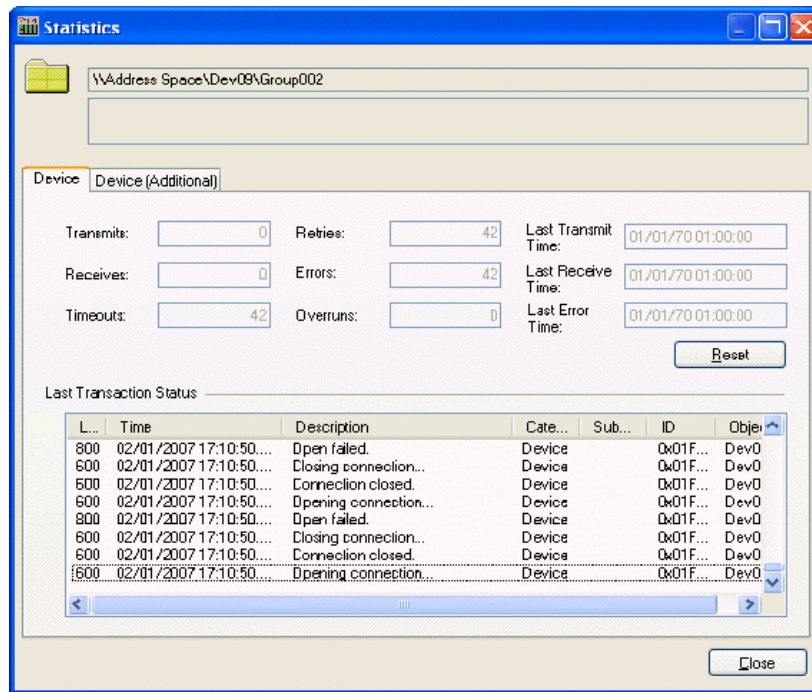
6.5 Statistics Mode

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 blocks
- Data tags

To change to statistics mode:

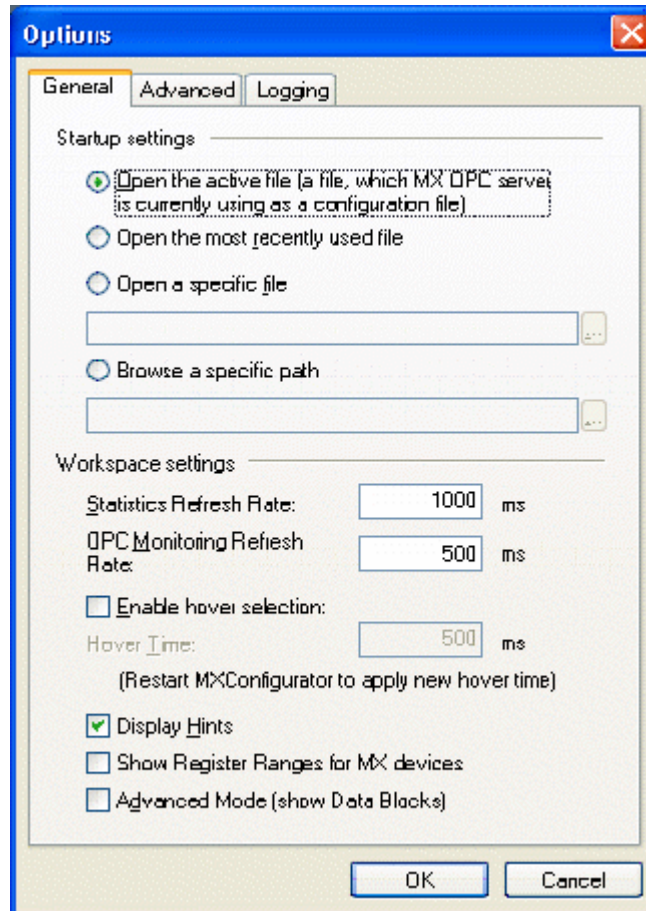
1. Select **Statistics** from the **View** menu.
2. The Configurator switches the current dialog view to data statistics mode, as shown in the figure below.



Statistics Mode View

6.5.1 Setting the Statistics Mode Refresh Rate

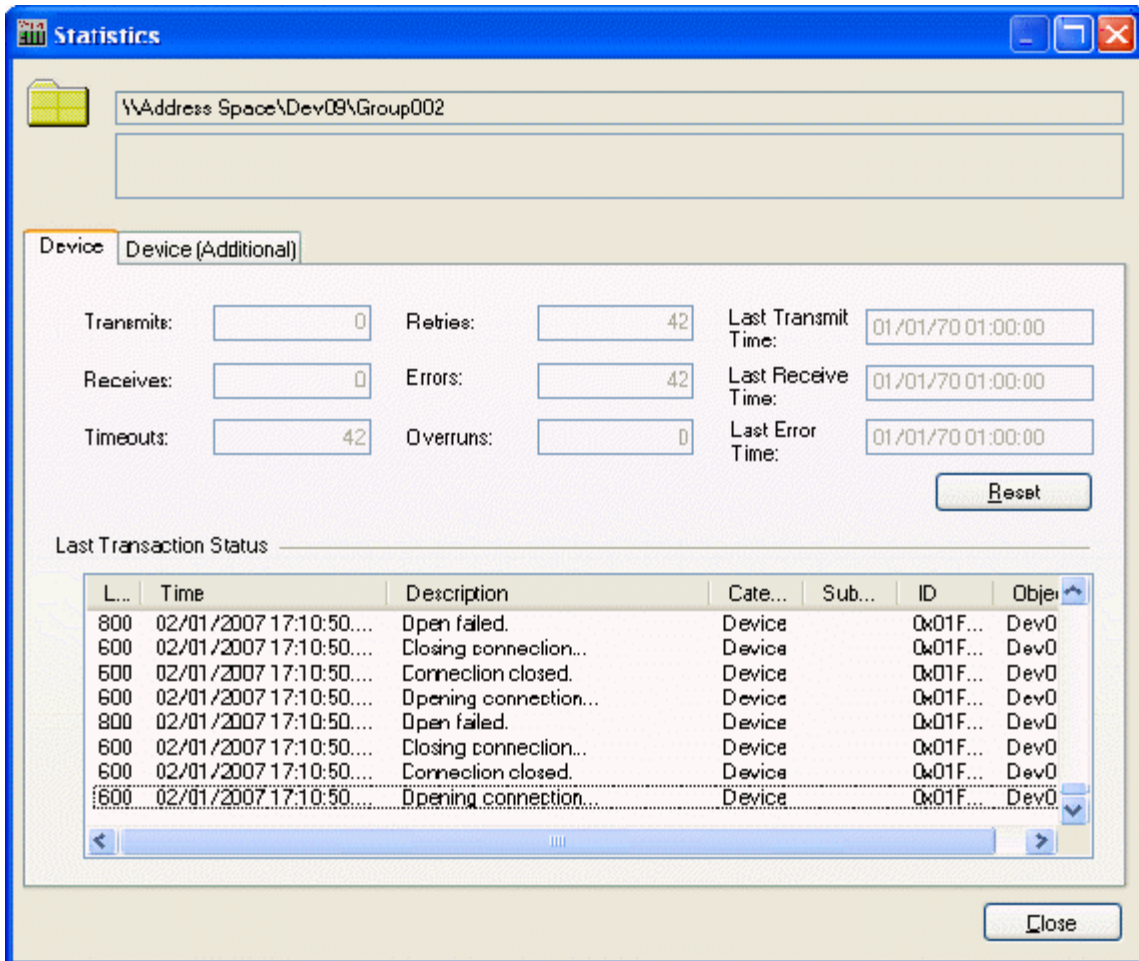
To set the Statistics Mode **Refresh Rate** (in milliseconds), select **Options** from the **Tools** menu and click on the **General** tab of the **Options** dialog box, as shown in the figure below.



Options Dialog Box: General Tab

6.5.2 Device Statistics

To view device statistics in statistics mode, click on a device 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.

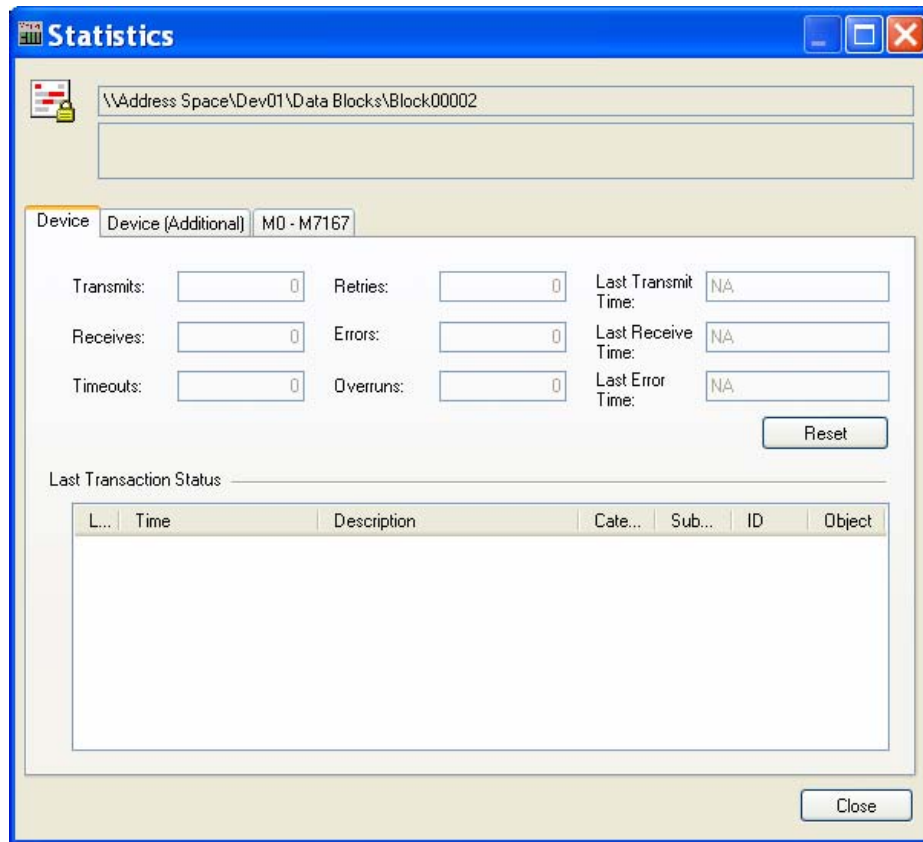
Last Receive Time: The last time data was successfully received from the device.

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

Reset: Resets all data statistics.

6.5.3 Data Block Statistics

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



Viewing Runtime Data Statistics for Data Blocks

Transmits: Displays the number of messages sent to the process hardware from the selected data block.

Receives: Displays the total number of messages the data block 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 data block 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 data block. 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 data block. 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 are not updating at the specified poll rate.

Last Read Time: Displays the last time and date that the I/O driver successfully read data from the selected data block.

Last Write Time: Displays the last time and date that the driver successfully wrote data to the selected data block.

Last Error Time: Displays the last error received by the data block from the process hardware. Errors displayed in this field are specific to the hardware.

Data Quality: Displays the OPC quality of data in the data block.

Last Transaction Status: Displays the last transaction status messages for the selected data block.

Reset: Resets all data statistics.

6.5.4 Data Tag Statistics

To view data tag statistics in statistics mode, click on a data tag under the **Address Space** tree control. This opens the **Statistics** dialog box, which shows statistics for the tag's supported data blocks. Note that statistics are shown only for the tag's supported data blocks defined in the Configurator. No data block statistics are shown for Modbus tags.

Transmits: Displays the number of messages sent to the process hardware from the selected data block.

Receives: Displays the total number of messages the data block 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 data block 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 data block. 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 data block. 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 are not updating at the specified poll rate.

Last Read Time: Displays the last time and date that the I/O driver successfully read data from the selected data block.

Last Write Time: Displays the last time and date that the driver successfully wrote data to the selected data block.

Last Error Time: Displays the last error received by the data block from the process hardware. Errors displayed in this field are specific to the hardware.

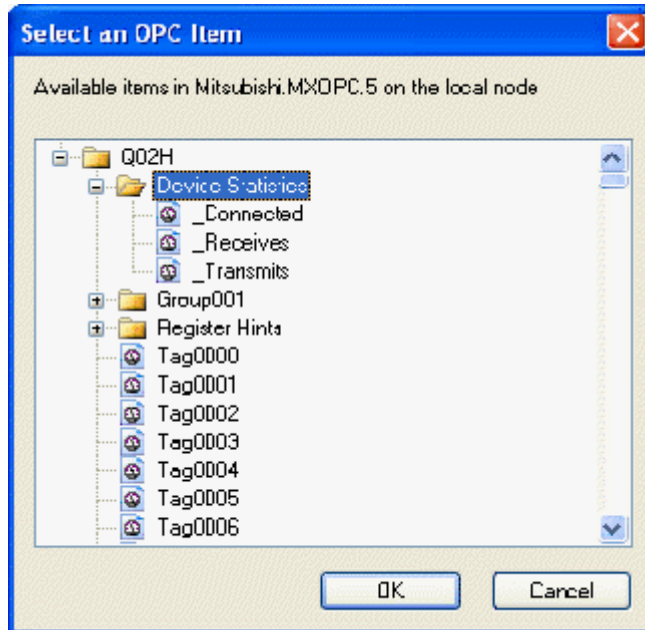
Data Quality: Displays the OPC quality of data in the data block.

Last Transaction Status: Displays the last transaction status messages for the selected data block.

Reset: Resets all data statistics.

6.5.5 Statistical Tags in OPC Address Space

Similar to the “Hints”, the MXRuntime exposes three different tags in a group called “Device Statistics”, as seen on the next screenshot:



These tags are exposed always and can not be switched off in the configurator. The meaning of the tags is as follows:

- **Connected:** Returns a number indicating whether the runtime is connected to the device or not. 1 means connected, 0 not connected.
- **Receives:** Number of receives. The same as the statistical counter “Receives” in the statistics dialog.
- **Transmits:** Number of transmits. The same as the statistical counter “Transmits” in the statistics dialog.

6.5.6 Device Statistics (Additional)

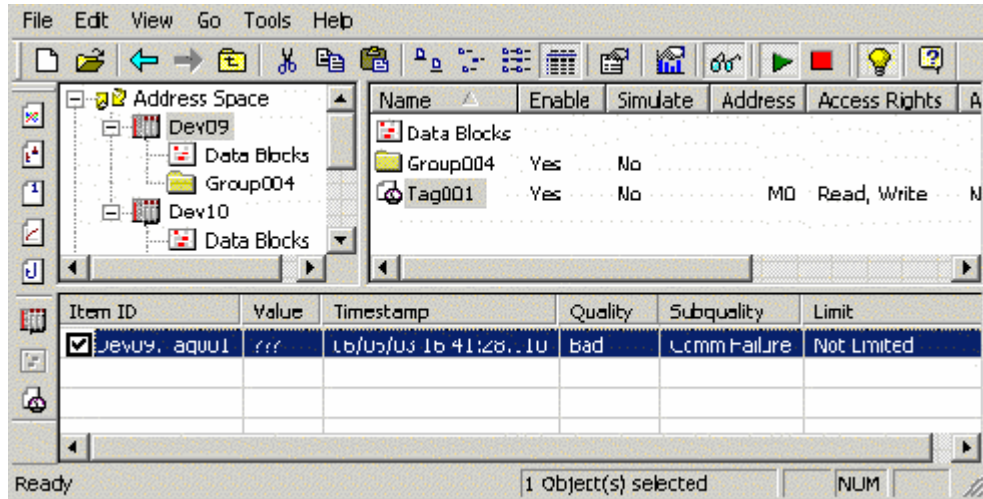
- **Consecutive Faults** – Displays the total number of faults without an interposing success on this device.
- **Add Point Calls** – The total number of points added during the lifetime of this device. Note that the device lifetime is not the same as the running time of the server, as the server may 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.
- **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.

6.6 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.

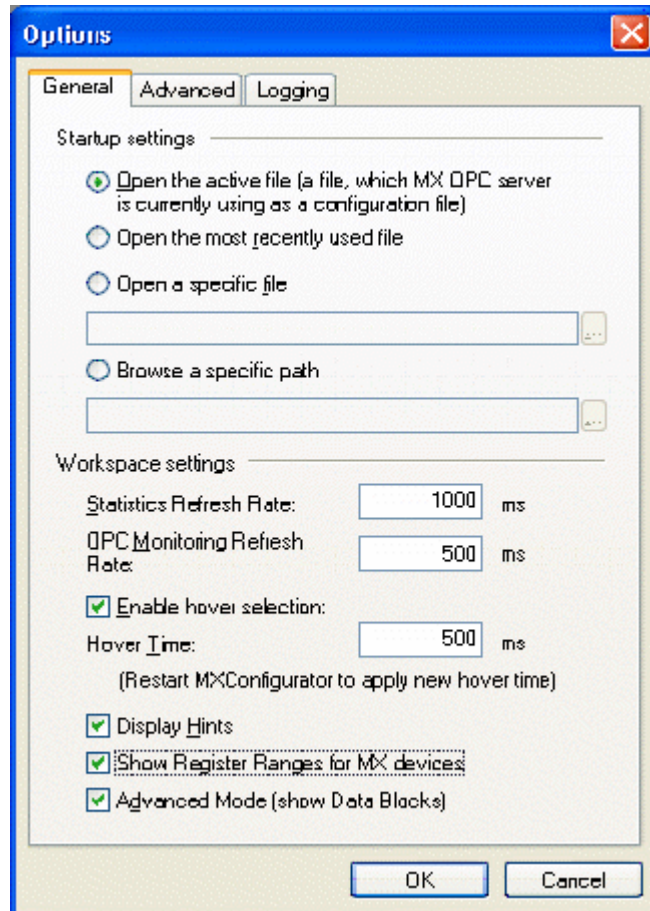
The MX OPC Configurator includes a runtime monitor for viewing server data. To change to the monitor view, select **Monitor View** from the **View** menu. The runtime monitor appears in the bottom pane of the Configurator, as shown in the figure below. During runtime, the monitor scans the server and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



Runtime Monitor View

6.6.1 Setting the Monitor View Refresh Rate

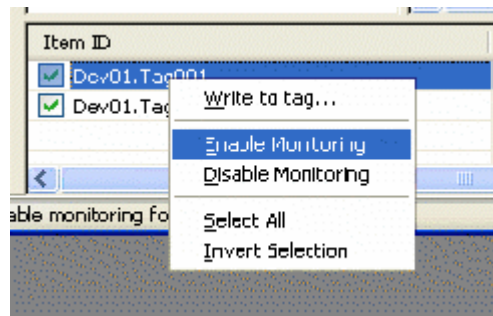
To set the **OPC Monitoring Refresh Rate** (in milliseconds), select **Options** from the **Tools** menu and click on the **General** tab of the **Options** dialog box, as shown in the figure below.



Options Dialog Box: General Tab

6.6.2 Enable Monitoring

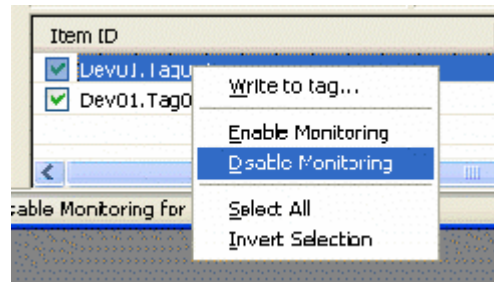
Monitoring is enabled for each item with a check mark next to it. To enable/disable monitoring for an item, you can click on the box to the left of the item. A check mark inside the box means the item is enabled for monitoring. If there is no check mark, then the item is disabled. To enable monitoring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

6.6.3 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

6.7 Writing to Tags

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

The MX OPC Server also includes a bit selection within a tag using the dot notation. A client can address a bit within a tag using the syntax "tagname.bitNr" (e.g. tagname.0 for bit 0).

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

In the "Bit access" feature, if the item is of a type other than BOOL or STRING, you can access its single bits. They are implemented as custom "OPC Properties," and also can be accessed that way.

Examples:

TagD0

type: WORD

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

'TagD0.0', 'TagD0.0', ..., 'TagD0.16'

TagD2

type: REAL

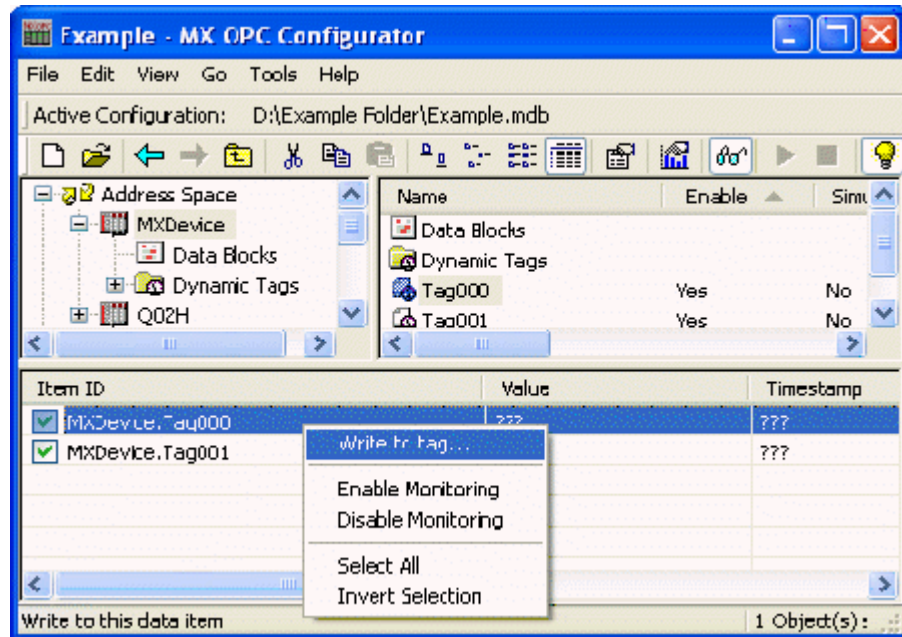
You can access 'TagD2', but also its properties:

'TagD2.0', 'TagD2.0', ..., 'TagD2.32'

Writing to Tags

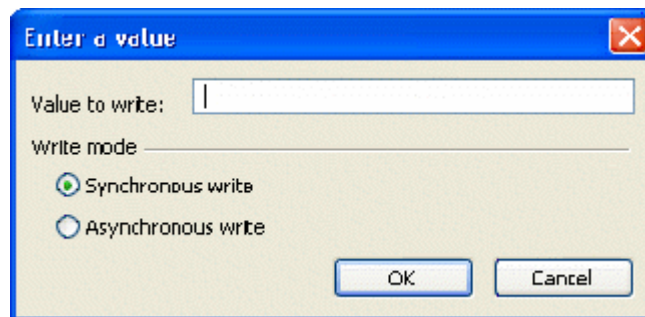
Writing to Tags

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.

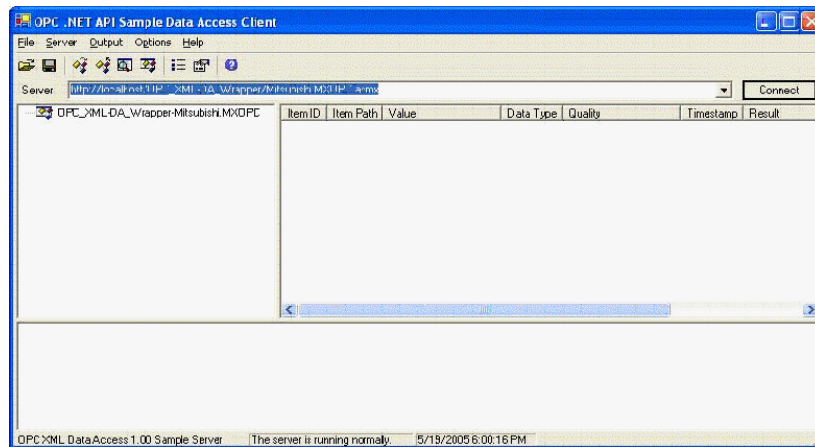
When any of the previous formatting requirements are not met, a dialog box appears showing the error, as in the example below.



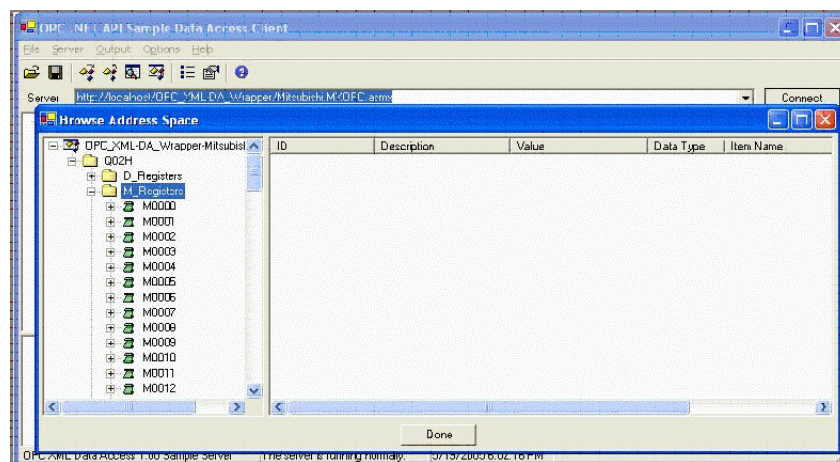
6.8 Connecting to MX OPC Server via XML Data Access Browsing

As an alternative to DCOM communications, the MX OPC Server 5.0 supports the OPC XML Data Access (DA) specification (version 1.01). XML-DA is a general way to access and browse OPC servers instead of via DCOM. This means you can go beyond the Microsoft network scope and access the server from anywhere on the Internet. In order for the XML-DA connection to work, the client that connects to the MX OPC Server must support XML-DA. The following example shows how a client can connect remotely to the MX OPC Server, browse for and select data items (tags) in the server, and monitor the selected data items in runtime.

The figure below shows how the sample XML-DA client connects to the MX OPC Server. The full path of the MX OPC server address is entered in the **Server** field, and then a connection to the server is established by clicking the **Connect** button. You can test the connection to the server by entering the URL address of the server in any Web browser.



Once a connection to the MX OPC Server is established, the data items in the Address Space of the server's configuration database are exposed to the client. This enables you to browse for data items (tags) in runtime, as shown in the figure below.



Browsing the MX OPC Server Address Space

The selected data items are listed in the **Item ID** field of the client, as shown in the figure below. This allows you to remotely monitor the MX OPC Server data in runtime. The data item values are updated in real time in the sample client.

Item ID	Item Path	Value	Data Type	Quality	Timestamp	Result
Q02H.D_Registers.D0000		60154	UInt16	good	2005-05-19 18:07:01.968	S_OK
Q02H.D_Registers.D0001		60154	UInt16	good	2005-05-19 18:07:01.968	S_OK
Q02H.D_Registers.D0002		60154	UInt16	good	2005-05-19 18:07:01.968	S_OK
Q02H.D_Registers.D0003		60154	UInt16	good	2005-05-19 18:07:01.968	S_OK
Q02H.M_Registers.M0000		False	Boolean	good	2005-05-19 18:07:01.984	S_OK
Q02H.M_Registers.M0001		False	Boolean	good	2005-05-19 18:07:01.984	S_OK
Q02H.M_Registers.M0002		False	Boolean	good	2005-05-19 18:07:01.984	S_OK
Q02H.TN_registers.TN0000		8	UInt16	good	2005-05-19 18:07:01.937	S_OK
Q02H.TN_registers.TN0001		0	UInt16	good	2005-05-19 18:07:01.937	S_OK
Q02H.TN_registers.TN0002		0	UInt16	good	2005-05-19 18:07:01.937	S_OK

Monitoring MX OPC Server Data in Runtime

OPC XML-DA Installation

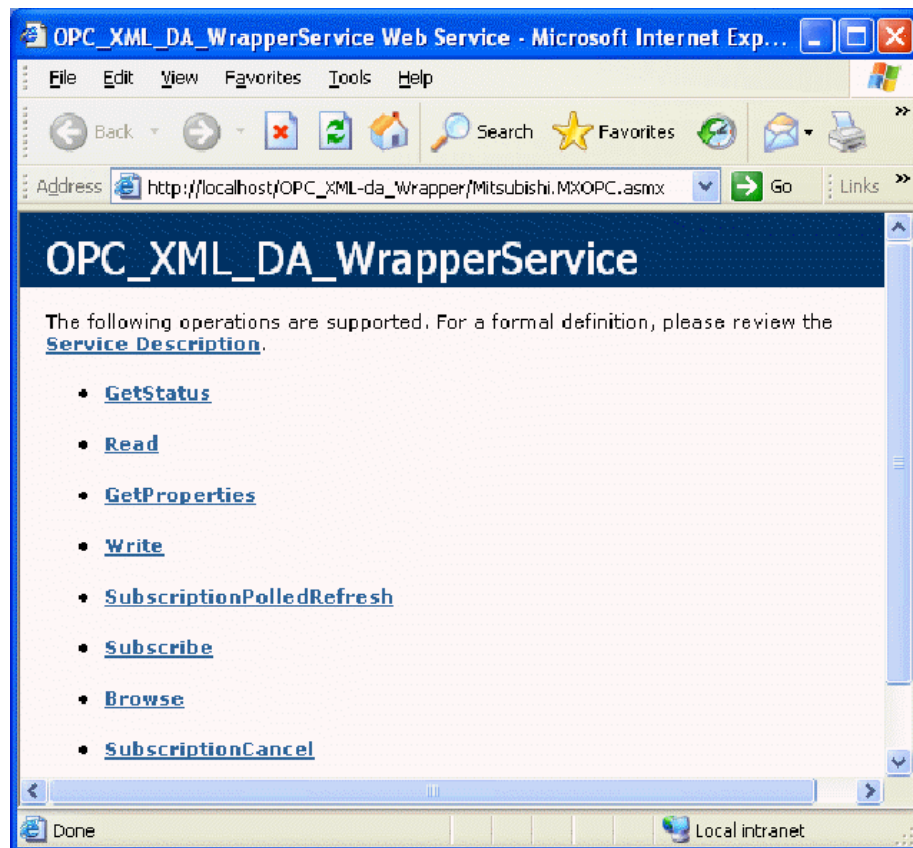
Prerequisites: The Wrapper requires the Microsoft.NET Framework version 1.1 and the Microsoft Internet Information Web Server (IIS).

Installation

1. Using the installation MSI, install the Wrapper to a location of your choosing.
2. Using the IIS Administration Tool, add a virtual directory, <vdir>, that points to the install directory from step one. (<vdir> is a place holder for the name you choose for the virtual directory, for example **OPC_XML-da_Wrapper**).

Simple Configuration

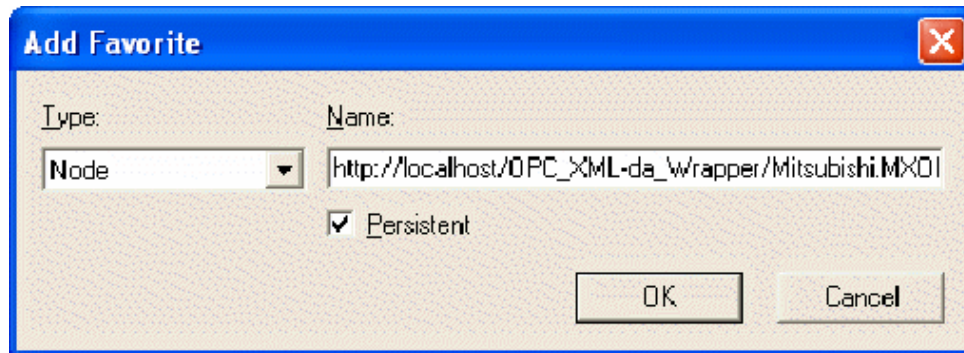
The simplest way to expose a local COM OPC Server is to make a copy of **OPC_XML-DAWrapperService.asmx** and change the base name of the file to match the ProgID of the COM server. The ProgID is human-readable unique identifier for the COM Server commonly shown when browsing for OPC servers. For the Mitsubishi OPC server the file name would be: **Mitsubishi.MXOPC.asmx** That's it. You should be able to access the server via the URL: <http://localhost/<vdir>/Mitsubishi.MXOPC.asmx>, as on the next screenshot:



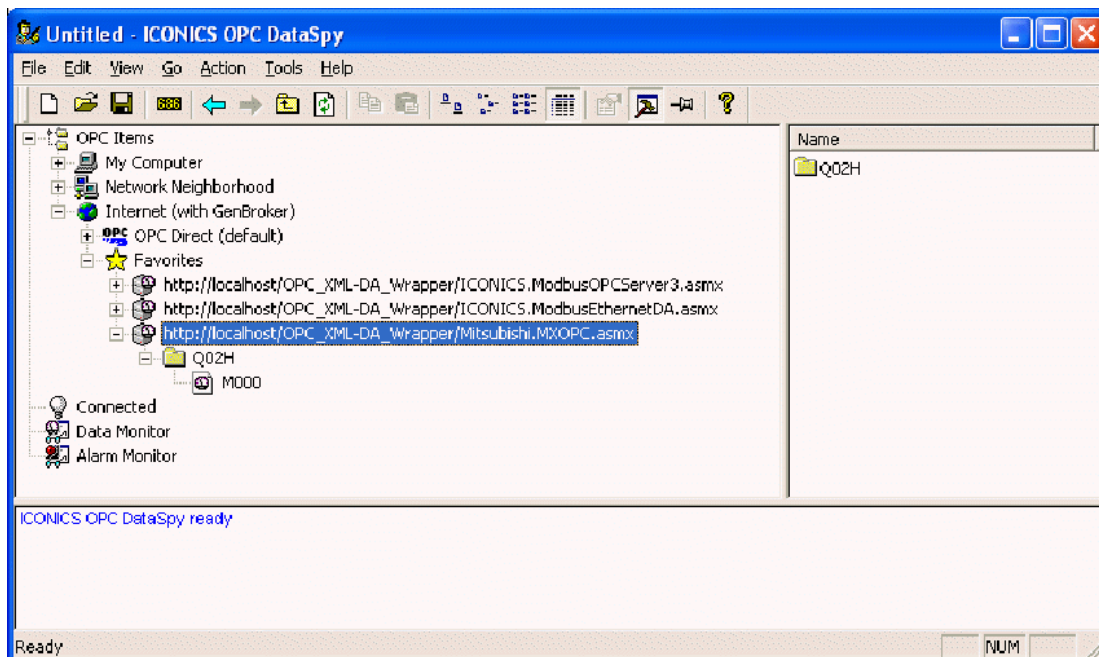
Connecting the MX OPC Server via XML (HTTP) from Microsoft Internet Explorer

Viewing Data from OPC DataSpy

1. Open any client such as **OPC DataSpy**.
2. In OPC DataSpy, go to the location **My Computer** → **Internet** → **Favorites**
3. Right click on **Favorites** and select **Add Favorite**
4. Add the address of the XML-DA wrapper as a favorite and click OK



The result will look like on the next screenshot:



Browsing the MX OPC Server's address space via XML (HTTP)

6.9 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. 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 Appendix A), 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>]
```

which specifies “bit-within-word” addressing to read the bit number specified (decimal format), similar to addressing bits within a tag using the ‘Tag.1’ syntax. The alternative is:

```
<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.

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:

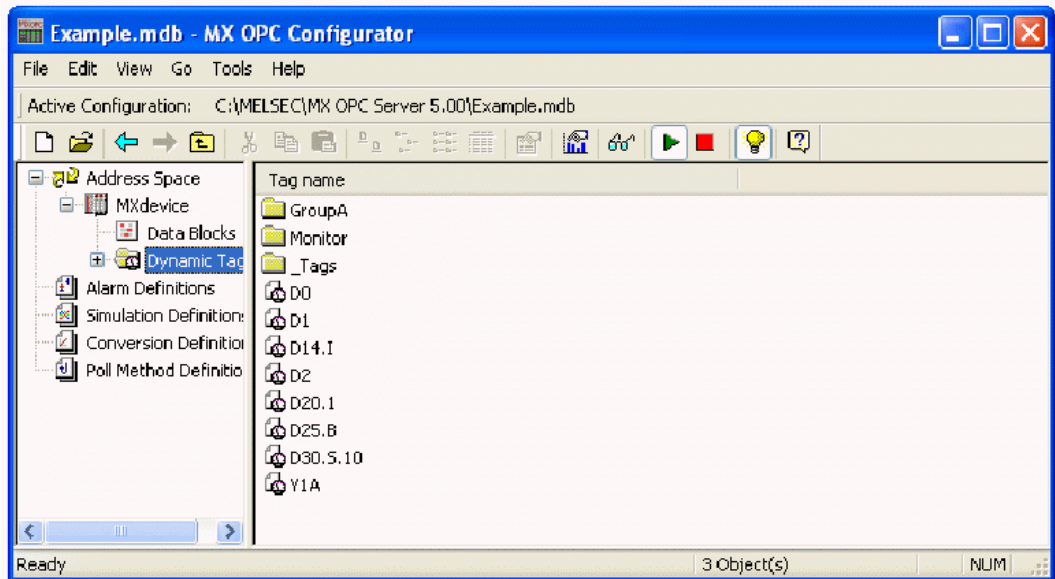
<Type> Part	Abbreviation
Bool	B
Word	W
Dword	DW
Int	I
Dint	DI
Real	R
String	S

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 'Q02H'.

Q02HD20	Address D20, default type
Q02HD20.W	Address D20, read as word type
Q02HD20.1	Bit 1 of address D20 (will be Boolean)
Q02H\GroupA\D40	Address D40, within a group called 'GroupA'
Q02HD41.Dint	DINT at address D41
Q02HX1A	Address X1A, in a register type that uses hexadecimal addressing on this PLC
Q02HD50.I.10	Array[10] of int
Q02HD100.S.10	String of length 10

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.



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.

Data Blocks and Overlapping Address Ranges

Data blocks for dynamic tags are always calculated automatically, whenever dynamic tags are added or released.

Although tags are not normally allowed to overlap in OPC server, dynamic tags are processed separately from fixed tags (tags that were added to the database by the configurator). As such, it is possible to use a fixed tag at e.g. address `D0` and simultaneously use a dynamic tag such as `PLC\D0.I`. However, dynamic tags may not be added with addresses that overlap other dynamic tags on the same MX device (such as two tags `PLC\D0.I.10` and `PLC\D0.I`). If this happens, the last overlapping item will return a bad quality when the application tries to use it.

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.**

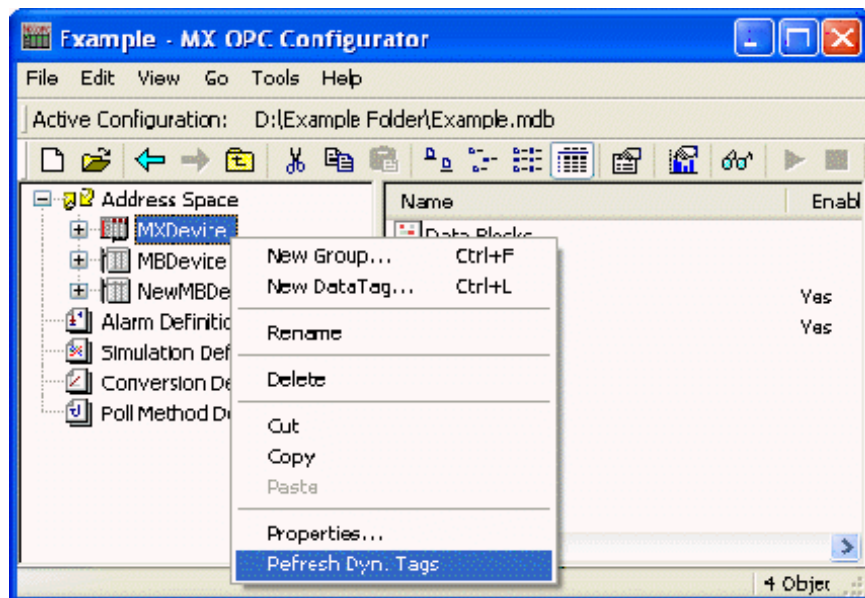
This new item is available for the Address Space, MX Devices and the Dynamic Tags folder in each device, and for individual dynamic tag groups. Refreshing each of the items has the following results:

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.

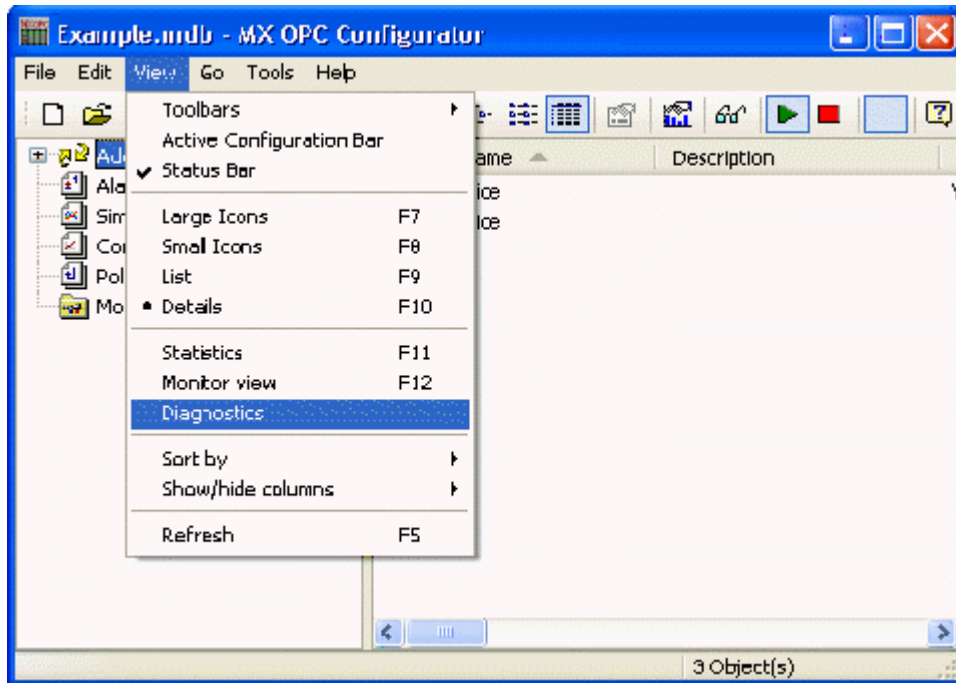


6.10 Automatic Backup

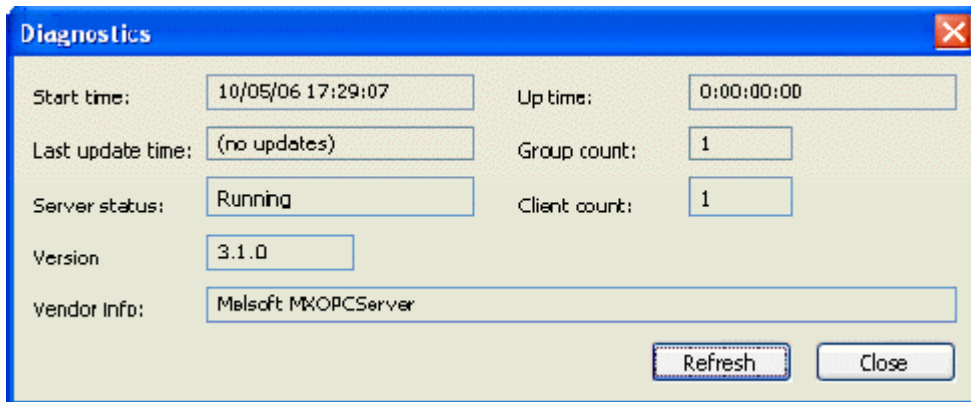
Each time MX OPC Configurator starts up, it now makes a backup copy of the opening database to the same folder where the original resides. The name of the backed-up database is the original database name but with the extension "bak" added. The resulting filename will be for example "MXConfigurator.mdb.bak".

6.11 Diagnostics

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



Selecting this item will open a new modeless dialog box, which shows various diagnostic information from the connected OPC server.



- **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 5.00. These options are described in [Section 2.5.3](#).

6.12 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.

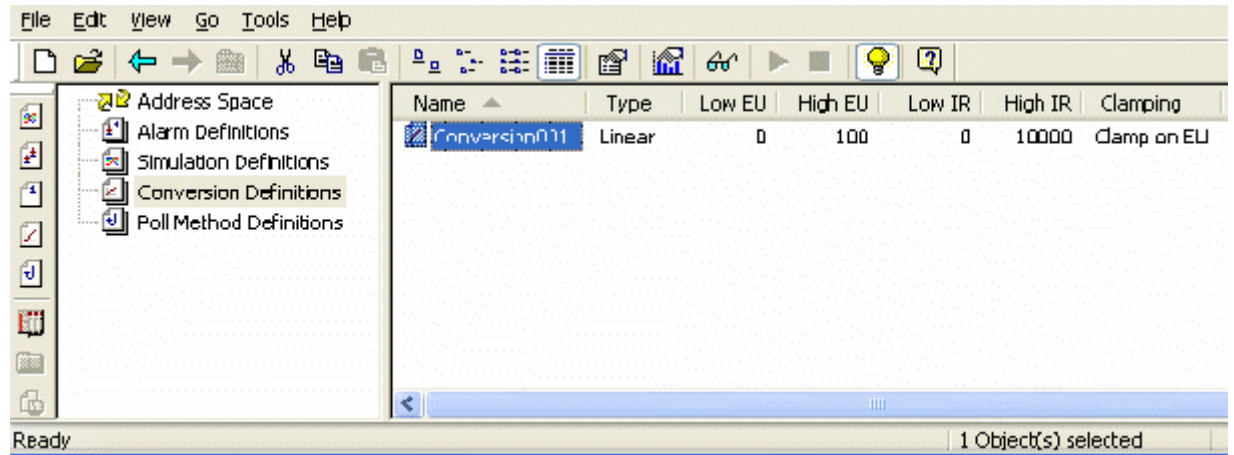
This function can be enabled or disabled through the Options dialog.

Conversion Definitions

7

7.1 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

7.1.1 Basic Conversion Properties

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

Conversion Properties: Basic Tab

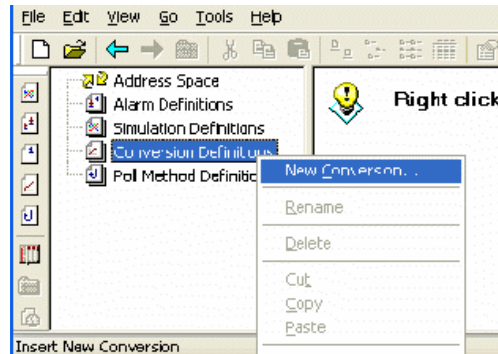
- **Name:** Specifies the name of the conversion definition. The name can be up to 12 alphanumeric characters, including underscores (_) and hyphens (-).
- **Conversion Type:** For each conversion definition, select one of the following conversion types from the drop-down list:
 - * **None (make float):** 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.

7.1.2 Creating a New Conversion Definition

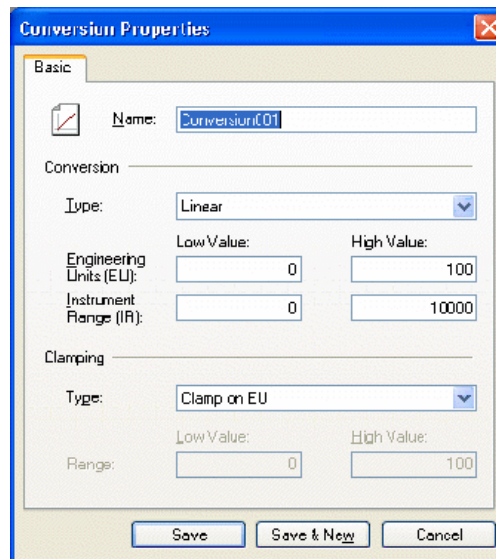
To create a new conversion definition:

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

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

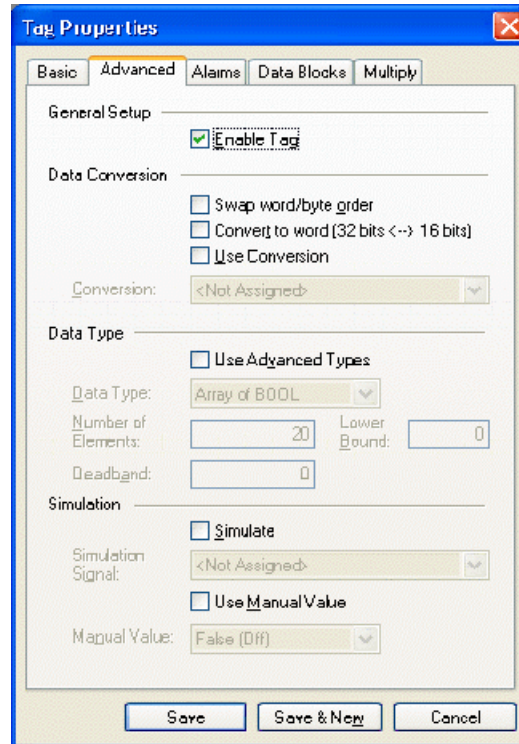


Setting the Properties for the New Conversion Definition

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

7.2 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

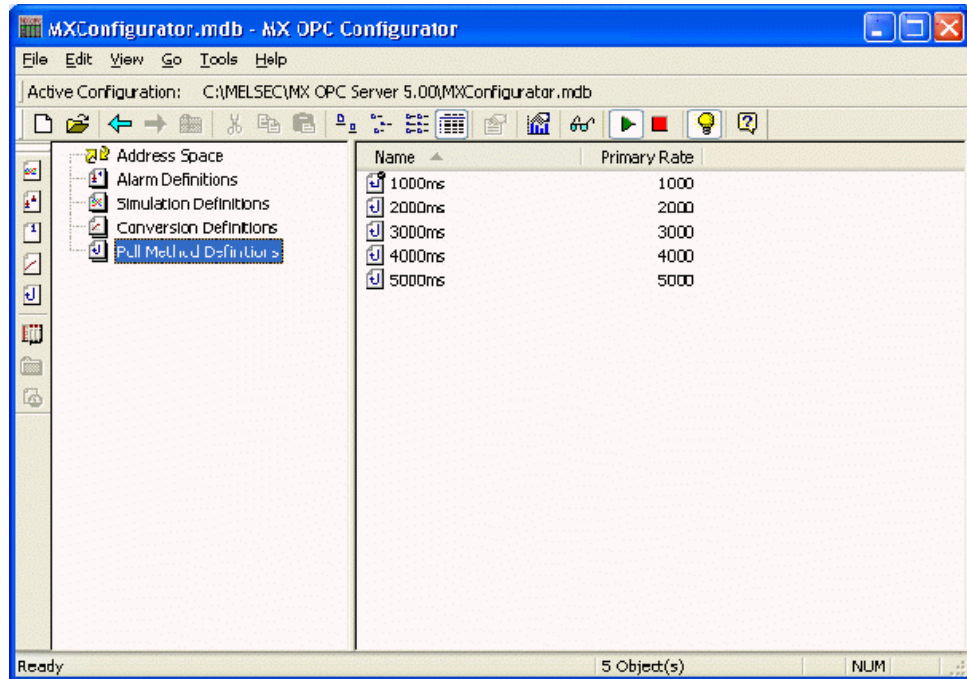
To select a conversion definition, check the **Use Conversion** check box and then select a conversion definition from the drop-down list, which lists all conversion definitions configured in the **Conversion Definitions** tree control of the Configurator.

Poll Method Definitions

8

8.1 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 and data blocks. Polling methods can be associated with data tags and data blocks via the tag properties and data block properties dialog boxes. The configurator provides several preconfigured polling methods.

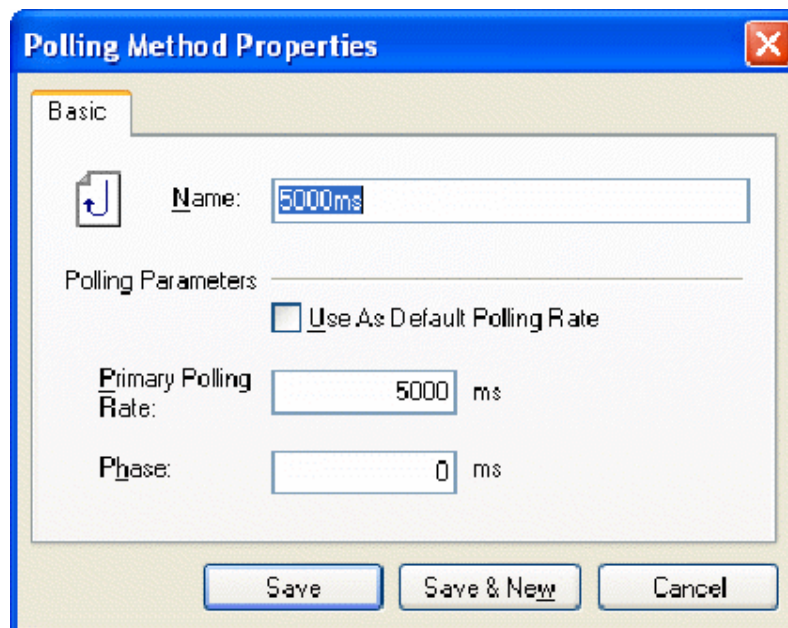


Poll Method Definitions Tree Control

8.1.1 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 12 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 block 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.

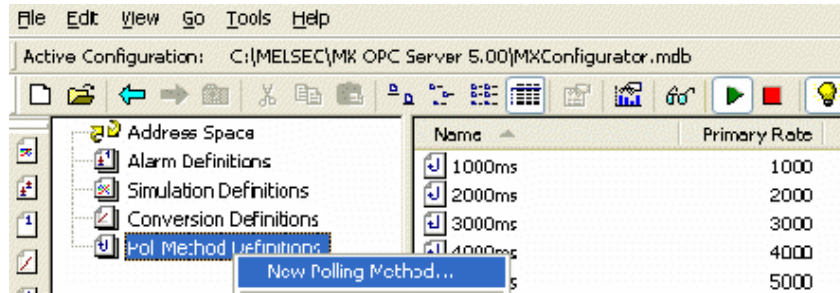


Polling Method Properties: Basic Tab

8.1.2 Creating a New Poll Method Definition

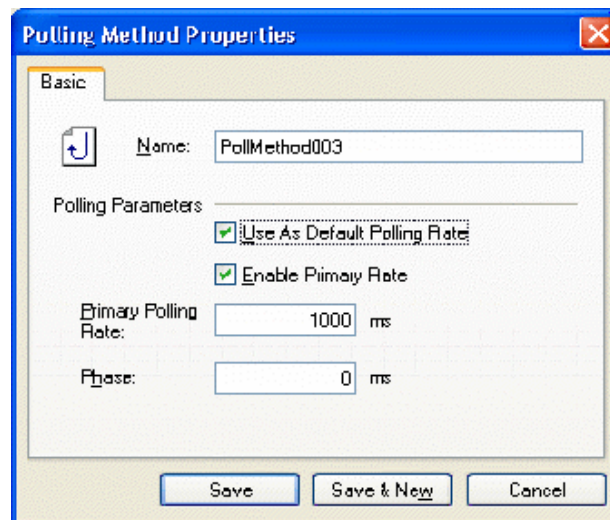
To create a new poll method definition:

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.

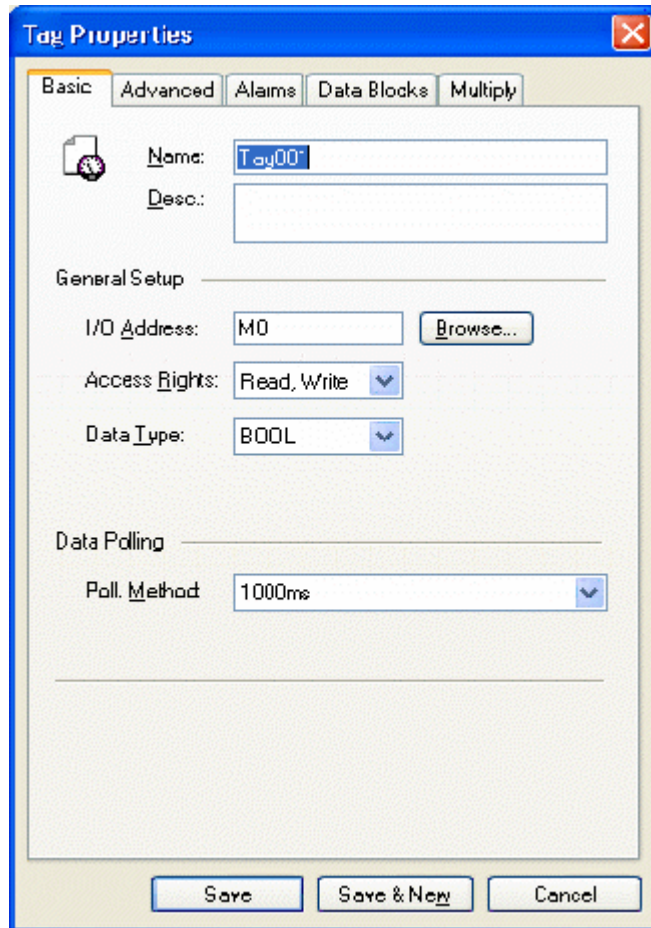


Setting the Properties for the New Polling Method

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 block 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.

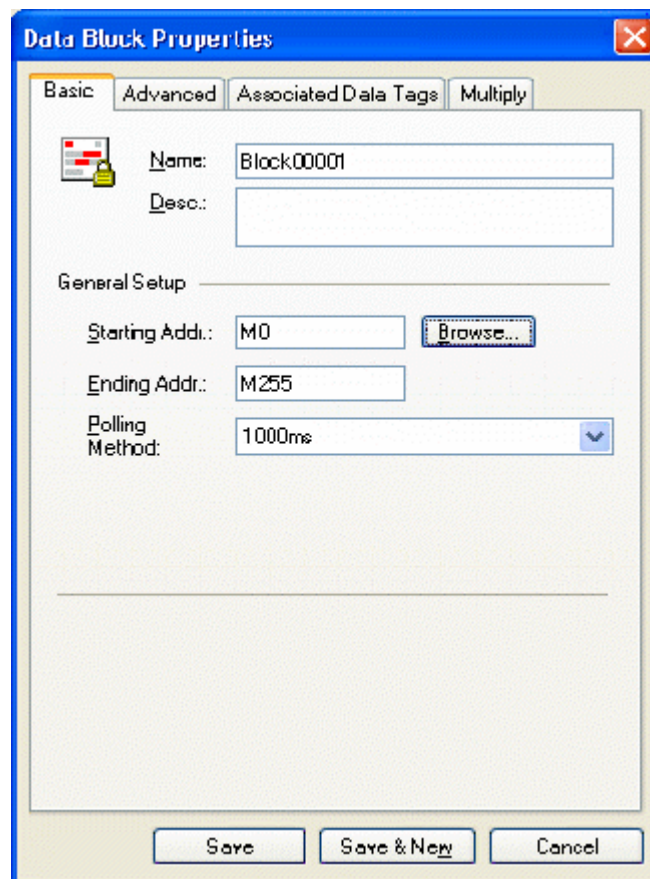
8.2 Assigning Poll Method Definitions to Tags and Data Blocks

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

Polling methods also can be associated with data blocks via the **Basic** tab of the **Data Block 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 Block Properties: Basic Tab

***Using an OPC Server
from Visual Basic***

A

A.1 Introduction

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.

A.2 Prerequisites

You will need:

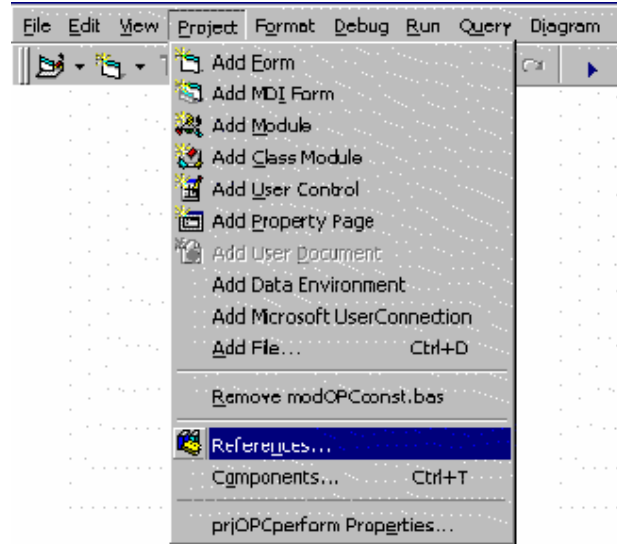
- Visual Basic - the examples in this document were developed with Visual Basic 6 Service Pack 5, but could also be used with other applications that will support VBA (Visual Basic for Applications).
- OPC automation support files (OPCdaAuto.dll and OPCenum.exe) – these must be correctly installed in e.g. the <WINDOWS\SYSTEM32> directory, and must be registered with REGSVR32.
- An installed OPC server, e.g. Mitsubishi.MXOPC

A.3 Terminology

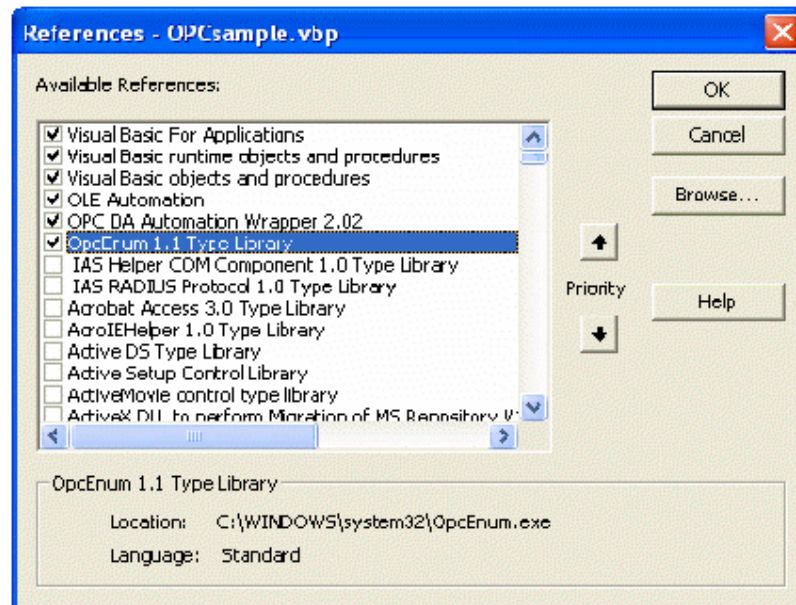
Term	Description
OPC	“OLE for Process Control” – an open standard for connecting to process control equipment such as PLCs and SCADA databases. OPC is intended to solve the problem of every SCADA package needing a different driver for every PLC to which it connects. With OPC, only one driver needs to 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 (Visual Basic for Applications).
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.

A.4 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.



A.5 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

A.6 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.5', 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
```

A.7 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"
```

A.8 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 tvwBrowse. 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](#) and [Writing data values to the server](#)). 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.I' 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'.

' 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: twwT - 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 twwT 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 = twwT.Nodes.Add(nodPar, twwChild, , _
 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 = twwT.Nodes.Add(nodPar, twwChild, _
                          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 twwT, 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
    twwBrowse.PathSeparator = "."

    ' Fill in treeview (first adding a root node)
    twwBrowse.Nodes.Clear
    Set nodRoot = twwBrowse.Nodes.Add(, , strRootKey, _
        cmbServer.List(cmbServer.ListIndex))
    FillBranches twwBrowse, 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(IngCnt) = 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()
```

A.9 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
        End If

        ' Item defined correctly - get server handle
        lngSvrHandles(lngLp) = objItem.ServerHandle

        ' Clean up reference
        Set objItem = Nothing
    Next lngLp
End Sub

```

End If ' Item failed to define / defined OK

Next IngLp ' 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

A.10 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 1&, 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 1&, 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"

A.11 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
```

A.12 References

The references below are recommended for further information about using OPC from Visual Basic, and for information about OPC in general.

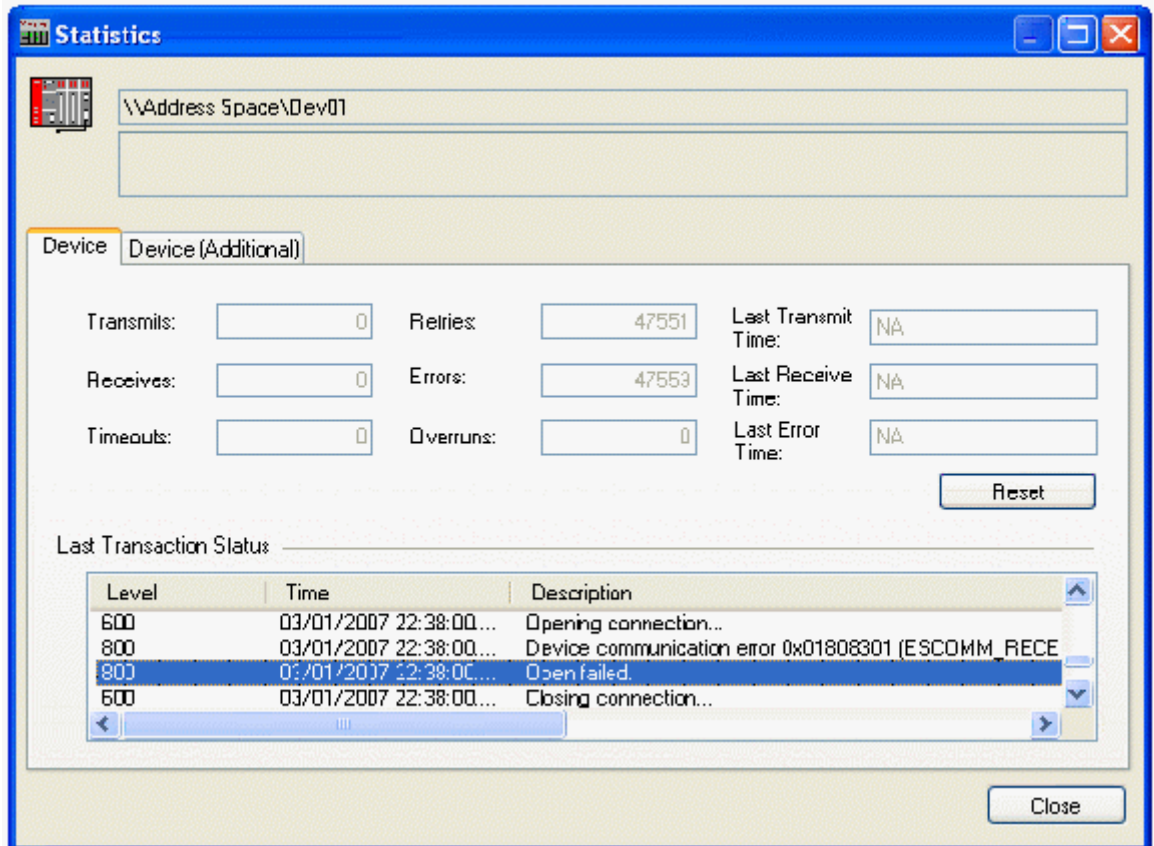
- OPC Data Access Automation Specification
- <http://www.opcfoundation.org>
- http://www.iconics.com/support/free_tools.asp

ACT error codes

B

B.1 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 section B.2

ACT error codes, returned by the communications layer – see section B.3

CPU, module and network board error codes, returned by a particular device – see section B.4

B.2 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.

B.3

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.

Error Code: 0x01802002

Error Definition: Device number error. The device character string number specified in the method is an unauthorised device number.

Action: Review the device number.

Error Code: 0x01802003

Error Definition: Program Type Error

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802004

Error Definition: Sumcheck error. The sumcheck value of the received data is abnormal.

Action: 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.

Error Code: 0x01802005

Error Definition: Size error. The number of points specified in the method is unauthorised.

Action: Check the number of points specified in the method. Review the system, e.g. PLC CPU, module setting and cable status. Exit the program and restart the computer. Reinstall MX Component.

Error Code: 0x01802006

Error Definition: Block number error. The block specifying number in the device character string specified in the method is unauthorised.

Action: Review the block specifying number in the device character string specified in the method.

Error Code: 0x01802007

Error Definition: Receive data error. The data received is abnormal.

Action: Review the system, e.g. PLC CPU, module setting and cable status. Check the cable. Exit the program and restart the computer.

Error Code: 0x01802008

Error Definition: Write Protect Error

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802009

Error Definition: Reading Parameters error

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x0180200A
Error Definition: Writing Parameters error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0180200B
Error Definition: 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.
Action: 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.
- Error Code:** 0x0180200C
Error Definition: Request Cancel Error. The request was cancelled while being processed.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0180200D
Error Definition: Drive Name Error. The specified drive name is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0180200E
Error Definition: Beginning Step Error. The beginning step specified is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0180200F
Error Definition: Parameter Type Error. The parameter type is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01802010
Error Definition: File Name Error. The file name is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01802011
Error Definition: Status Error. The status of Registration/Cancellation/Setting is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01802012
Error Definition: Detailed Condition Field Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x01802013
Error Definition: Step Condition Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01802014
Error Definition: Bit Device Condition Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01802015
Error Definition: Parameter Settings Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01802016
Error Definition: Error in specifying telephone exchange number. Method does not support the operations corresponding to the specified telephone exchange number.
Action: Check the telephone exchange number. Check if the method being executed is supported or not. Check the system configuration such as PLC, unit, etc.
- Error Code:** 0x01802017
Error Definition: Keyword Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01802018
Error Definition: Read/Write Flag Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01802019
Error Definition: Refresh Method Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0180201A
Error Definition: Buffer Access Method Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0180201B
Error Definition: Start Mode/Stop Mode Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x0180201C
Error Definition: Written clock data error. Clock data specified for write cannot be written properly since that data is in error.
Action: Review the clock data to be written.
- Error Code:** 0x0180201D
Error Definition: Online clock data write error. Write of clock data failed. Clock data cannot be written since the PLC CPU is in RUN.
Action: Place the PLC CPU in the STOP status.
- Error Code:** 0x0180201E
Error Definition: ROM drive Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0180201F
Error Definition: While Tracing error. Invalid operation was carried out during trace.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01802020
Error Definition: First I/O number error. The first I/O number specified in the method is an unauthorised value.
Action: 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.
- Error Code:** 0x01802021
Error Definition: First address error. The buffer address specified in the method is an unauthorised value.
Action: Check the value of the buffer address specified in the method. Exit the program and restart the computer.
- Error Code:** 0x01802022
Error Definition: Pattern Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01802023
Error Definition: SFC Block No. Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01802024
Error Definition: SFC Step No. Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802025
Error Definition: Step No. Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802026
Error Definition: Data Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802027
Error Definition: System Data Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802028
Error Definition: Error in number of TC settings Value
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802029
Error Definition: Clear Mode Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0180202A
Error Definition: Signal Flow Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0180202B
Error Definition: Version Control Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0180202C
Error Definition: Monitor Not Registered error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0180202D
Error Definition: PI Type Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0180202E
Error Definition: PI No Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0180202F
Error Definition: Error in Number of PIs
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802030
Error Definition: Shift Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802031
Error Definition: File Type Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802032
Error Definition: Specified Unit error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802033
Error Definition: Error check flag Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802034
Error Definition: Step RUN operation error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802035
Error Definition: Step RUN data error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802036
Error Definition: During Step RUN error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802037
Error Definition: Write error while running program corresponding to E2PROM
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802038
Error Definition: Clock data read/write error The clock data read/write method was executed for the PLC CPU which does not have the clock devices.
Action: Do not execute clock data read/write.

Error Code: 0x01802039
Error Definition: Trace not completed error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0180203A
Error Definition: Registration Clear Flag Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0180203B
Error Definition: Operation error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0180203C
Error Definition: Error in the number of exchanges
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0180203D
Error Definition: Error in number of loops specified
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0180203E
Error Definition: Retrieve data selection
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0180203F
Error Definition: Error in number of SFC cycles
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802040
Error Definition: Motion PLC Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802041
Error Definition: Motion PLC Communication error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802042
Error Definition: Fixed execution time setting error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x01802043
Error Definition: Error in number of functions
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01802044
Error Definition: System information specification error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01802045
Error Definition: Registration Condition Not Formed error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01802046
Error Definition: Function No. Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01802047
Error Definition: RAM drive error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01802048
Error Definition: ROM drive error at the booting side
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01802049
Error Definition: Transfer mode specification error at the booting side
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0180204A
Error Definition: Insufficient memory error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0180204B
Error Definition: Back up drive ROM error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0180204C
Error Definition: Block size error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0180204D
Error Definition: Detached during RUN state error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0180204E
Error Definition: Unit Already Registered error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0180204F
Error Definition: Password Registration Data Full error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802050
Error Definition: Password Not Registered error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802051
Error Definition: Remote Password Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802052
Error Definition: IP Address Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802053
Error Definition: Timeout value out of range error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802054
Error Definition: Command not detected error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802055
Error Definition: Trace execution type error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x01802056
Error Definition: Version error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x01802057
Error Definition: Tracking cable error. The tracking cable is faulty. The PLC CPU status is error.
Action: Reexamine the system such as the PLC CPU, module setting and cable status.
- Error Code:** 0x0180205C
Error Definition: Keyword protection error PLC is protected by the key word.
Action: Disable the keyword and execute again.
- Error Code:** 0x0180205D
Error Definition: Keyword disable error. The inputted keyword is wrong.
Action: Input a correct keyword.
- Error Code:** 0x0180205E
Error Definition: Keyword protecting error. PLC did not accept the protecting command.
Action: Execute again or re-switch the power of the PLC.
- Error Code:** 0x0180205F
Error Definition: Keyword entry error. An illegal character is included in the inputted keyword.
Action: Input a correct keyword.
- Error Code:** 0x01802060
Error Definition: Keyword deletion error. The inputted keyword is wrong.
Action: Input a correct keyword.
- Error Code:** 0x01808001
Error Definition: Multiple Open Error. Open method was executed while it was open
Action: Exit the program and restart the computer. Execute any method other than Open.
- Error Code:** 0x01808002
Error Definition: Channel number specifying error. The port number set to the property and the port number set on the communication settings utility are unauthorised values.
Action: Set the correct value to the port number of the property. Make communication settings again on the communication settings utility.
- Error Code:** 0x01808003
Error Definition: Driver not yet started. The network board driver is not started.
Action: The network board driver is not started. Start the driver.
- Error Code:** 0x01808004
Error Definition: Error in overlap event generation
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x01808005
Error Definition: MUTEX generation error. Creation of MUTEX to exercise exclusive control failed.
Action: Exit the program and restart the computer. Reinstall MX Component.
- Error Code:** 0x01808006
Error Definition: Error in socket object generation. Socket object could not be created
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01808007
Error Definition: Socket object generation error. Creation of the Socket object failed.
Action: 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.
- Error Code:** 0x01808008
Error Definition: Port connection error. Establishment of connection failed. The other end does not respond.
Action: 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.
- Error Code:** 0x01808009
Error Definition: 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.
Action: Check for an application which uses the COM port. Exit the program and restart the computer.
- Error Code:** 0x0180800A
Error Definition: Buffer size setting error. Setting of the COM port buffer size failed.
Action: 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.
- Error Code:** 0x0180800B
Error Definition: DCB value acquisition error. Acquisition of the COM port DCB value failed.
Action: 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.
- Error Code:** 0x0180800C
Error Definition: DCB setting error. Setting of the COM port DCB value failed.
Action: 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.

- Error Code:** 0x0180800D
Error Definition: Time-out value setting error. Setting of the COM port time-out value failed.
Action: 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.
- Error Code:** 0x0180800E
Error Definition: Shared memory open error. Open processing of shared memory failed.
Action: Check whether the GX Simulator has started. Exit the program and restart the computer.
- Error Code:** 0x01808101
Error Definition: Duplex close error
Action: Exit the program and restart the computer.
- Error Code:** 0x01808102
Error Definition: Handle close error. Closing of the COM port handle failed.
Action: Exit the program and restart the computer.
- Error Code:** 0x01808103
Error Definition: Driver close error. Closing of the driver handle failed.
Action: Exit the program and restart the computer.
- Error Code:** 0x01808104
Error Definition: Overlap Event Close Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01808105
Error Definition: Mutex Handle Close Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01808106
Error Definition: COM Port Handle Close Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01808201
Error Definition: Send error Data send failed.
Action: 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.

- Error Code:** 0x01808202
Error Definition: Send data size error. Data send failed.
Action: Exit the program and restart the computer.
- Error Code:** 0x01808203
Error Definition: Queue clear error. Clearing of the COM port queue failed.
Action: Exit the program and restart the computer. Perform Close once and execute Open again.
- Error Code:** 0x01808301
Error Definition: Receive error. Data receive failed.
Action: 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.
- Error Code:** 0x01808302
Error Definition: Not Sent error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01808303
Error Definition: Error in retrieving Overlap Event
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01808304
Error Definition: Receive buffer size shortage. Receive data was larger than the receive buffer size prepared for the system.
Action: Exit the program and restart the computer.
- Error Code:** 0x01808401
Error Definition: Control error. Changing of the COM port communication control failed.
Action: Exit the program and restart the computer.
- Error Code:** 0x01808402
Error Definition: Signal Line Control Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01808403
Error Definition: Signal line specifying error. Changing of the COM port communication control failed.
Action: Exit the program and restart the computer.
- Error Code:** 0x01808404
Error Definition: Open not yet executed
Action: Execute Open. Exit the program and restart the computer.

- Error Code:** 0x01808405
Error Definition: Communication parameter error. The data bit and stop bit combination of the properties is unauthorised.
Action: Review the data bit and stop bit values of the properties. Set them again on the communication settings utility.
- Error Code:** 0x01808406
Error Definition: Transmission speed value specifying error. The transmission speed of the property is unauthorised.
Action: Review the transmission speed value of the property. Set it again on the communication settings utility.
- Error Code:** 0x01808407
Error Definition: Data length error. The data bit value of the property is unauthorised.
Action: Review the data bit value of the property. Set it again on the communication settings utility.
- Error Code:** 0x01808408
Error Definition: Parity specifying error. The parity value of the property is unauthorised.
Action: Review the parity value of the property. Set it again on the communication settings utility.
- Error Code:** 0x01808409
Error Definition: Stop bit specifying error. The stop bit value of the property is unauthorised.
Action: Review the stop bit value of the property. Set it again on the communication settings utility.
- Error Code:** 0x0180840A
Error Definition: Communication control setting error. The control value of the property is unauthorised.
Action: Review the control value of the property. Set it again on the communication settings utility.
- Error Code:** 0x0180840B
Error Definition: Time-out error. Though the time-out period had elapsed, data could not be received.
Action: 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.
- Error Code:** 0x0180840C
Error Definition: Connect error
Action: Exit the program and restart the computer.
- Error Code:** 0x0180840D
Error Definition: Duplex connect error
Action: Exit the program and restart the computer.

- Error Code:** 0x0180840E
Error Definition: Attach failure. Attaching of the socket object failed.
Action: Exit the program and restart the computer.
- Error Code:** 0x0180840F
Error Definition: Signal line status acquisition failure. Acquisition of the COM port signal line status failed.
Action: Exit the program and restart the computer.
- Error Code:** 0x01808410
Error Definition: CD signal line OFF. The CD signal on the other end of communication is in the OFF status.
Action: Review the system, e.g. PLC CPU, module setting and cable status. Exit the program and restart the computer.
- Error Code:** 0x01808411
Error Definition: Password mismatch error
Action: Check the remote password of the property.
- Error Code:** 0x01808412
Error Definition: TEL Communication Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x01808501
Error Definition: USB driver load error. Loading of the USB driver failed.
Action: Exit the program and restart the computer. Reinstall MX Component.
- Error Code:** 0x01808502
Error Definition: USB driver connect error. Connection of the USB driver failed.
Action: Exit the program and restart the computer. Reinstall MX Component.
- Error Code:** 0x01808503
Error Definition: USB driver send error. Data send failed.
Action: Review the system, e.g. PLC CPU, module setting and cable status. Make USB setting on the control panel (device manger) of the OS. Retry the method. Exit the program and restart the computer.
- Error Code:** 0x01808504
Error Definition: USB driver receive error. Data receive failed.
Action: Review the system, e.g. PLC CPU, module setting and cable status. Make USB setting on the control panel (device manger) of the OS. Retry the method. Exit the program and restart the computer.
- Error Code:** 0x01808505
Error Definition: USB Driver Timeout Error
Action: Recheck the timeout value. Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x01808506
Error Definition: USB driver initialisation error. Initialisation of the USB driver failed.
Action: Make USB setting on the control panel (device manger) of the OS. Exit the program and restart the computer.
- Error Code:** 0x01808507
Error Definition: Other USB error. Error related to data send/receive occurred.
Action: Disconnect the cable once, then reconnect. Exit the program and restart the computer. Reinstall MX Component.
- Error Code:** 0x02000001
Error Definition: Points Exceeded error. The number of points registered in the monitoring server is very high.
Action: 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.
- Error Code:** 0x02000002
Error Definition: Shared memory creation error. Failed in creating shared memory.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x02000003
Error Definition: Shared memory access error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x02000004
Error Definition: Memory Secure error. Failed in securing memory for the monitoring server.
Action: 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.
- Error Code:** 0x02000005
Error Definition: Device Not Registered error. Monitor has not been registered
Action: 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.
- Error Code:** 0x02000006
Error Definition: Monitoring Server Startup Error. Monitoring Server is not started.
Action: Start the Monitoring Server. Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x02000010
Error Definition: Yet to retrieve Device Value error. Monitoring is not yet completed
Action: 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.
- Error Code:** 0x03000001
Error Definition: Command not Supported. Command is not supported.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03000002
Error Definition: Memory Lock Error Failed while locking memory.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03000003
Error Definition: Error Securing Memory. Failed in securing the memory.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03000004
Error Definition: DLL read error. Failed in reading DLL.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03000005
Error Definition: Error in securing Resources. Failed in securing the resources.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03010001
Error Definition: File Creation Error. Failed in creating the file.
Action: 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.
- Error Code:** 0x03010002
Error Definition: File Open Error. Failed to open the file.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03010003
Error Definition: Buffer Size Error. The buffer size specified is either incorrect or not enough.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x03010004
Error Definition: SIL Sentence formation error. SIL sentence formation is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03010005
Error Definition: Filename Error. The specified filename is too long.
Action: Specify a shorter filename. Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03010006
Error Definition: File does not exist error. The specified file does not exist.
Action: 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.
- Error Code:** 0x03010007
Error Definition: File Structure Error. The data structure in the specified file is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03010008
Error Definition: File already exists error. The specified file already exists.
Action: Check the filename. Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03010009
Error Definition: File does not exist error. The specified file does not exist.
Action: Check the filename. Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0301000A
Error Definition: File Deletion Error. The specified file could not be deleted.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0301000B
Error Definition: Multiple Open Error. The specified project has been opened twice.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0301000C
Error Definition: Filename Error. The specified filename is incorrect.
Action: Check the filename. Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x0301000D
Error Definition: File Read Error. Failed in reading the file.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0301000E
Error Definition: File Write Error. Failed in writing the file.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0301000F
Error Definition: File Seek Error. File seek failed.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03010010
Error Definition: File Close Error. Failed while closing the file.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03010011
Error Definition: Folder Creation Error. Failed while creating the folder.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03010012
Error Definition: File Copy Error. Failed while copying the file.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03010013
Error Definition: Project Path Error. The length of the project path is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03010014
Error Definition: Project Type Error. The project type is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03010015
Error Definition: File Type Error. The file type is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03010016
Error Definition: Sub-File Type Error. The sub-file type is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x03010017
Error Definition: Insufficient Disk space error. The disk space is insufficient.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x03020002
Error Definition: Multiple Open Error. Tried to open DBProduct more than once.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x03020003
Error Definition: Not Opened error. DBProduct is not opened.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x03020004
Error Definition: Extract Error. DBProduct is not extracted.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x03020010
Error Definition: Parameter Error. The parameters of DBProduct are incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x03020011
Error Definition: Language Error. The language parameter is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x03020012
Error Definition: Error in specifying Maker. The maker parameter is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x03020013
Error Definition: Error in specifying Unit. The unit parameter is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x03020014
Error Definition: SQL Parameter Error SIL, SQL Parameter of DBProduct is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x03020015
Error Definition: SIL Sentence formation error. SIL sentence formation is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x03020016
Error Definition: Field Key Input Error. The field key entered is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03020050
Error Definition: Record Data Construction Error. Failed in reconstructing the record data of DBProduct.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03020060
Error Definition: Error Retrieving Record Data. Failed while retrieving DBProduct record data.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03020061
Error Definition: Last Record error. Cannot retrieve the next record since the current record is the last record.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03FF0000
Error Definition: Initialization error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03FF0001
Error Definition: Not Initialized error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03FF0002
Error Definition: Multiple Initialization error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03FF0003
Error Definition: Workspace Initialization Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03FF0004
Error Definition: Database Initialization Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x03FF0005
Error Definition: Recordset Initialization Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03FF0006
Error Definition: Error Closing Database
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03FF0007
Error Definition: Error Closing Recordset
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03FF0008
Error Definition: Database Not Opened error. Database is not opened.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03FF0009
Error Definition: Recordset Not Opened error. Recordset is not opened.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03FF000A
Error Definition: Table Initialization Error. Failed in initializing TtableInformation table
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03FF000B
Error Definition: Table Initialization Error. Failed in initializing TfieldInformation table
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03FF000C
Error Definition: Table Initialization Error. Failed in initializing TrelationInformation table
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x03FF000D
Error Definition: Table Initialization Error. Failed in initializing Tlanguage table
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x03FF000E
Error Definition: Table Initialization Error. Failed in initializing Tmaker table
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x03FF000F
Error Definition: Table Initialization Error. Failed in initializing TOpenDatabase table
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x03FF0010
Error Definition: Field Value Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x03FF0011
Error Definition: Field Value Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x03FF0012
Error Definition: Exit Error. Failed to exit the database.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x03FF0100
Error Definition: Moving Record error. Failed while moving the record.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x03FF0101
Error Definition: Retrieving Record Count Error. Failed to retrieve the record count.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x03FF0110
Error Definition: Retrieving Field Value Error. Failed in retrieving the field value.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x03FF0111
Error Definition: Setting Field Value Error. Failed in setting the field value.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x03FFFFFF
Error Definition: Other Errors
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x04000001
Error Definition: No command error. The specified CPU type cannot be used to perform processing.
Action: Check the CPU type set to ActCpuType. Check whether the system configuration is supported or not. Exist the program and restart the computer. Reinstall MX Component.
- Error Code:** 0x04000002
Error Definition: Memory lock error. Failed in locking memory.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04000003
Error Definition: Securing Memory Error. Failed in securing the memory.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04000004
Error Definition: Internal server DLL load error. Start of the internal server failed.
Action: Check for the deleted or moved installation file of MX Component. Exit the program and restart the computer. Reinstall MX Component.
- Error Code:** 0x04000005
Error Definition: Securing Resources Error. Failed in securing the resources.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04000006
Error Definition: Error Loading Main Object. Failed in reading the file.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04000007
Error Definition: Error Loading Conversion Table. Failed in reading table data.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04000100
Error Definition: Incorrect Intermediate Code Size error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04010001
Error Definition: Intermediate Code Not Converted error. The converted machine code for one command is more than 256 bytes.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x04010002
Error Definition: Intermediate Code Completion Error. Intermediate code area of the code to be converted ended abruptly.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04010003
Error Definition: Insufficient Intermediate Code error. The intermediate code of the code to be converted was insufficient.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04010004
Error Definition: Intermediate Code Data Error. The intermediate code to be converted is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04010005
Error Definition: Intermediate Code Structure Error. The number of steps in the intermediate code is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04010006
Error Definition: Error in Number of Steps. The number of steps in comment intermediate code is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04010007
Error Definition: Insufficient Storage Space for Machine Code error. The storage space for machine code is insufficient.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04010008
Error Definition: Other Errors (Other errors generated during the conversion of Intermediate code to machine code.)
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04011001
Error Definition: Machine Code Not Converted error. The converted intermediate code for one command is more than 256 bytes.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04011002
Error Definition: Machine Code Completion Error. The machine code area to be converted ended abruptly.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04011003

Error Definition: Abnormal Machine Code. Could not convert since the machine code to be converted was abnormal.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04011004

Error Definition: Insufficient Storage Space for Intermediate Code error. The storage area for intermediate code is insufficient.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04011005

Error Definition: Other Errors. Other errors generated while converting machine code to Intermediate code.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04020001

Error Definition: Text Code Not Converted error. The converted intermediate code for one command is more than 256 bytes.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04020002

Error Definition: No Input error. The input list code is insufficient.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04020003

Error Definition: Command Error. The command name of list code to be converted is incorrect.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04020004

Error Definition: Device Error. The device name of list code to be converted is incorrect.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04020005

Error Definition: Device Number Error. The device number of the list code to be converted is out of range.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04020006

Error Definition: Conversion Error. The list code to be converted conversion could not be identified.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x04020007
Error Definition: Text Data Error. The list code to be converted is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04020008
Error Definition: Error in SFC Operation Output. The output command of SFC operation is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04020009
Error Definition: SFC Shift Condition Error. SFC shift condition command is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0402000A
Error Definition: Error in Statements between lines. The statements entered between lines are incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0402000B
Error Definition: P.I Statement Error. The P.I statement entered is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0402000C
Error Definition: Note Error. The Note entered is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0402000D
Error Definition: Comment Error. The comment entered is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0402000E
Error Definition: Other Errors (Other errors generated during the conversion of list to Intermediate code)
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04021001
Error Definition: Intermediate Code Not Converted error. The converted list code for one command has exceeded 256 bytes.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x04021002
Error Definition: Intermediate Code Area Full error. Intermediate code area to be converted is full.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04021003
Error Definition: Command Error. The command specified by the intermediate code to be converted is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04021004
Error Definition: Device Error. The device specified in the intermediate code to be converted is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04021005
Error Definition: Intermediate Code Error. The structure of intermediate code to be converted is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04021006
Error Definition: Insufficient List Storage Space error. The space for storing the converted list code is insufficient.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04021007
Error Definition: Other Errors (Other errors generated during the conversion of intermediate code to list)
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04030001
Error Definition: Not Converted error. The storage space for converted intermediate code is insufficient.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04030002
Error Definition: Bad Circuit Creation error. The character memory circuit is not completed in a sequence.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04030003
Error Definition: Specified Circuit Size Exceeded. Specified circuit size is too big.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04030004

Error Definition: Incorrect Return Circuit error. There is no consistency before and after the return circuit. The setting for the return circuit is too high.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04030005

Error Definition: Other Errors (Other errors generated while converting from Character Memory to Intermediate Code)

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04031001

Error Definition: Not Converted error. The size (vertical/horizontal) of the character memory specified is incorrect.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04031002

Error Definition: Abnormal Command Code error. The command intermediate code to be converted is incorrect.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04031003

Error Definition: Bad Circuit Creation error. Could not be converted to Sequence Circuit. There is no END command.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04031004

Error Definition: Specified Circuit Size exceeded error. Specified circuit size is too big.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04031005

Error Definition: Fatal Error. Fatal Error has occurred.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04031006

Error Definition: Insufficient number of storage blocks error. The space to store the converted character memory circuit blocks is not sufficient.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04031007

Error Definition: Circuit Block Search Error. Data is broken off in the circuit block.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x04031008
Error Definition: Other Errors (Other errors generated during the conversion of intermediate code to character memory)
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04040001
Error Definition: CAD Data Error. There is no CAD data to be converted. The CAD data format is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04040002
Error Definition: Output Data Error. The input CAD data type and the output CAD data type are not matching.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04040003
Error Definition: Library Load Error. Failed to load the library.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04040004
Error Definition: Storage Space Secure Error. The space secured to store the converted data is not sufficient.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04040005
Error Definition: No END Command error. There is no END command in the CAD data to be converted.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04040006
Error Definition: Abnormal Command Code. There is abnormal command code in the CAD data to be converted.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04040007
Error Definition: Device No. Error. The device number is out of range.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04040008
Error Definition: Step No. Error. The step number is out of range.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x04040009
Error Definition: The specified circuit size exceeded error. 1 circuit block is too big.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0404000A
Error Definition: Return Circuit Error. The return circuit is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0404000B
Error Definition: Bad Circuit Creation error. The circuit data is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0404000C
Error Definition: SFC Data Error. The SFC data to be converted is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0404000D
Error Definition: List Data Error. The list data to be converted is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0404000E
Error Definition: Comment Data Error. The comment data to be converted is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0404000F
Error Definition: Statement Error. The statement data to be converted is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04040010
Error Definition: Other Errors (Other errors generated during the conversion of CAD code to Intermediate code.)
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04041001
Error Definition: Intermediate Code Data Error. There is no intermediate code to be converted. The format of the intermediate code is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x04041002
Error Definition: CAD Data Type Error. The input CAD data type and the output CAD data type are not matching.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04041003
Error Definition: Library Error. Failed to load the library.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04041004
Error Definition: Insufficient Input Data error. Data to be converted is insufficient.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04041005
Error Definition: Insufficient Storage Space error. There is not enough space to store the CAD data to be converted.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04041006
Error Definition: No END Command error. There is no END command in the CAD data to be converted.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04041007
Error Definition: Abnormal Command Code. There is abnormal command code in the CAD data to be converted.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04041008
Error Definition: Device No. Error. The device number is out of range.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04041009
Error Definition: Step No. Error. The step number is out of range.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0404100A
Error Definition: The specified circuit size exceeded error. 1 circuit block is too big.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x0404100B
Error Definition: Return Circuit Error. The return circuit is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0404100C
Error Definition: Bad Circuit Creation error. The circuit data is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0404100D
Error Definition: SFC Data Error. The SFC data to be converted is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0404100E
Error Definition: List Data Error. The list data to be converted is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0404100F
Error Definition: Comment Data Error. The comment data to be converted is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04041010
Error Definition: Statement Error. The statement data to be converted is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04041011
Error Definition: Other Errors (Other errors generated during the conversion of Intermediate code to CAD code.)
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x040A0001
Error Definition: Insufficient Intermediate Code Storage Space. The space to store the data after conversion is insufficient.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x040A0002
Error Definition: The space to store addition SFC information is not sufficient
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x040A0003
Error Definition: Conversion Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x040A0004
Error Definition: Non-SFC Program Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x040A1001
Error Definition: Step Not Used / No Output error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x040A1002
Error Definition: Step No out of range error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x040A1003
Error Definition: Step Not Used / No Output error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x040A1004
Error Definition: Transition No out of range.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x040A1005
Error Definition: Maximum Number Exceeded error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x040A1006
Error Definition: Microcontroller Program space Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x040A1007
Error Definition: Non-SFC Program Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x040B0001
Error Definition: Insufficient Intermediate Code Storage Space. The space to store the data after conversion is insufficient.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x040B0002
Error Definition: Conversion Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x040B1001
Error Definition: Failed in creating Step Start position table
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x040B1002
Error Definition: Error Reading Step Information
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x040B1003
Error Definition: Step No. Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x040B1004
Error Definition: Failed in reading the output of operation/Transition condition intermediate code error.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x040B1005
Error Definition: Securing Internal Work Area Failed error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x040B1006
Error Definition: Error in setting the maximum value of X direction for character memory
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x040B1007
Error Definition: Insufficient Internal Work Area error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x040B1008
Error Definition: Stack Overflow, Abnormal Character Memory
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x040B1009
Error Definition: Insufficient No of Storage Blocks error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x040B100A
Error Definition: Non-SFC Program Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04050001
Error Definition: Abnormal Character String Specified error. Device character string specified is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04050002
Error Definition: Device Points Error. Device points are out of range
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04050003
Error Definition: Other Errors (The errors generated during the conversion of the Device Character String to Device Intermediate Code)
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04051001
Error Definition: Device Name Error. The classification specified for the device intermediate code is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04051002
Error Definition: Device Name Error. The classification specified for the extended specification device intermediate code is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04051003
Error Definition: Other Errors (The errors generated during the conversion of the Device Intermediate Code to Device Character String)
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04052001
Error Definition: Abnormal Character String Specified error. Device character string specified is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04052002
Error Definition: Device Points Error. Device points are out of range.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x04052003
Error Definition: Other Errors (The errors generated during the conversion of the Device Character String to Device Representation Code)
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04053001
Error Definition: Device Representation Error. The classification specified for the device intermediate code is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04053002
Error Definition: Device Representation Error. The classification specified for the extended specification device intermediate code is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04053003
Error Definition: Device Representation Error. The rectification part specified for the device is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04053004
Error Definition: Device Representation Error. The rectification part specified for the extended device is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04053005
Error Definition: Other Errors (The errors generated during the conversion of the Device Representation Code to Device Character String)
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04064001
Error Definition: Abnormal Device Intermediate Code error. The intermediate code for the device is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04064002
Error Definition: Other Errors (Other errors generated during the conversion of the Intermediate code for the Device to Device Name)
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04065001
Error Definition: Abnormal Device Name error. The classification specified for the intermediate code of the device is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04065002

Error Definition: Abnormal Device Name error. The classification for the intermediate code of the extended specification device is incorrect.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04065003

Error Definition: Other Errors (Other errors generated during the conversion of the device name to Intermediate code)

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04066001

Error Definition: Device Intermediate Code Error. The intermediate code for the device is incorrect.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04066002

Error Definition: Other Errors (Other errors generated during the conversion of the device intermediate code to device representation code.)

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04067001

Error Definition: Device Representation Error. The classification specified for the intermediate code of the device is incorrect.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04067002

Error Definition: Device Representation Error. The classification for the intermediate code of the extended specification device is incorrect.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04067003

Error Definition: Device Representation Error. The rectification part specified for the device is incorrect.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04067004

Error Definition: Device Representation Error. The rectification part specified for the extended device is incorrect.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x04067005
Error Definition: Other Errors (Other errors generated during the conversion of device representation code to the device intermediate code)
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04070001
Error Definition: Common Data Conversion Error. The input data of the device comment conversion is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04070002
Error Definition: Insufficient Common Data. The data to be converted is insufficient.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04070003
Error Definition: Insufficient Storage Area. The area where the conversion data is stored is insufficient.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04071001
Error Definition: Error in PLC Data Conversion. The input data of the device comment conversion is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04071002
Error Definition: Insufficient PLC Data error. The data to be converted is insufficient.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04071003
Error Definition: Insufficient Storage Area. The area where the conversion data is stored is insufficient.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04072001
Error Definition: Open Error. Failed in creating conversion object
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04072002
Error Definition: PLC Type Error. The specified PLC type does not exist.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04072003
Error Definition: Not Converted error. Converted object does not exist
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04072004
Error Definition: Input Data Error. The input data is incorrect
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04073001
Error Definition: Program Common Data Conversion Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04073002
Error Definition: Program Common Data Conversion Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04073101
Error Definition: Program PLC Data Conversion Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04074001
Error Definition: Common Data Parameter Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04074002
Error Definition: Network Parameter Common Data Error. The parameter block exists, but the data inside is not set.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04074101
Error Definition: Parameter PLC Data Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04074102
Error Definition: Network Parameter PLC Data Error. The parameter block exists, but the data inside is not set.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04074103
Error Definition: Offset Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04074201

Error Definition: Error in Specifying Network Type. The PLC specified does not support the network type.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04074202

Error Definition: Parameter Block Number Error. The Block corresponding to the parameter block number specified does not exist.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04074203

Error Definition: Parameter Block Content Error. It is different from the content supported by the specified.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04074204

Error Definition: Parameter Block Information Error. The specified block number does not exist.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04074205

Error Definition: Default Parameter Block is Abnormal. The specified block number does not exist.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04074301

Error Definition: Error in Conversion of the Common Parameter Block

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04074302

Error Definition: Error in Common Parameter Block No. 1001. The value of the RUN-PAUSE settings existence flag is incorrect.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04074303

Error Definition: Error in Common Parameter Block No. 1003

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04074304

Error Definition: Error in Common Parameter Block No. 1008

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x04074305
Error Definition: Error in Common Parameter Block No. 1100
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074306
Error Definition: Error in Common Parameter Block No. 2001 The device intermediate code specified does not exist.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074307
Error Definition: Error in Common Parameter Block No. 3000
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074308
Error Definition: Error in Common Parameter Block No. 3002
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074309
Error Definition: Error in Common Parameter Block No. 3004. The settings for the annunciator display mode is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407430A
Error Definition: Error in Common Parameter Block No. 4000. I/O Allotment Data is not created.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407430B
Error Definition: Error in Common Parameter Block No. 5000. The specified network is not supported.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407430C
Error Definition: Error in Common Parameter Block No. 5001. Valid unit No is not set while accessing other exchange.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407430D
Error Definition: Error in Common Parameter Block No. 5002
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x0407430E
Error Definition: Error in Common Parameter Block No. 5003
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407430F
Error Definition: Error in Common Parameter Block No. 5NM0
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074310
Error Definition: Error in Common Parameter Block No. 5NM1
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074311
Error Definition: Error in Common Parameter Block No. 5NM2
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074312
Error Definition: Error in Common Parameter Block No. 5NM3
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074313
Error Definition: Error in Common Parameter Block No. 6000
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074314
Error Definition: Error in Common Parameter Block No. FF18. Link parameter Capacity is not set.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074315
Error Definition: Error in Common Parameter Block No. FF25. Calculation circuit check is not set.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074316
Error Definition: Error in Common Parameter Block No. FF30. Sampling Trace Data is not created.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x04074317
Error Definition: Error in Common Parameter Block No. FF31. Status latch data is not created.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074318
Error Definition: Error in Common Parameter Block No. FF42. Timer processing points are not set.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074319
Error Definition: Error in Common Parameter Block No. FF30. Setting value device for specified extended timer does not exist.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407431A
Error Definition: Error in Common Parameter Block No. FF44
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407431B
Error Definition: Error in Common Parameter Block No. FF45
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407431C
Error Definition: Error in Common Parameter Block No. FF60. Terminal Settings are not set.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407431D
Error Definition: Error in Common Parameter Block No. FF70. User Release area is not set.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074401
Error Definition: Error in Conversion of PLC Parameter Block
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074402
Error Definition: Error in PLC Parameter Block No.1001
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x04074403
Error Definition: Error in PLC Parameter Block No.1003
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074404
Error Definition: Error in PLC Parameter Block No.1008
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074405
Error Definition: Error in PLC Parameter Block No.1100
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074406
Error Definition: Error in PLC Parameter Block No.2001
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074407
Error Definition: Error in PLC Parameter Block No.3000
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074408
Error Definition: Error in PLC Parameter Block No.3002
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074409
Error Definition: Error in PLC Parameter Block No.3004
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407440A
Error Definition: Error in PLC Parameter Block No.4000
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407440B
Error Definition: Error in PLC Parameter Block No.5000. The specified network type is not supported.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407440C
Error Definition: Error in PLC Parameter Block No.5001
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0407440D
Error Definition: Error in PLC Parameter Block No.5002
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0407440E
Error Definition: Error in PLC Parameter Block No.5003
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0407440F
Error Definition: Error in PLC Parameter Block No. 5NM0. The specified network type is not supported.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04074410
Error Definition: Error in PLC Parameter Block No. 5NM1
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04074411
Error Definition: Error in PLC Parameter Block No. 5NM2. The specified network type is not supported.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04074412
Error Definition: Error in PLC Parameter Block No. 5NM3
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04074413
Error Definition: Error in PLC Parameter Block No. 6000
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04074414
Error Definition: Error in PLC Parameter Block No. FF18
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04074415
Error Definition: Error in PLC Parameter Block No. FF25
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04074416
Error Definition: Error in PLC Parameter Block No. FF30
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x04074417
Error Definition: Error in PLC Parameter Block No. FF31
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074418
Error Definition: Error in PLC Parameter Block No. FF42
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04074419
Error Definition: Error in PLC Parameter Block No. FF43
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407441A
Error Definition: Error in PLC Parameter Block No. FF44
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407441B
Error Definition: Error in PLC Parameter Block No. FF45
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407441C
Error Definition: Error in PLC Parameter Block No. FF60
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407441D
Error Definition: Error in PLC Parameter Block No. FF70
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04075001
Error Definition: Common Data Conversion Error. Failed while converting the device memory settings portion.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04075002
Error Definition: Common Data Conversion Error. Failed while converting the device memory data portion.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04075003
Error Definition: Common Data Conversion Error. Device memory data portion did not exist.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04075101
Error Definition: PLC Data Conversion Error. Failed while converting the settings portion of the device memory.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04075102
Error Definition: PLC Data Conversion Error. Failed while converting the data portion of the device memory.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04076001
Error Definition: Common Data Conversion Error. Failed while converting the settings portion of the device comments.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04076002
Error Definition: Common Data Conversion Error. Failed while converting the data portion of the device comments.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04076101
Error Definition: PLC Data Conversion Error. Failed while converting the settings portion of the device comments.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04076102
Error Definition: PLC Data Conversion Error. Failed while converting the settings portion of the device comments.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04077001
Error Definition: Common Data Conversion Error. Failed during the conversion of sampling trace settings portion.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04077002
Error Definition: Common Data Conversion Error. Failed during the conversion of sampling trace data portion.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x04077101
Error Definition: PLC Data Conversion Error. Failed during the conversion of sampling trace settings portion.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04077102
Error Definition: PLC Data Conversion Error. Failed during the conversion of sampling trace data portion.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04078001
Error Definition: Common Data Conversion Error. Failed in the conversion of the status latch settings portion.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04078002
Error Definition: Common Data Conversion Error. Failed in the conversion of the status latch data portion.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04078101
Error Definition: PLC Data Conversion Error. Failed in the conversion of the status latch settings portion.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04078102
Error Definition: PLC Data Conversion Error. Failed in the conversion of the status latch data portion.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04079101
Error Definition: Failure history PLC Data Conversion error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407A101
Error Definition: File List PLC Data Conversion Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407B101
Error Definition: Error Information PLC Data Conversion Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x0407C001
Error Definition: Error in Conversion of Indirect Address to Device Name. The device name storage area is not secured.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407C002
Error Definition: Error in Conversion of Device Name to Indirect Address. Indirect Address storage area is not secured.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407C003
Error Definition: Error in Conversion of Indirect Address to Device Representation. The device representation storage area is not secured.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407C004
Error Definition: Error in Conversion of Device Representation to Indirect Address. Indirect Address storage area is not secured.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407C005
Error Definition: Error in Conversion of Indirect Address to Device Character String. Device Character String storage area is not secured.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407C006
Error Definition: Error in Conversion of Device Character String to Indirect Address. Indirect Address storage area is not secured.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407C007
Error Definition: Error in Conversion of Intermediate Code to Device Name. Device Name storage area is not secured.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407C008
Error Definition: Error in Conversion of Device Name to Intermediate Code. Intermediate Code storage area is not secured.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407C009
Error Definition: Error in Conversion of Intermediate Code to Device representation. Device Representation storage area is not secured.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x0407C00A
Error Definition: Error in Conversion of Device Representation to Intermediate Code. Intermediate Code storage area is not secured.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407C00B
Error Definition: Error in Conversion of Intermediate Code to Indirect Address. Indirect Address storage area is not secured.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407C00C
Error Definition: Error in Conversion of Indirect Address to Intermediate Code. Intermediate Code storage area is not secured.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407C00D
Error Definition: PLC Type Error The specified PLC type is not supported.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407C00E
Error Definition: Device Character String Error. The specified device is not supported.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407C00F
Error Definition: Device Character String Error. The specified device character string, type is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407C010
Error Definition: Device Error. The specified device is not supported by the specified PLC
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407C011
Error Definition: PLC Type Error. The specified PLC is not supported.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x0407C012
Error Definition: Device out of Range Error. For AnA system, a device out of AnA system range was specified.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0407D001

Error Definition: Common Data Conversion Error. Error in Conversion of SFC trace condition settings portion.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0407D002

Error Definition: Common Data Conversion Error. Error in Conversion of SFC trace condition data portion.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0407D101

Error Definition: PLC Data Conversion Error. Error in Conversion of SFC trace condition settings portion.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x0407D102

Error Definition: PLC Data Conversion Error. Error in Conversion of SFC trace condition data portion.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04080001

Error Definition: Intermediate Code classification out of range error. The intermediate code classification specified is out of range.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04080002

Error Definition: Extended specification Intermediate Code classification out of range error. The extended specification intermediate code specified is out of range.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04080003

Error Definition: Device Points check absent error. The device does not check the device points.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x04090001

Error Definition: GPP Project Error. The specified PLC type and GPP project type are not matching.

Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x04090002
Error Definition: File Type Error. The specified GPP project type and file type are not matching.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04090010
Error Definition: Insufficient GPP Data to be converted. There is no data to be converted. The data size specified is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04090011
Error Definition: Insufficient Storage Space for Converted Data. The space for storing converted data is insufficient.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04090012
Error Definition: Error in GPP Data to be converted. The GPP data to be converted is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04090110
Error Definition: Insufficient Data to be converted error. There is no data to be converted. The data size specified is insufficient.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04090111
Error Definition: Insufficient Storage Space for Converted Data error. The storage space for converted data is insufficient.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04090112
Error Definition: Error in data to be converted. The data to be converted is incorrect.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x04FFFFFF
Error Definition: Other Errors
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x10000001
Error Definition: No Command error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x10000002
Error Definition: Start of communication DLL of MX Component failed.
Action: Exit the program and restart the computer. Reinstall MX Component.
- Error Code:** 0x10000003
Error Definition: Open failed. (DiskDrive)
Action: Exit the program and restart the computer. Reinstall MX Component.
- Error Code:** 0x10000004
Error Definition: Duplex open error
Action: Exit the program and restart the computer.
- Error Code:** 0x10000005
Error Definition: File Access Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x10000006
Error Definition: Incorrect Folder Name error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x10000007
Error Definition: File Access Denied error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x10000008
Error Definition: Disk Full Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x10000009
Error Definition: File Delete Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x1000000A
Error Definition: Incorrect File Name error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x1000000C
Error Definition: Execution failed since another application or thread is making a request.
Action: Execute again after some time. Perform programming according to the multithread rules of COM and ActiveX. Exit the program and restart the computer.

Error Code:	0x1000000D
Error Definition:	Folder Creation Error
Action:	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
Error Code:	0x1000000E
Error Definition:	Folder/ File Type Error
Action:	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
Error Code:	0x1000000F
Error Definition:	Offset Address Error
Action:	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
Error Code:	0x10000010
Error Definition:	Request Cancel. Cancel Process has occurred.
Action:	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
Error Code:	0x10000011
Error Definition:	Memory securing error
Action:	Exit the program and restart the computer. Reinstall MX Component.
Error Code:	0x10000012
Error Definition:	Open not yet executed
Action:	Exit the program and restart the computer.
Error Code:	0x10000013
Error Definition:	Attach Not Executed error
Action:	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
Error Code:	0x10000014
Error Definition:	Object Invalid error
Action:	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
Error Code:	0x10000015
Error Definition:	Request Cancel Failed error
Action:	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
Error Code:	0x10000016
Error Definition:	Failed in Reading Status error
Action:	Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x10000017
Error Definition: The specified size (number of devices) is unauthorised.
Action: Check the number of points specified in the method. Exit the program and restart the computer.
- Error Code:** 0x10000018
Error Definition: There is no registered device.
Action: Exit the program and restart the computer.
- Error Code:** 0x10000019
Error Definition: Dataset Not Executed
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x1000001A
Error Definition: Read Not Executed error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x1000001B
Error Definition: Incorrect Create Flag error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x1000001C
Error Definition: Operation Over Access
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x1000001D
Error Definition: Redundant Device error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x1000001E
Error Definition: Registry search failed.
Action: Exit the program and restart the computer. Exit other programs and secure free memory area. Reinstall MX Component.
- Error Code:** 0x1000001F
Error Definition: File Type Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x10000020
Error Definition: Device Memory Type Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x10000021
Error Definition: Program Range Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x10000022
Error Definition: TEL Type Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x10000023
Error Definition: TEL Access Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x10000024
Error Definition: Cancel Flag Type Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x10000030
Error Definition: Multiple Device Registration Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x10000031
Error Definition: Device Not Registered error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x10000032
Error Definition: Specified device error
Action: Review the specified device data. Exit the program and restart the computer. Exit other programs and secure free memory area.
- Error Code:** 0x10000033
Error Definition: Specified device range error
Action: Review the specified device data. Exit the program and restart the computer. Exit other programs and secure free memory area.
- Error Code:** 0x10000034
Error Definition: File Write Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x10000040
Error Definition: Server start failed.
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

- Error Code:** 0x10000041
Error Definition: Server Stop Error. Failed while stopping the server
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x10000042
Error Definition: Server Started Twice error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x10000043
Error Definition: Server Not Started error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x10000044
Error Definition: Resource Timeout Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x10000045
Error Definition: Server Type Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x10000046
Error Definition: Failed to Access Server error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x10000047
Error Definition: Server Already Accessed error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x10000048
Error Definition: Failed in Simulator Startup
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x10000049
Error Definition: Failed in exiting Simulator
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0x1000004A
Error Definition: Simulator Not Started error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x100004B
Error Definition: Simulator Type Error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x100004C
Error Definition: Simulator Not Supported error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x100004D
Error Definition: Simulator Started Twice error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0x100004E
Error Definition: Shared Memory Not Started error
Action: Exit the program and restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.

Error Code: 0xF0000001
Error Definition: No-license error. The license is not given to the IBM-PC/AT compatible.
Action: Using the license FD, give the license to the IBM-PC/AT compatible.

Error Code: 0xF0000002
Error Definition: Set data read error. Reading of the set data of the logical station number failed.
Action: Specify the correct logical station number. Set the logical station number on the communication settings utility.

Error Code: 0xF0000003
Error Definition: Already open error. The Open method was executed in the open status.
Action: When changing the communication target CPU, execute the Open method after performing Close.

Error Code: 0xF0000004
Error Definition: Not yet open error. The Open method is not yet executed.
Action: After executing the Open method, execute the corresponding method.

Error Code: 0xF0000005
Error Definition: Initialisation error. Initialisation of the object possessed internally in MX Component failed.
Action: Exit the program and restart the computer. Reinstall MX Component.

Error Code: 0xF0000006
Error Definition: Memory securing error. Securing of MX Component internal memory failed.
Action: Exit the program and restart the computer. Exit other programs and secure free memory area.

Error Code: 0xF000007

Error Definition: Function non-support error. The method does not support.

Action: Can not use because the corresponding method is not supported.

Error Code: 0xF100001

Error Definition: Character code conversion error. Character code conversion (UNICODE ASCII code or ASCII code UNICODE) failed.

Action: 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.

Error Code: 0xF100002

Error Definition: First I/O number error. The first I/O number specified is an unauthorised value. A matching first I/O number does not exist.

Action: 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).

Error Code: 0xF100003

Error Definition: Buffer address error. The buffer address specified is an unauthorised value. The buffer address is outside the range.

Action: Check the value of the buffer address specified in the method.

Error Code: 0xF100004

Error Definition: Buffer read size error. As a result of buffer read, the specified size could not be acquired.

Action: Perform reopen processing. Review the system, e.g. PLC CPU, module setting and cable status. Retry. Exit the program.

Error Code: 0xF100005

Error Definition: 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.

Action: Check the size specified in the method.

Error Code: 0xF100006

Error Definition: Operation error. The operation specified for remote operation is an abnormal value.

Action: Check the operation specifying value specified in the method.

Error Code: 0xF100007

Error Definition: Clock data error. The clock data is abnormal.

Action: Check the clock data specified in the method. Set the correct clock data to the clock data of the PLC CPU.

Error Code: 0xF100008

Error Definition: 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.

Action: Register the device points between 1 and 20 in the EntryDeviceStaus method.

- Error Code:** 0xF100009
Error Definition: Monitored device data registration error
Action: After making deregistration in the FreeDeviceStatus method, execute the EntryDeviceStatus method again.
- Error Code:** 0xF100010
Error Definition: Device status monitor processing failed to start. Device status monitor processing failed to end.
Action: Start/end the device status monitor processing again in the EntryDeviceStatus method.
- Error Code:** 0xF100011
Error Definition: The VARIANT argument data type is wrong.
Action: Reexamine the data type specified for the VARIANT argument. • Check whether the array variable size is large enough. • Check whether the data type specified in the corresponding method has been set.
- Error Code:** 0xF100012
Error Definition: The device status monitoring time interval is a value outside the range 1 second to 1 hour (1 to 3600).
Action: Specify the device status monitoring time between 1 and 3600.
- Error Code:** 0xF100013
Error Definition: Already Connected error. Connect was executed again after it was executed for the same object.
Action: Execute the Connect method after executing the Disconnect method.
- Error Code:** 0xF100014
Error Definition: Invalid Telephone Number error. Characters other than "0123456789-*#" that are allowed for telephone numbers are included.
Action: Rectify the Telephone number and try to Connect again.
- Error Code:** 0xF100015
Error Definition: Exclusive Control Failure error. There was failure in the exclusive control process while executing the Connect and Disconnect method.
Action: 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.
- Error Code:** 0xF100016
Error Definition: While connecting to the telephone line error. The telephone line is connected to some other application, other than the one using MXComponent.
Action: Try Connecting again after disconnecting the application that is using the telephone line.

Error Code: 0xF1000017

Error Definition: 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.

Action: (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.

Error Code: 0xF1000018

Error Definition: 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).

Action: 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).)

Error Code: 0xF1000019

Error Definition: Not Closed error. Disconnect was executed while in Open state.

Action: Try Disconnect again after executing Close.

Error Code: 0xF100001A

Error Definition: 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.)

Action: 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.

Error Code: 0xF100001B

Error Definition: 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.

Action: Execute Disconnect for the object currently connected to the telephone line and execute Connect once again after the telephone line gets disconnected.

Error Code: 0xF100001C

Error Definition: 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.

Action: Execute Disconnect for all the Connected objects. Try Disconnect again for the object that actually performed the telephone line connection.

- Error Code:** 0xF10001D
Error Definition: Not Connected error. Open was executed before Connect Or, Disconnect was executed.
Action: Execute Open again after executing Connect. Or execute Disconnect again after executing Connect.
- Error Code:** 0xF10001E
Error Definition: Fatal Error.
Action: Exit the program. Restart the computer. Reinstall MX Component. If the problem cannot be solved, contact technical support.
- Error Code:** 0xF10001F
Error Definition: 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.
Action: Check the telephone number and the port number. Check the Connect way.
- Error Code:** 0xF200002
Error Definition: There is an error response from the target telephone. Normally caused by a communication error.
Action: 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.
- Error Code:** 0xF200003
Error Definition: 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.
Action: Retry. Check the communication device used at the other end.
- Error Code:** 0xF200004
Error Definition: There is no response from the modem. Causes can include abnormality in the modem, or the wrong telephone number.
Action: 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.
- Error Code:** 0xF200005
Error Definition: There are chances that the line is not disconnected.
Action: Check the line.
- Error Code:** 0xF200006
Error Definition: The PC modem did not receive the AT command. Causes can include specifying an invalid AT command, or an abnormality in the modem.
Action: Check the contents of the AT command. Check the status of the modem.

Error Code: 0xF2000007
Error Definition: Modem did not respond properly to the standard escape command.
Action: Check the modem. Confirm whether the value of the time-out is too small. (5000ms or more is recommended.)

Error Code: 0xF2000009
Error Definition: Modem does not respond properly to the line Disconnect command.
Action: Check the modem.

Error Code: 0xF200000A
Error Definition: 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.
Action: Check the Receive settings of the modem at the other end. Check if the other end is busy. Check the telephone number.

Error Code: 0xF200000B
Error Definition: Timeout reached for the call back receive waiting time.
Action: Increase the call back receive waiting time ActCallbackReceptionWaitingTimeOut and execute connect again.

Error Code: 0xF200000C
Error Definition: Password of A6TEL, Q6TEL, QJ71C24 units could not be resolved.
Action: Set the password to ActPassword property and execute the failed method again.

Error Code: 0xF2010001
Error Definition: 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.
Action: 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.

Error Code: 0xF2010002
Error Definition: QJ71C24 did not receive the specified connection method. Causes can include an incorrect Connection method or an incorrect telephone number for Call back.
Action: Check whether the settings of QJ71C24 and the MXComponent are matching.

Error Code: 0xF2010003
Error Definition: QJ71C24 does not permit the automatic connection (during fixed Call back or when the number is specified.)
Action: Check the settings of QJ71C24.

Error Code: 0xF2100005

Error Definition: There are chances that the line is not disconnected.

Action: 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.

Error Code: 0xF2100008

Error Definition: There was no response from the modem for the data sent from the PC.

Action: 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: 0xF2100006

Error Definition: Modem did not receive the startup command AT.

Action: 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.

Error Code: 0xF2100007

Error Definition: The PC modem does not respond to the Escape command.

Action: 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.

Error Code: 0xF21000

Error Definition: There is no response from the modem. Causes can be the following. Abnormality in the modem. Wrong telephone number.

Action: 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.

Error Code: 0xF21001

Error Definition: 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.

Action: 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.

Error Code: 0xF202

Error Definition: 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.

Action: 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".

B.4 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 section B.3]
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.

DCOM configuration

C

C.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.

This appendix will explain the changes that need to be made to the configuration on the client and server PCs when DCOM is used on Windows 2000 or Windows XP service pack 2.

C.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.

C.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.

C.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 (currently service pack 4 for Windows 2000, and service pack 2 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) From the 'Start' menu, select 'run'
 - 2) Enter 'cmd' in the 'open' box then click 'OK'
 - 3) 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.
 - 4) 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.

C.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).

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 (Windows 2000 or Windows XP) is used on both computers.

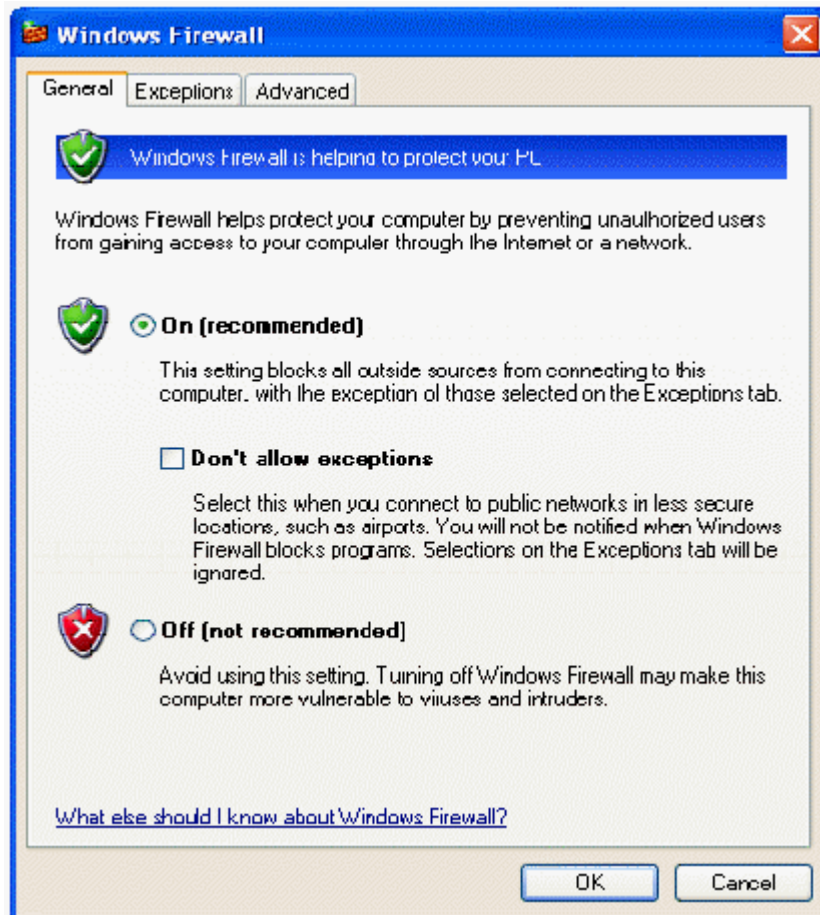
C.3 Configuring DCOM on Windows XP Service pack 2

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.

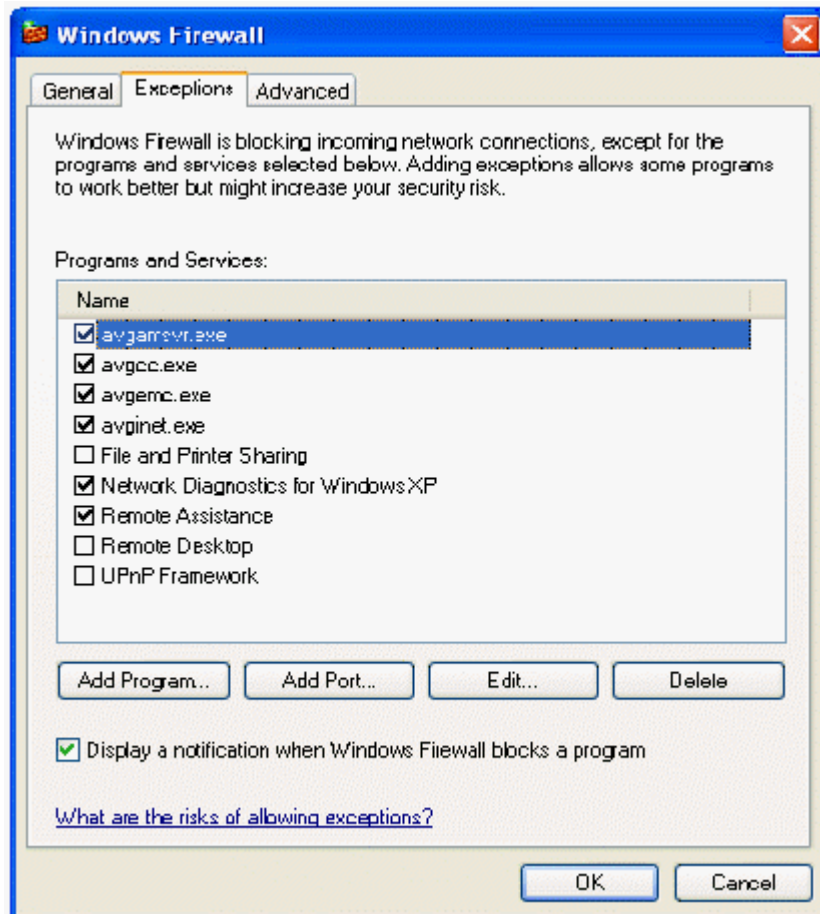
C.3.1 Configuring the Windows firewall

If you are using the Windows firewall provided with Windows XP service pack 2, 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.

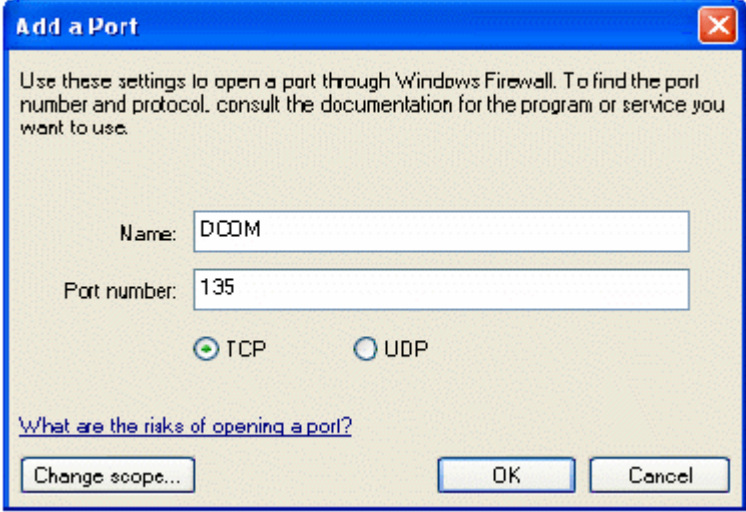
From the 'Start' menu, open 'Control panel'. If you are using 'Classic view', the 'Windows firewall' icon will already appear in the list, otherwise in 'Category view' it will be under the 'Network and internet connections' category. Double-click the icon to view the firewall settings screen (shown below).



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'.



Use these settings to open a port through Windows Firewall. To find the port number and protocol, consult the documentation for the program or service you want to use.

Name: DCOM

Port number: 135

TCP UDP

[What are the risks of opening a port?](#)

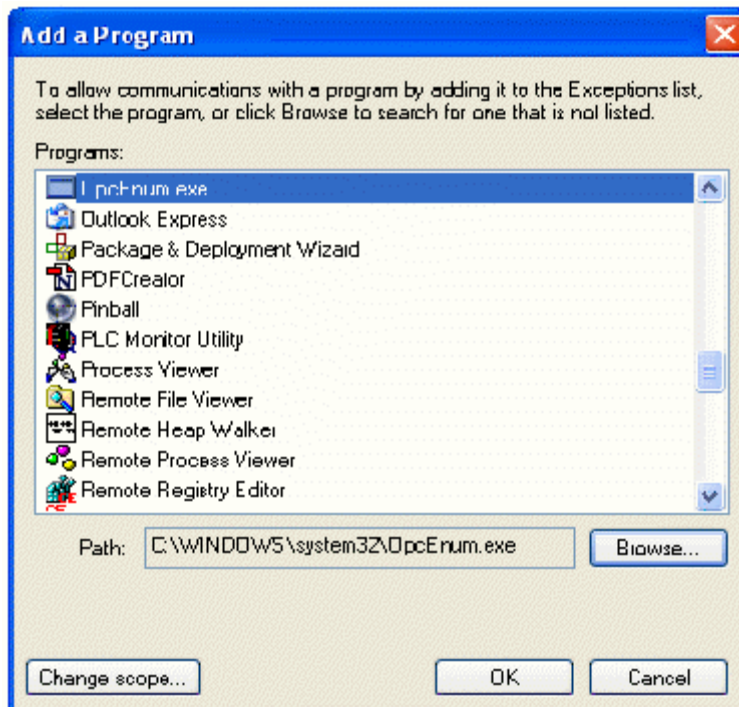
Change scope... OK Cancel

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:

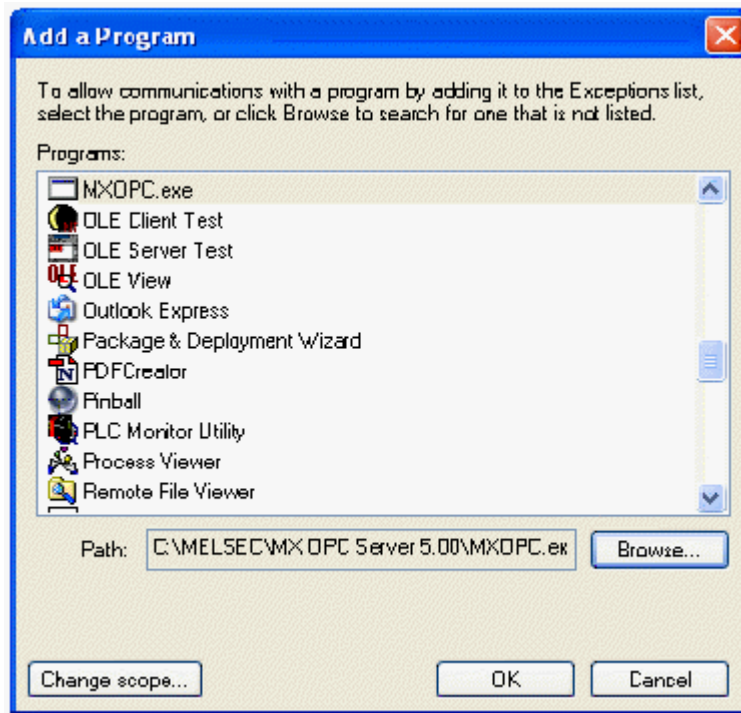
- The OPC server itself (on the server PC)
- The 'OPCenum' service (on the server PC)
- Each client application (on the client PC where it will run)

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 5.00\MXOPC.exe', but it will depend on the path selected during installation), as in the example screen below.



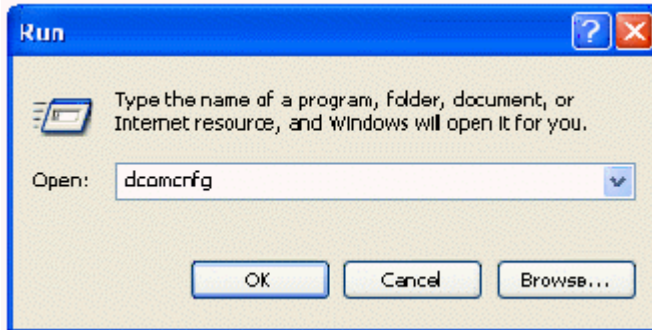
Repeat the 'Add program...' steps for the 'Microsoft Management Console' (this can usually be found in 'C:\Windows\System32\MMC.exe' but this will depend on your Windows installation).

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.

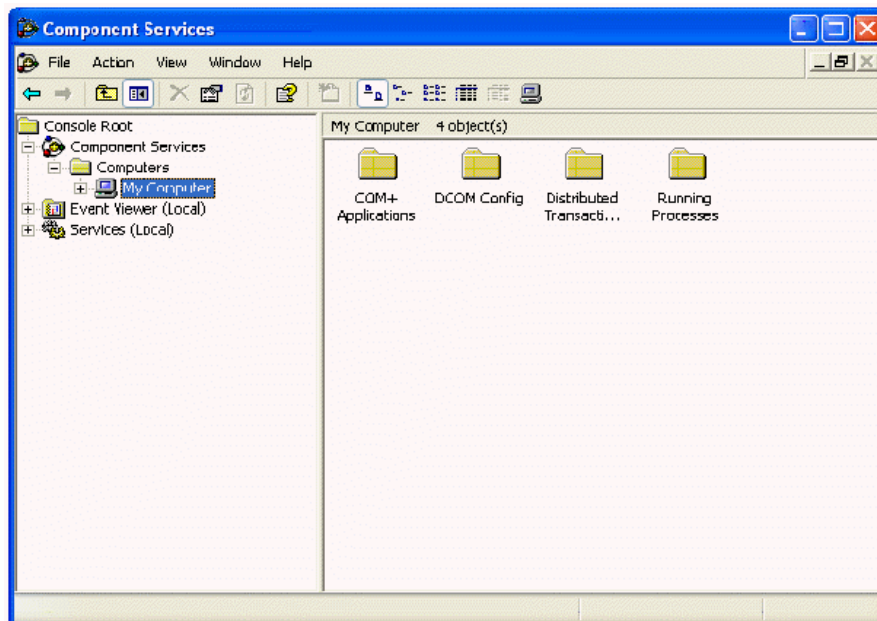
C.3.2 Configuring DCOM security

From the 'Start' menu, select 'Run' to show the window below.

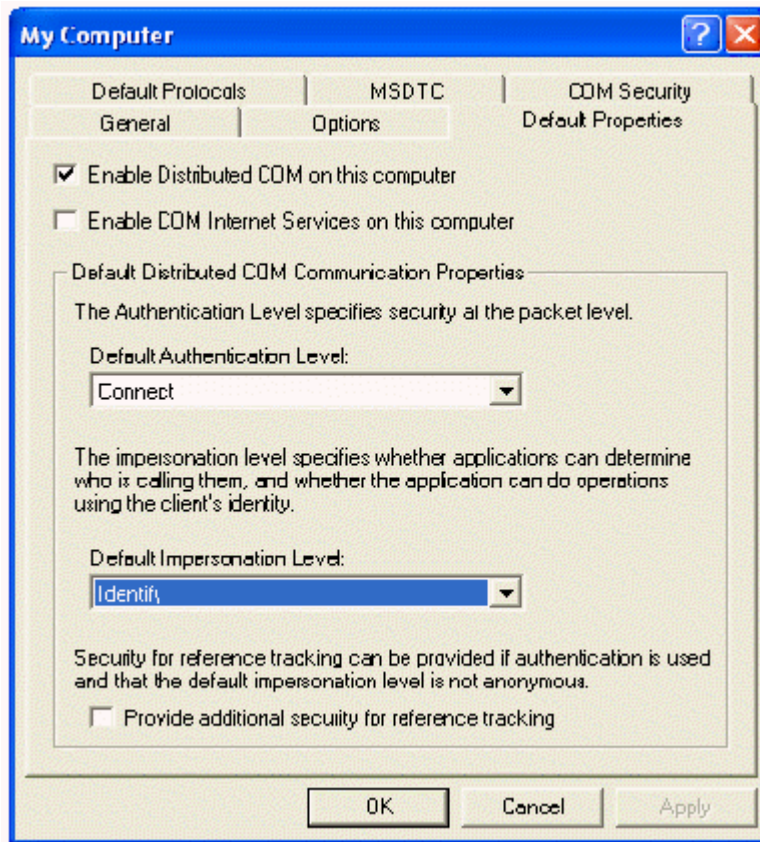


Enter 'dcomcnfg' and click 'OK' to start the DCOM configuration utility.

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:

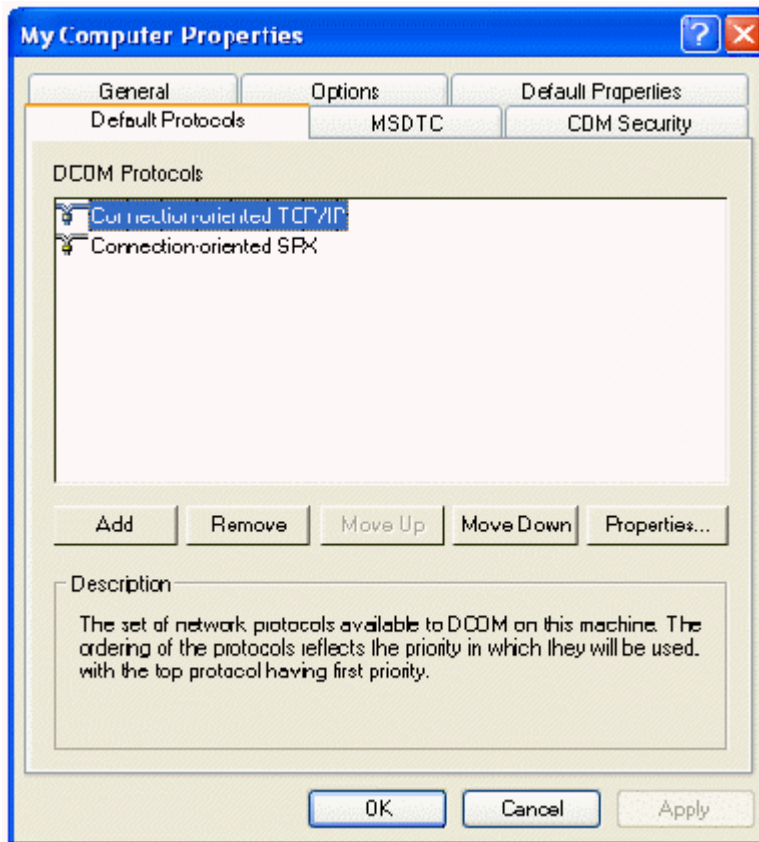


Make sure that 'Enable distributed COM on this computer' is checked.

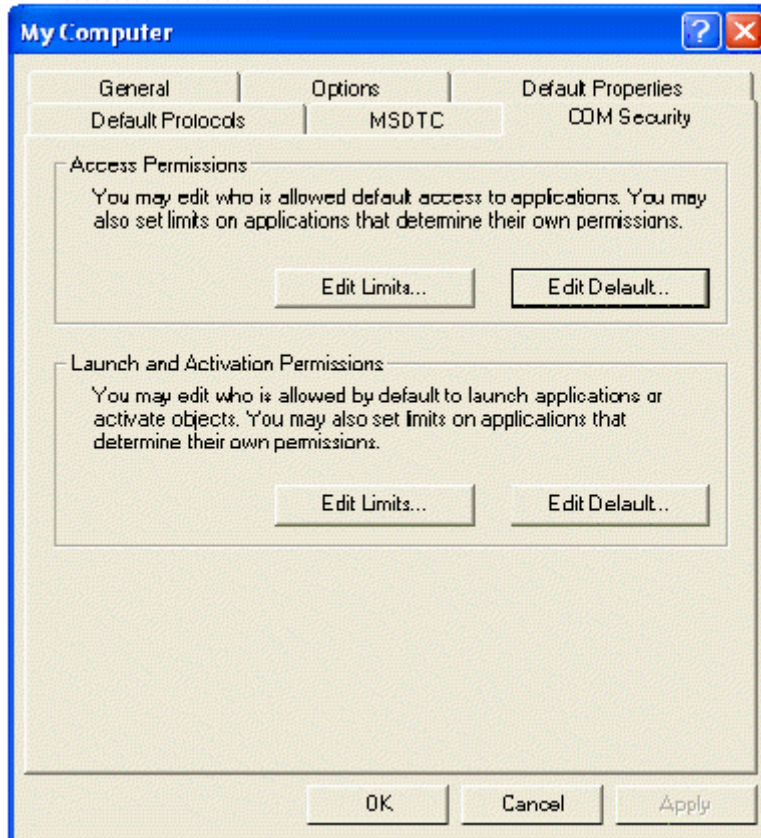
The 'Default authentication level' should normally be set to 'Connect', which checks the user's details once when a connection is first made. If there are problems with the connection, the minimal security option 'None' can be used (*Note: this requires that the 'Anonymous login' user is given access 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.

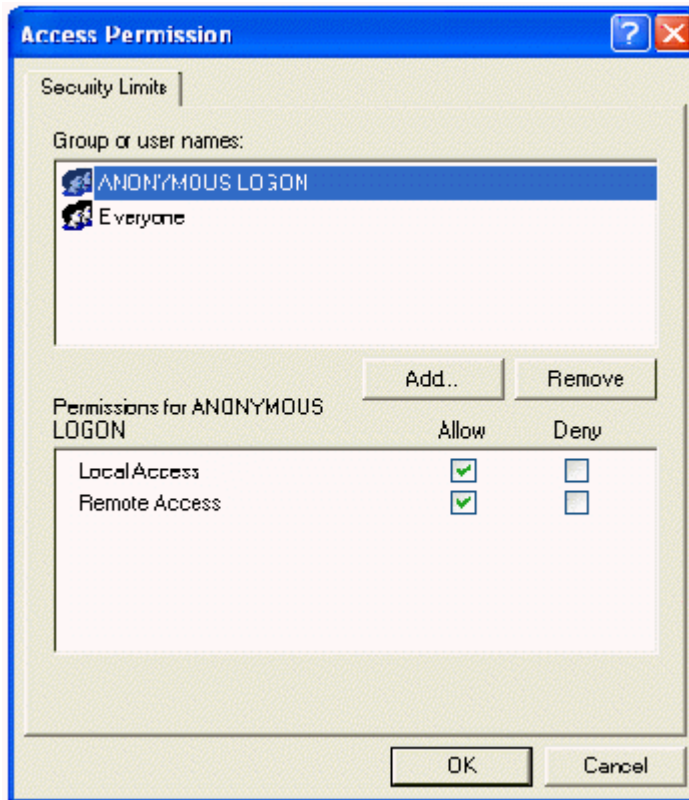


Click on the 'COM security' tab to show the dialog below:



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.

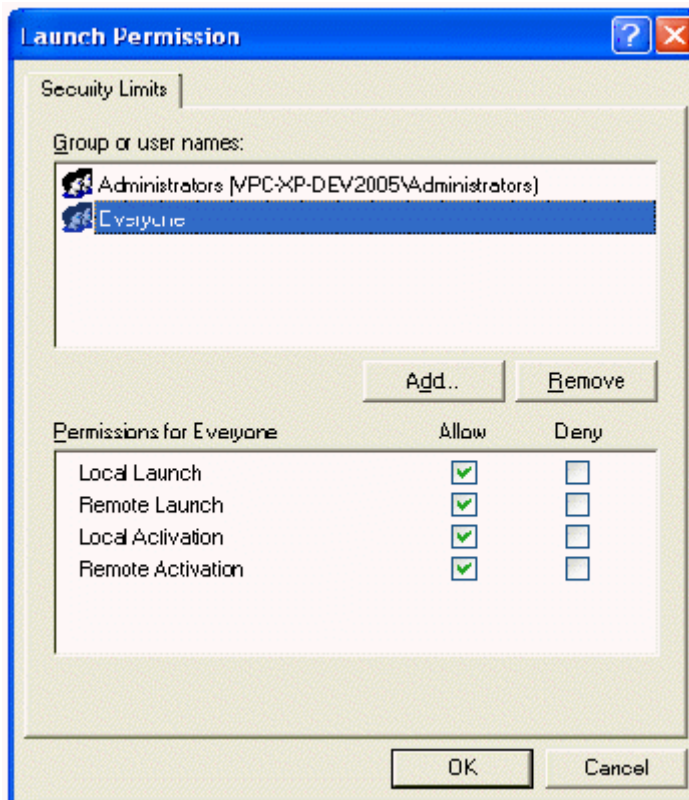


If 'ANONYMOUS LOGON' does not already appear in the list, add it – this is required to allow enumeration of OPC servers by remote PCs, and for applications where minimal security is being used due to other connection issues. If you do not need to fetch a list of available OPC servers from other PCs, 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:



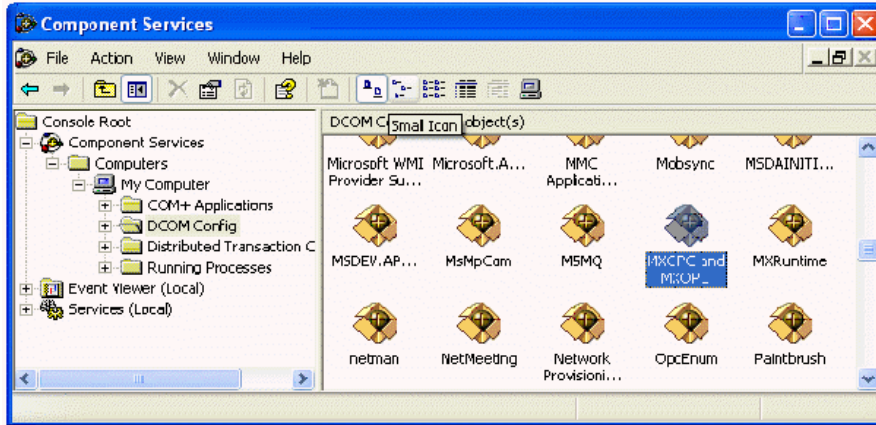
Ensure that all the 'Allow' checkboxes are set for each user or group that will access the server remotely. In the above picture, the 'Everyone' group has been enabled, but for a more secure configuration, you can add only the user(s) or group(s) that will actually use DCOM.

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.

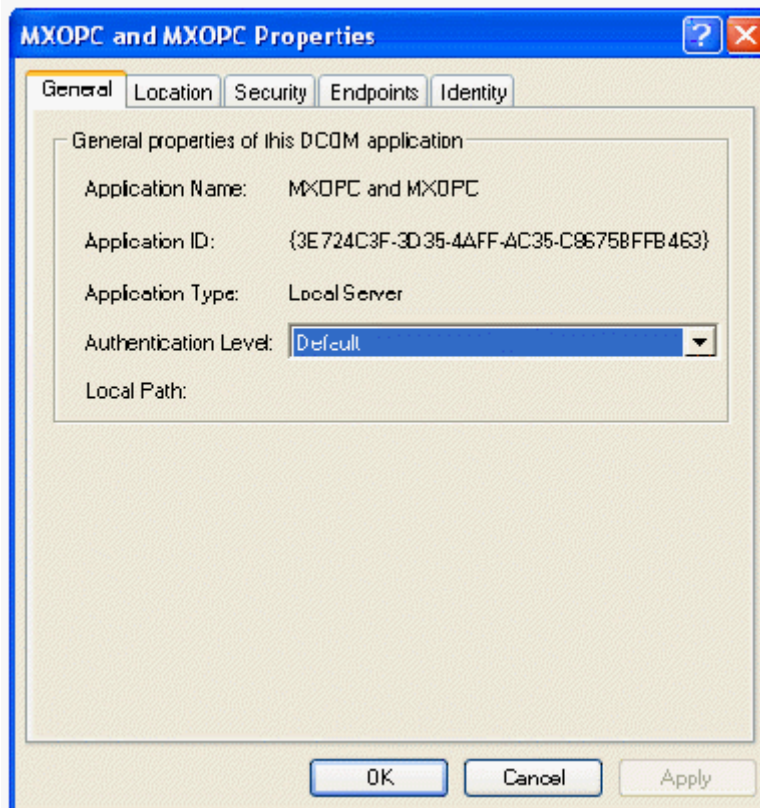
In the 'Edit defaults' setting for 'Access permissions', add the users or groups that will need OPC access, and ensure that the 'Allow' box is checked for both local and remote access. For a minimal security configuration, the 'Everyone' user can be given these rights.

Repeat this step for the 'Edit defaults' setting for 'Launch and activation permissions', ensuring that all of the 'allow' check boxes are set.

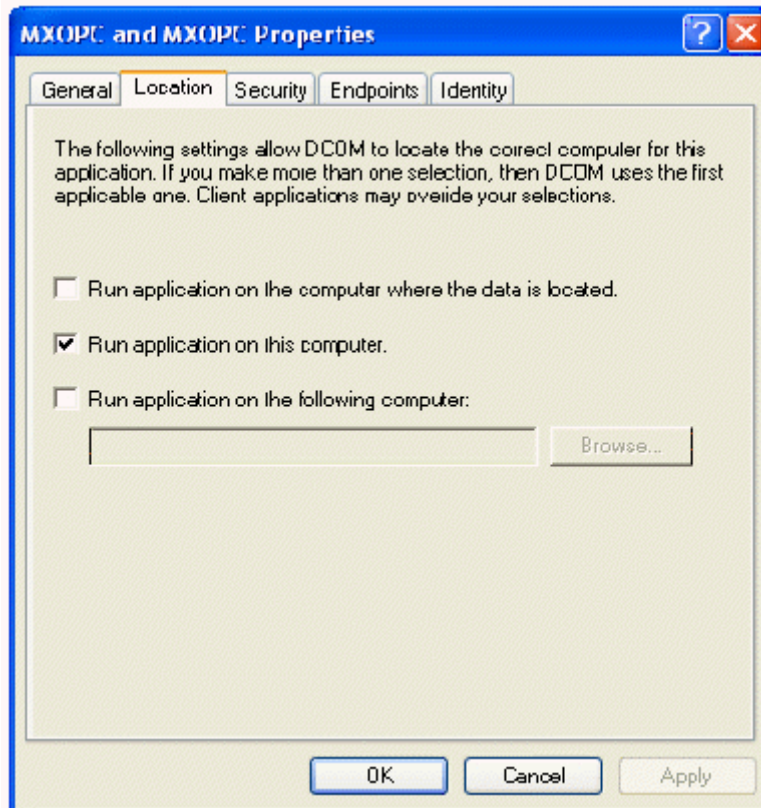
When you have finished, click 'OK' to return to the 'Component Services' dialog. For the server PC configuration (only), you will need to configure the security on the MX OPC server, MX Runtime and OPC enumeration services. To do this, double-click on the 'DCOM config' entry, and find 'MXOPC', 'MXRuntime' and 'OpcEnum' in the list, as shown in the picture below.



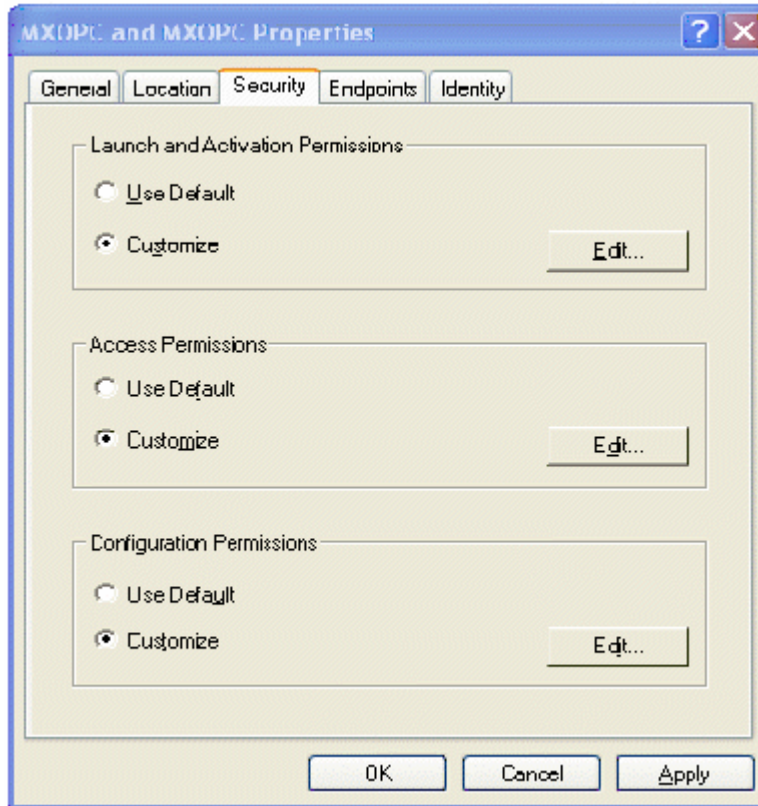
This step will need to be repeated for 'MXOPC', 'MXRuntime' and 'OpcEnum' in turn – 'MXOPC' will be used for this example. Right click on the 'MXOPC' icon and select 'properties'. The window below should appear; confirm that the authentication level setting is 'Default'.



Switch to the 'Location' tab as shown below. Confirm that the 'Run application on this computer' check box is selected.

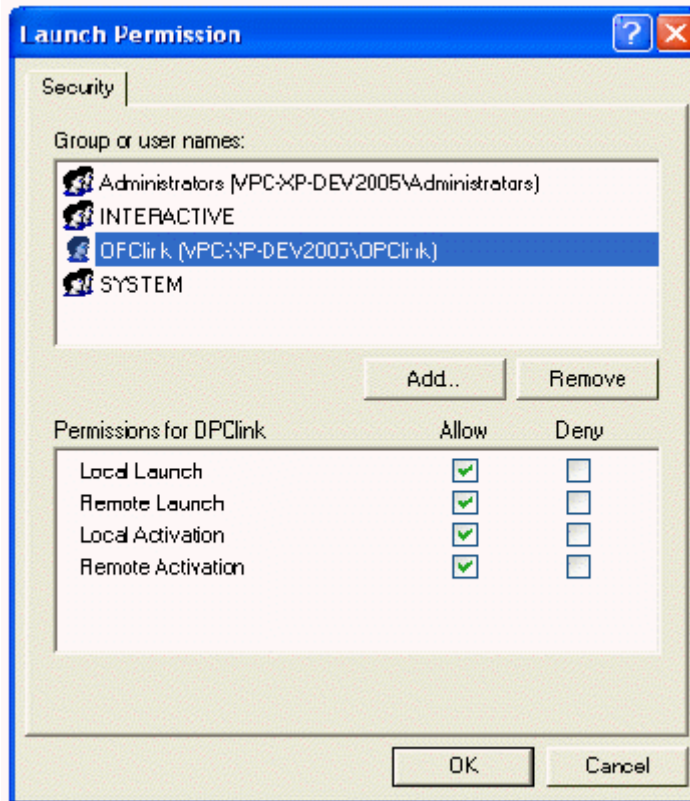


Now switch to the 'Security tab'.

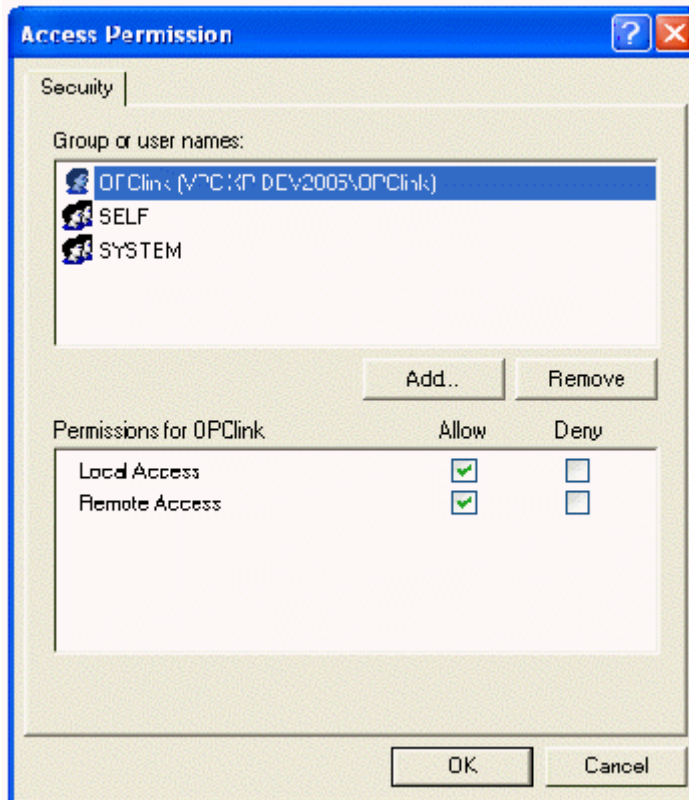


The 'Launch and activation permissions' and 'Access permissions' can both be left as 'Default' to use the default security settings configured earlier. Alternatively, they can be customized to allow more control over the users that can start and use each server. To do this, select 'customise' for each option and select 'Edit'.

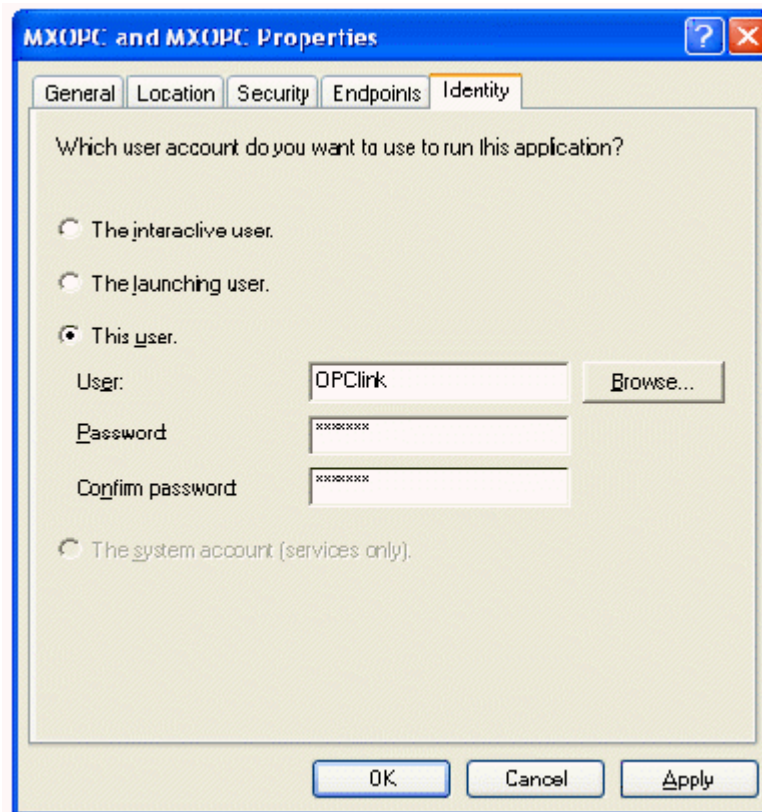
A sample 'Launch and activation permissions' edit window is shown below, in which a named user 'OPClink' has been added. As with the 'edit defaults' settings, all four 'allow' check boxes must be set for users requiring access through DCOM.



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.

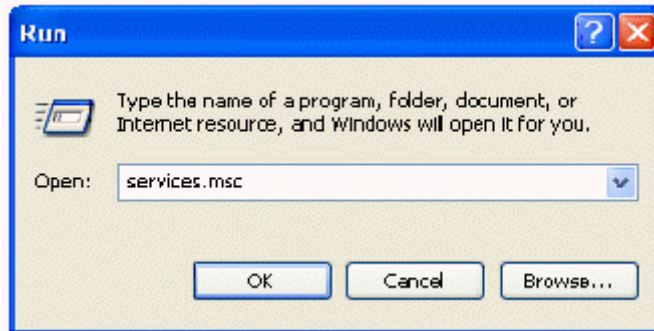
The system account, when the server is configured to run as a service.

Press 'OK' to return to the 'Component settings' page, which can then be closed.

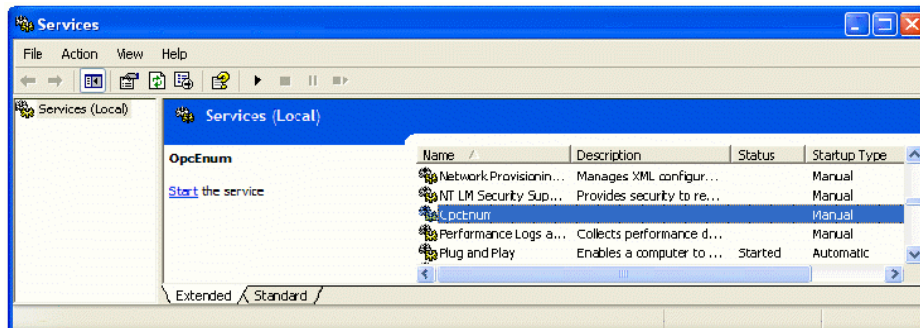
C.3.3 Configuring services to use a specific user account

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.

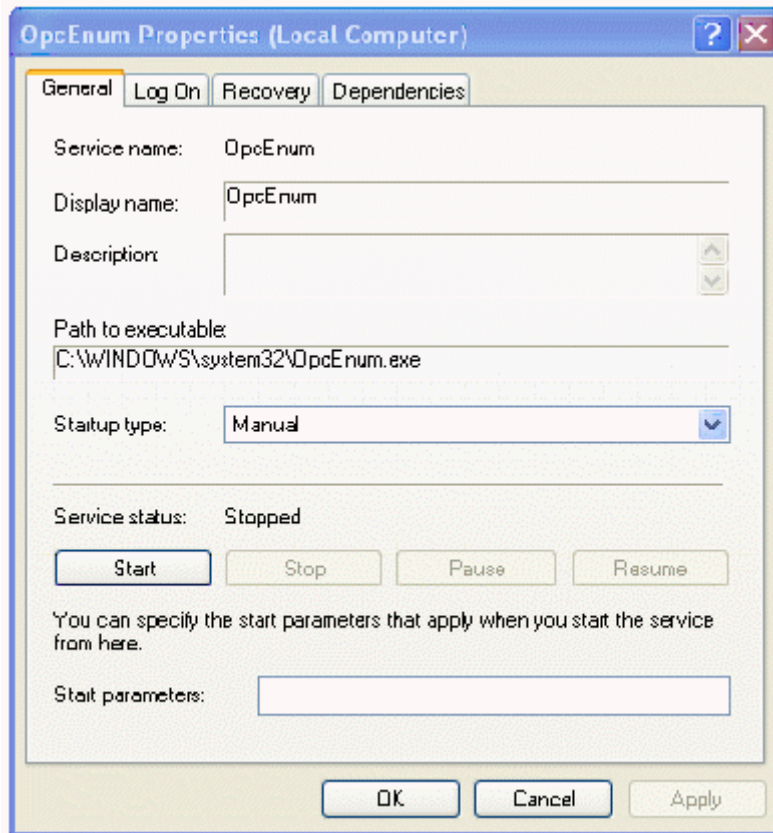
To open the 'services' page, select 'Start' -> 'Run', enter 'Services.msc' in the dialog that appears and click 'OK'.



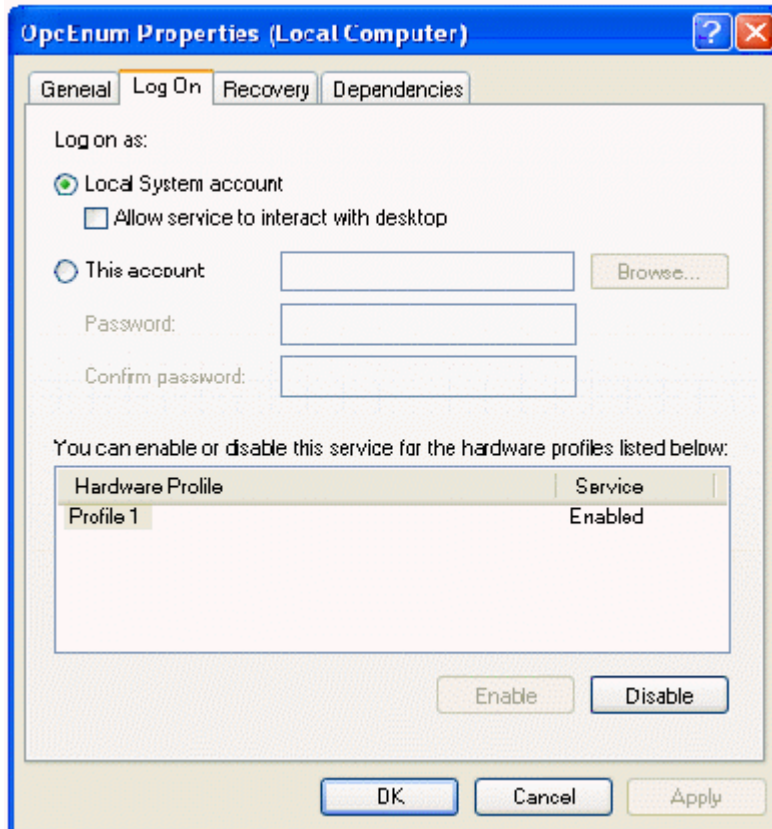
Find the service to be changed in the list. The sample screen below shows the OPC enumeration service.



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.

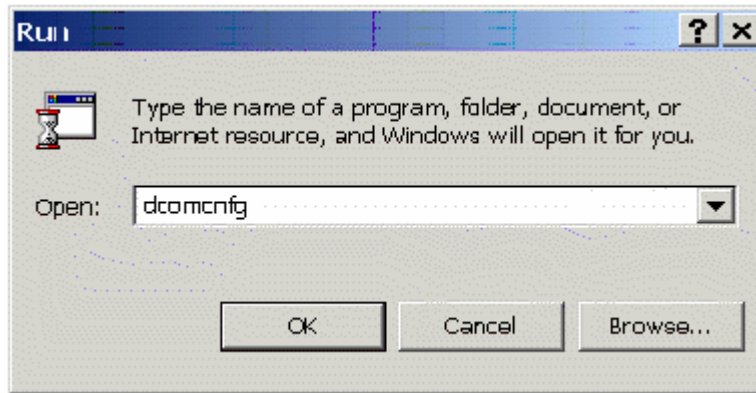
c.4 Configuring DCOM on Windows 2000

C.4.1 Firewalls

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 (see section C.3.1). Please refer to your firewall manual for instructions on how to do this.

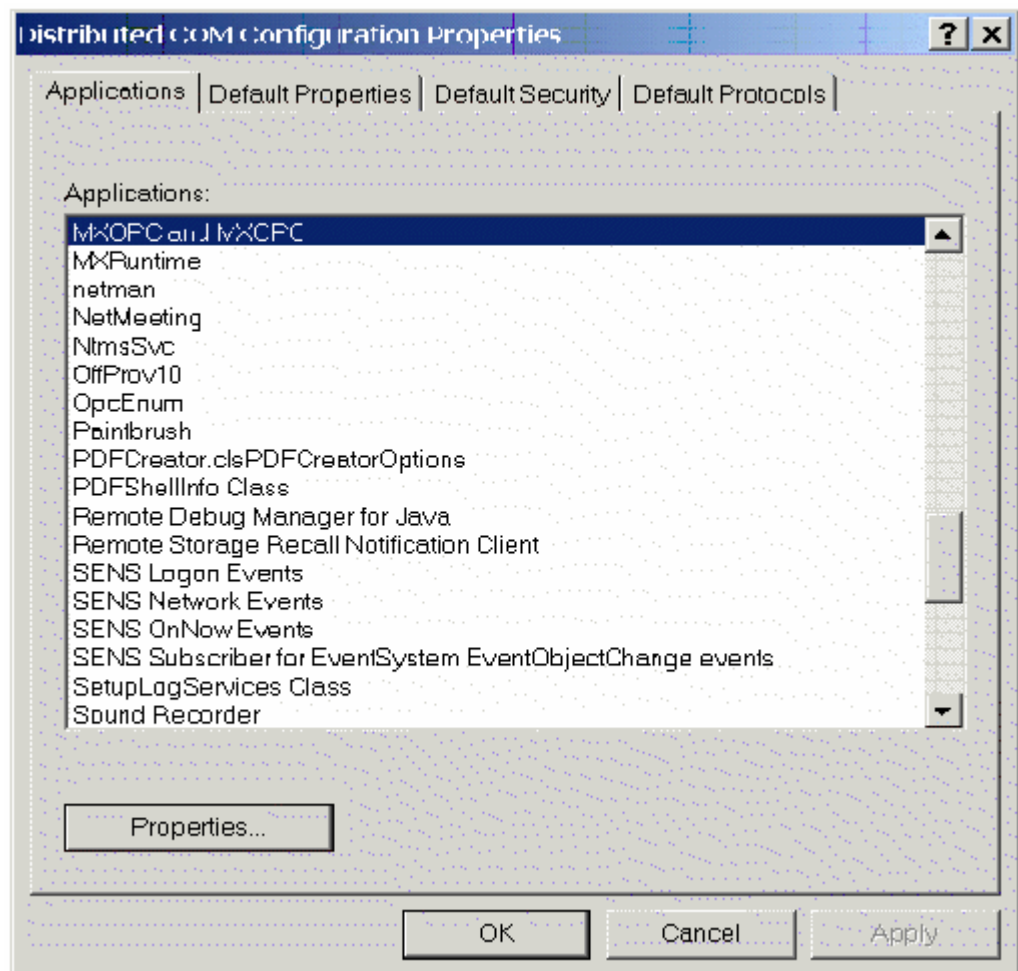
C.4.2 Configuring DCOM security

From the 'Start' menu, select 'Run' to show the window below.

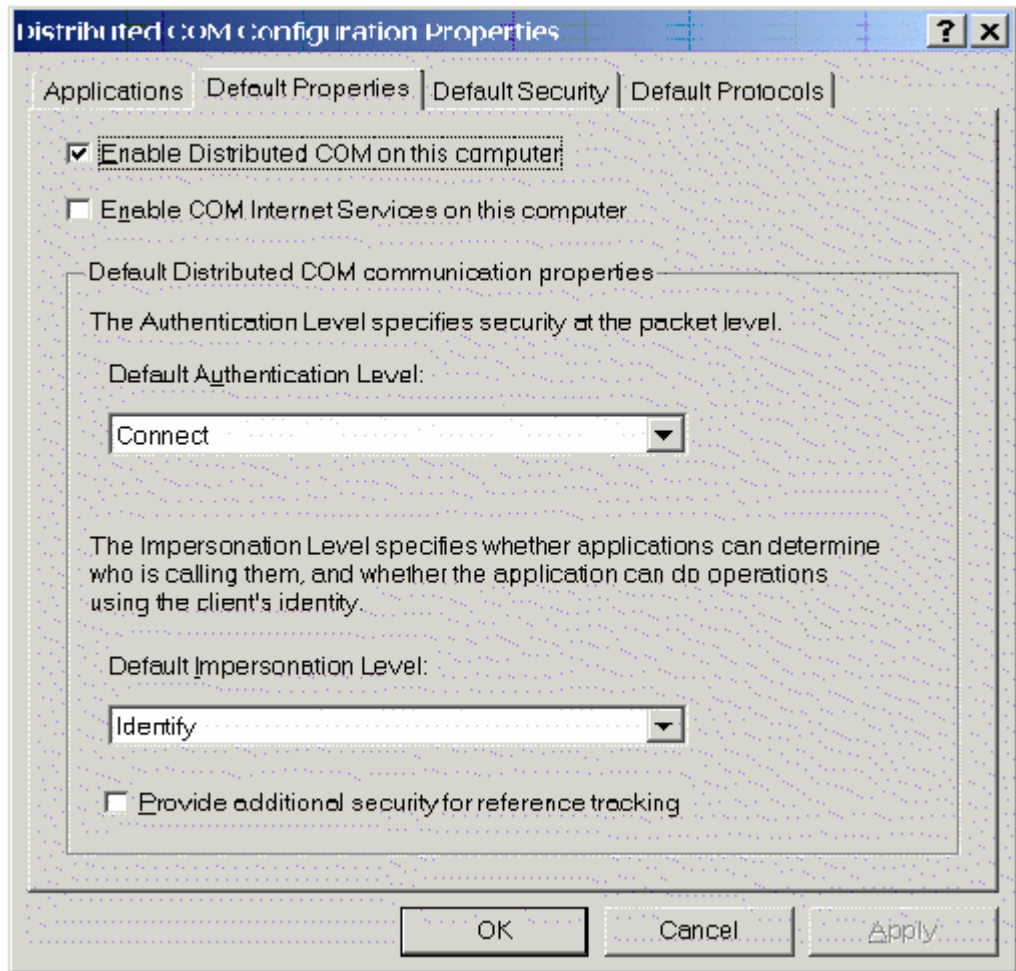


Enter 'dcomcnfg' and click 'OK' to start the DCOM configuration utility.

In the sample screen below, the 'MXOPC' and 'OPC enum' services are both visible.



Click on the 'Default properties' tab to show the window below.

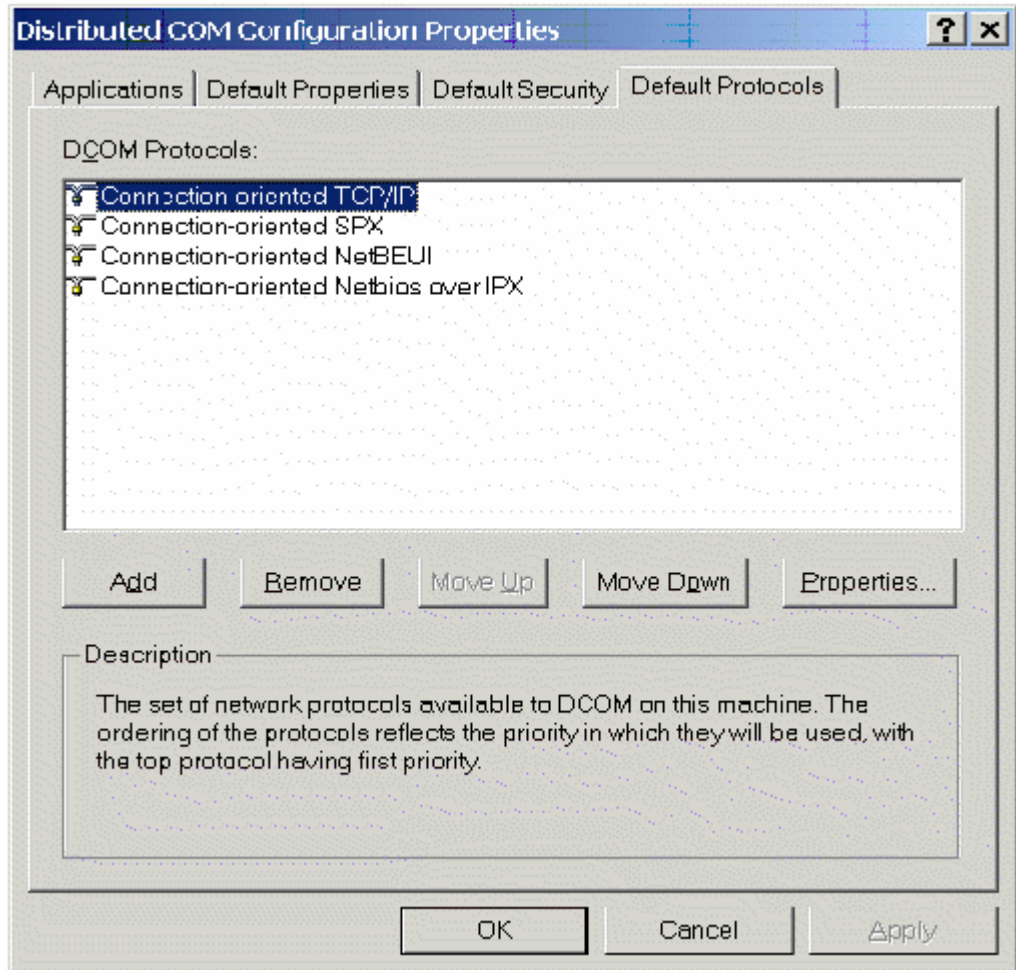


Make sure that 'Enable distributed COM on this computer' is checked.

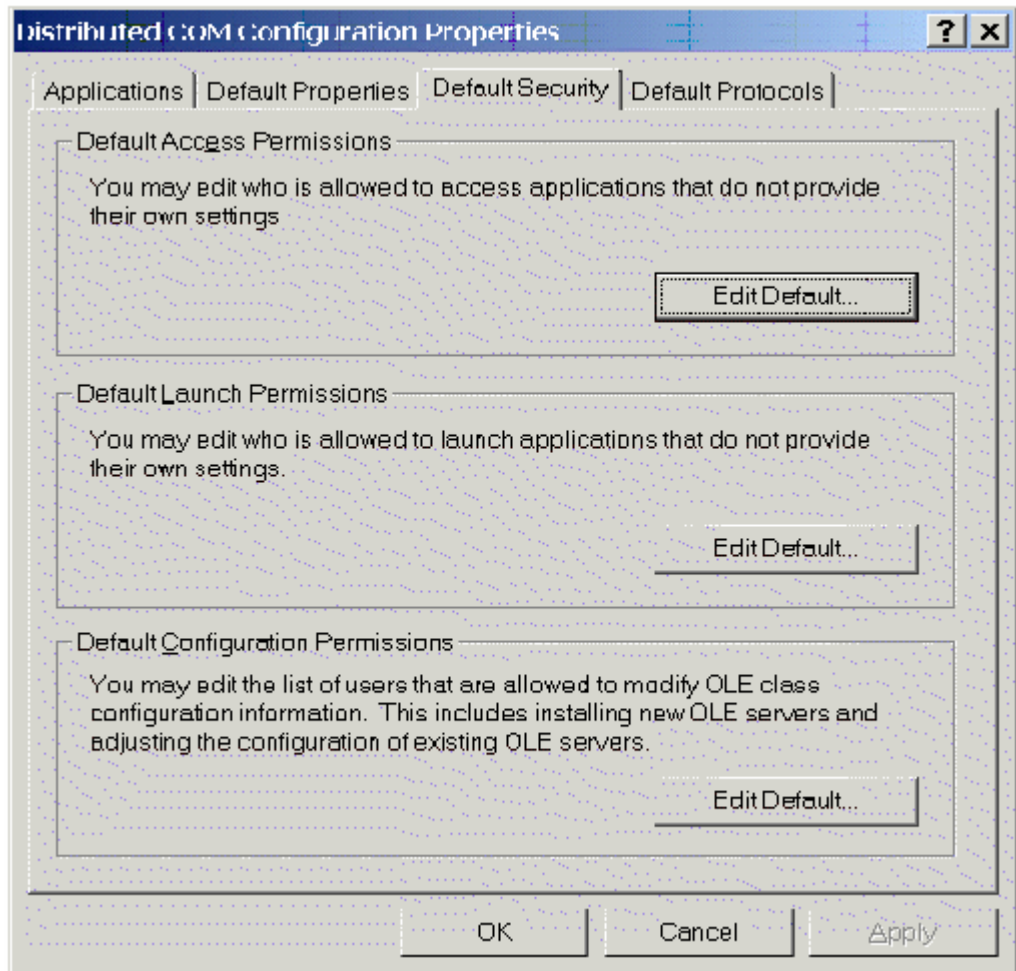
The 'Default authentication level' should normally be set to 'Connect', which checks the user's details once when a connection is first made. If there are problems with the connection, the minimal security option 'None' can be used (*Note: this requires that the 'Anonymous login' user is given access 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'.

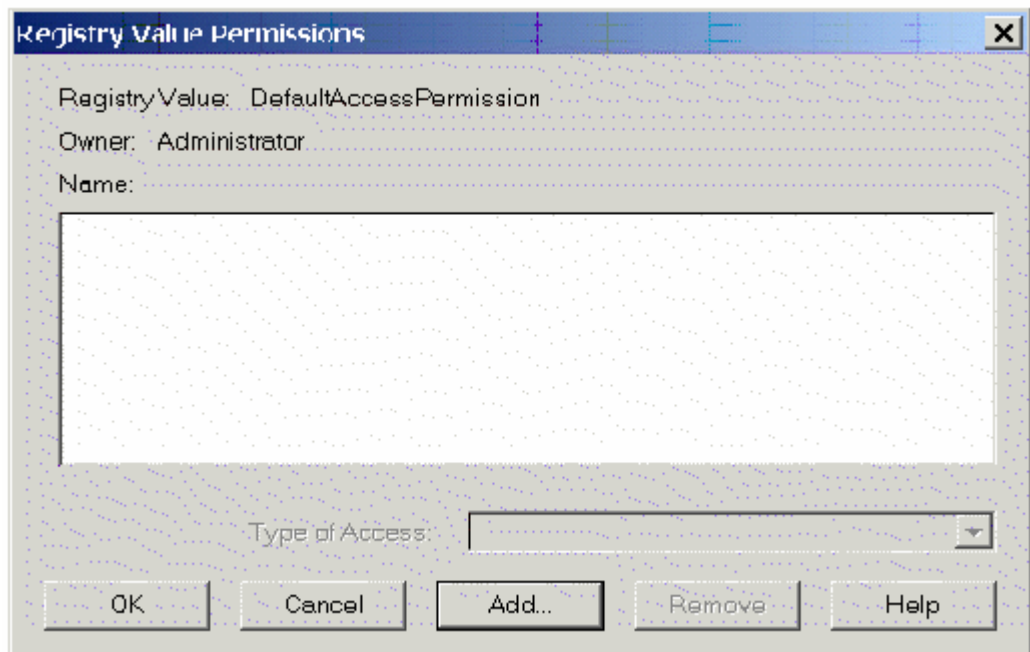
Switch to the 'Default protocols' tab. In this dialog (see example picture below), make sure that the 'Connection oriented TCP/IP' protocol appears at the top of the list. If you are not using the other protocols, remove them.



Click on the 'Default security' tab. The settings on this page determine the default access rights that each user is granted to COM objects.

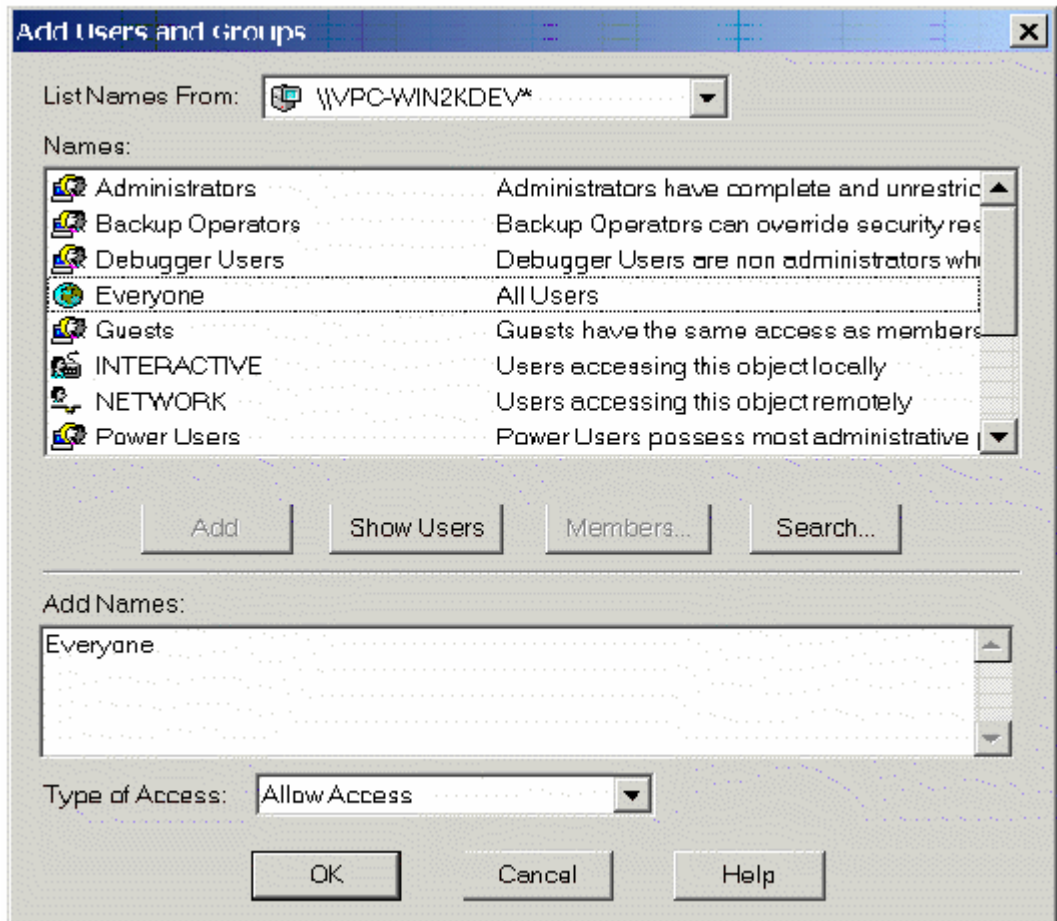


In the 'Default access permissions' section, click 'Edit default' and a dialog similar to the one below will appear:



The 'Add' button can be used to add users to the list.

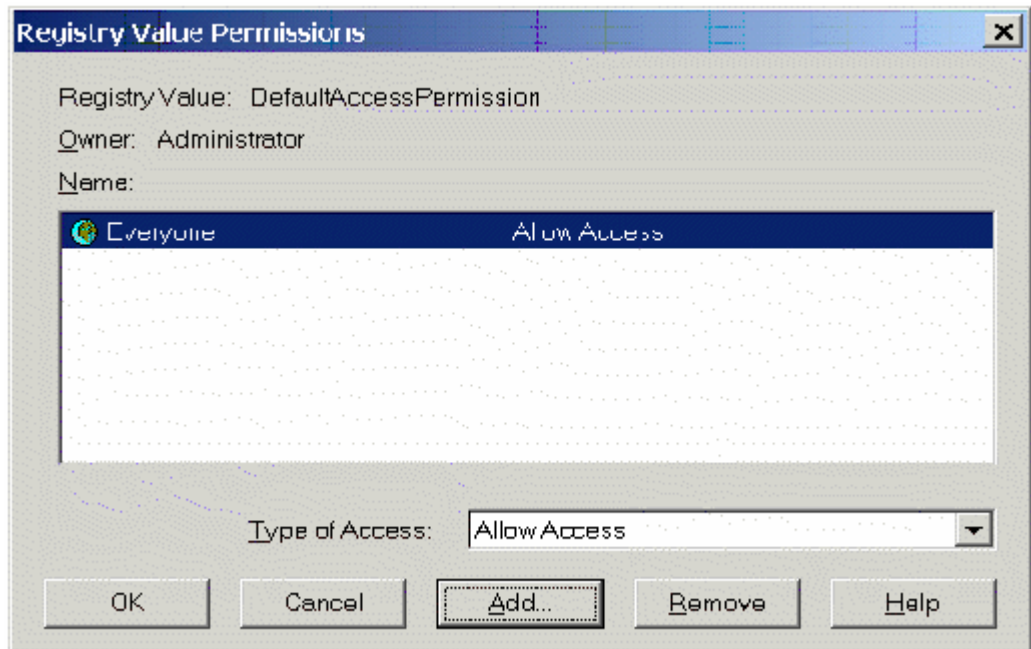
When it is clicked, the 'add users' dialog appears.



To add a user or group, select it from the list (using the 'Show users' button if the name is not already in the list), click 'Add', select 'Allow access' as the type of access, then click 'OK'.

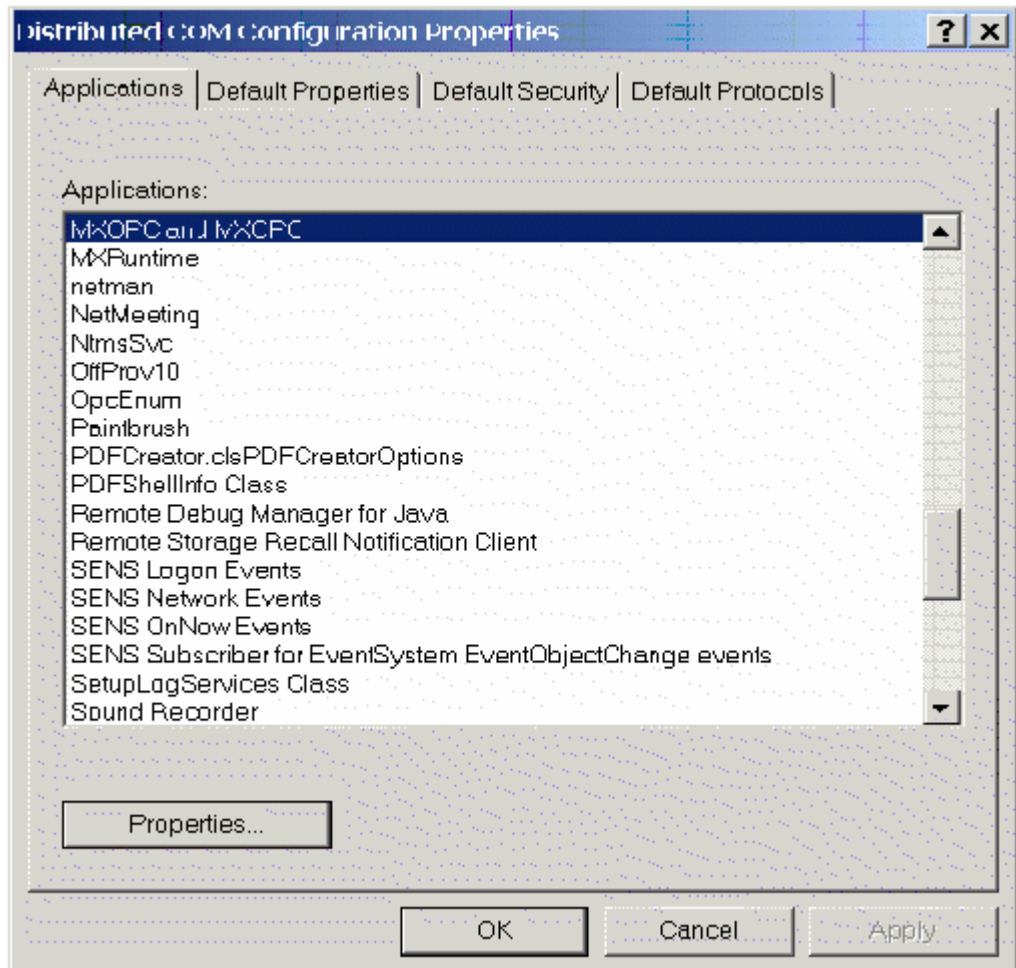
You may need to add the 'ANONYMOUS LOGON' user, by typing 'NT AUTHORITY\ANONYMOUS LOGON' into the 'Add names' window. This is sometimes required to allow enumeration of OPC servers by remote PCs, and for applications where minimal security is being used due to other connection issues. If you do not need 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 picture below, the group 'Everyone' has been added.

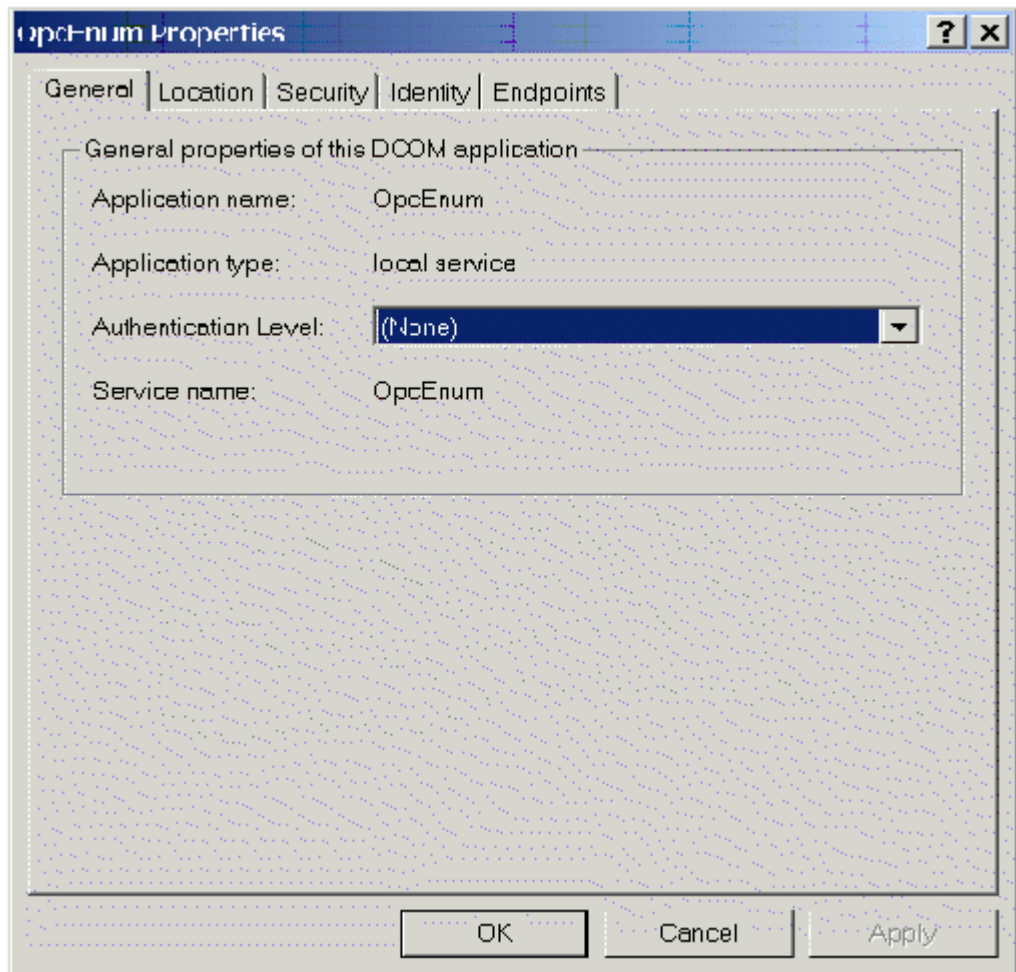


Click 'OK' to return to the 'Default security' screen, then click on the 'Edit defaults' button in the 'Default launch permissions' section and use the same method to add the users or groups requiring DCOM access.

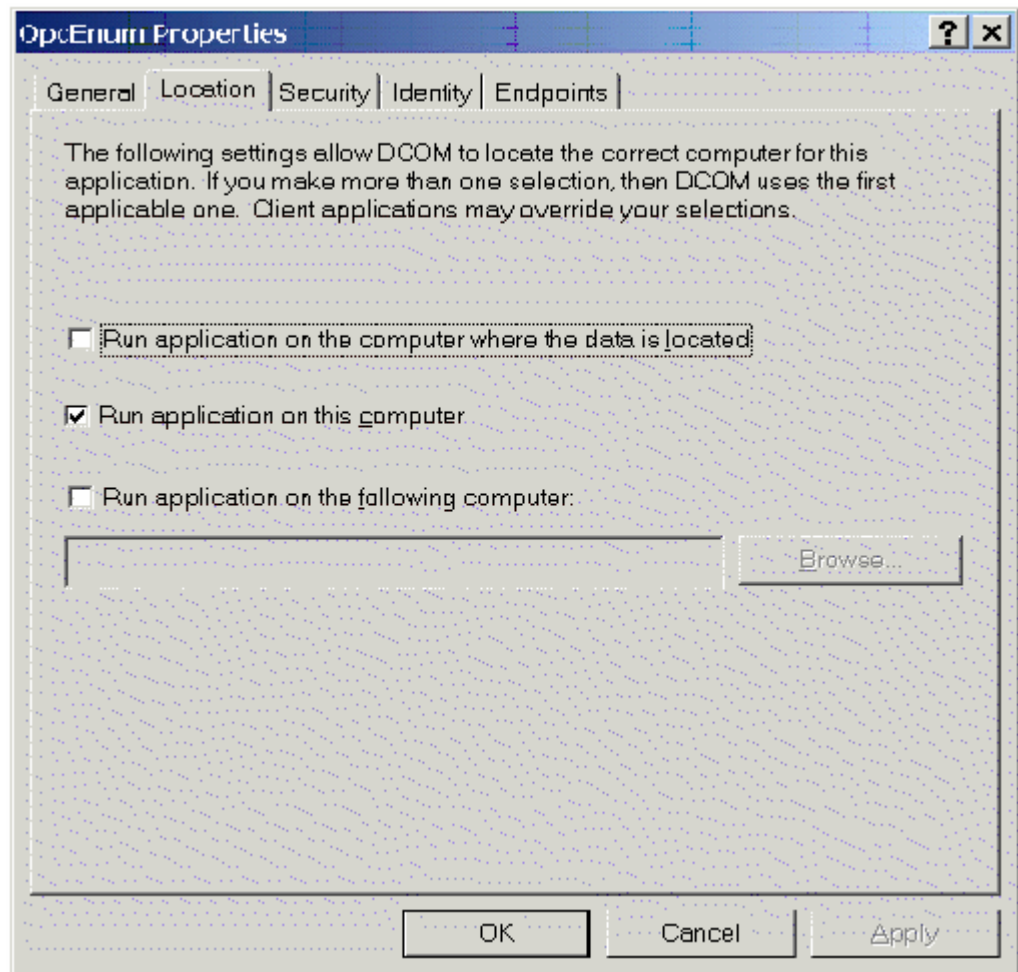
On the server PC, the DCOM server applications must be configured. Return to the 'Applications' tab (below) and find the entries 'OPCenum', 'MXRuntime' and 'MXOPC' in the list. The properties of these entries will need to be set separately, but for the following examples the 'OPCenum' service will be used.



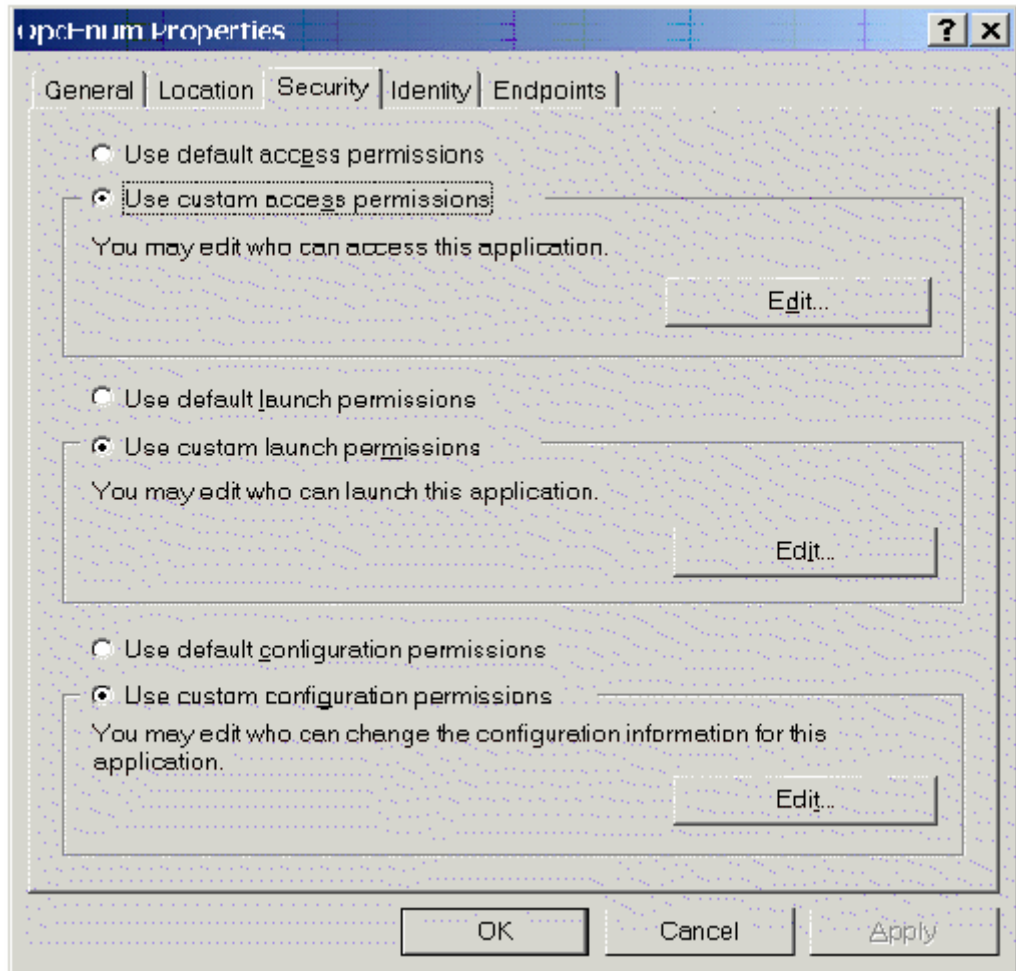
Select 'OPCenum' from the list, then click 'properties'. In the 'General' tab (below), the authentication setting can either be 'Default' or (for a lower security configuration) 'None'. The setting 'None' may be necessary for OPC server enumeration to work.



Switch to the 'Location' tab. As in the example picture, this setting should always be 'Run application on this computer'.

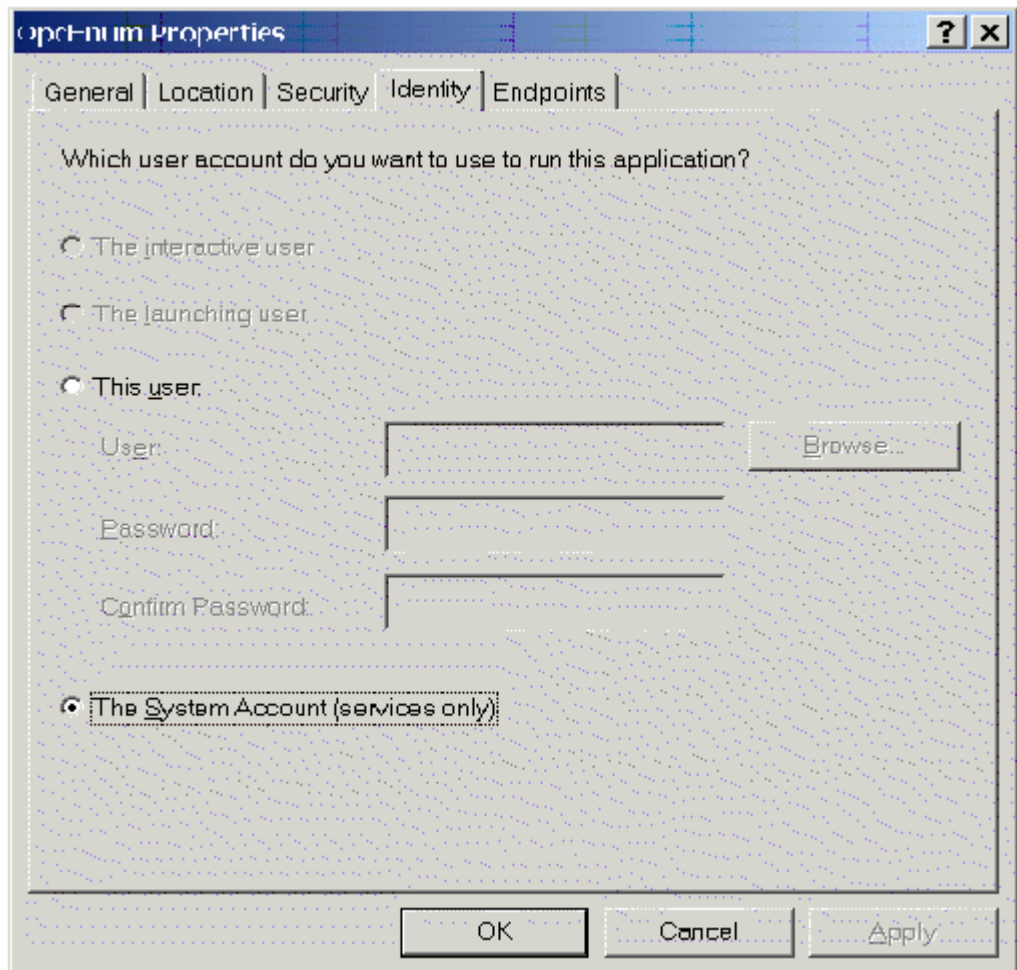


In the 'Security' tab (below), the 'Launch permissions' and 'Access permissions' can both be left as 'Use default' to use the default security settings configured earlier. Alternatively, they can be customized to allow more control over the users that can start and use each server. To do this, select 'Use custom access permissions' or 'Use custom launch permissions', then click the 'Edit' button and add any additional users or groups required. If you are using a lower-security configuration due to problems connecting, the 'Everyone' group can be given access rights.



The configuration permissions do not normally need to be changed.

Switch to the 'Identity' tab.



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.

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.

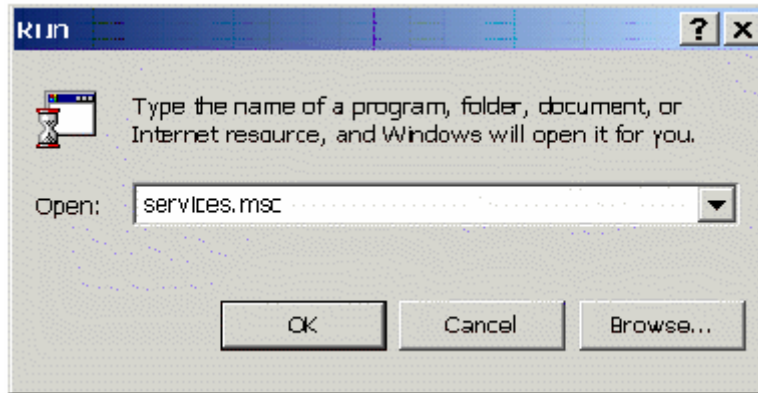
The system account, when the server is configured to run as a service.

Click 'OK' to return to the 'Distributed COM configuration properties' page. After changing the properties of any other DCOM objects required (such as 'MXOPC' and 'MXRuntime') in the same way, this dialog can then be closed.

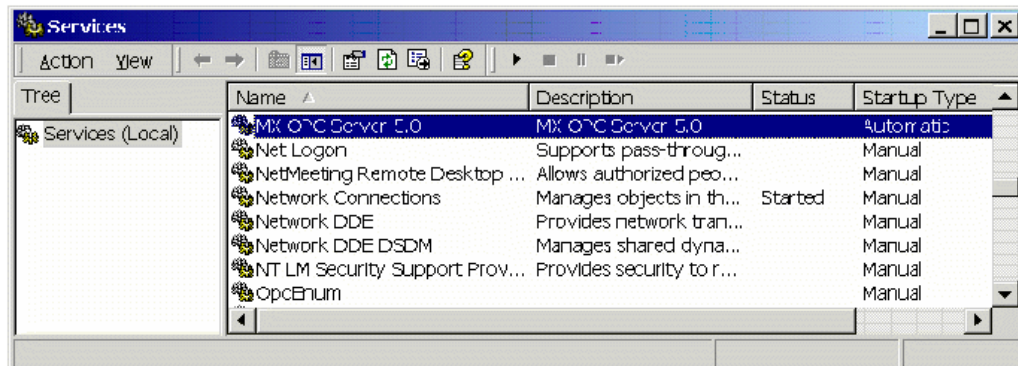
C.4.3 Configuring services to use a specific user account

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.

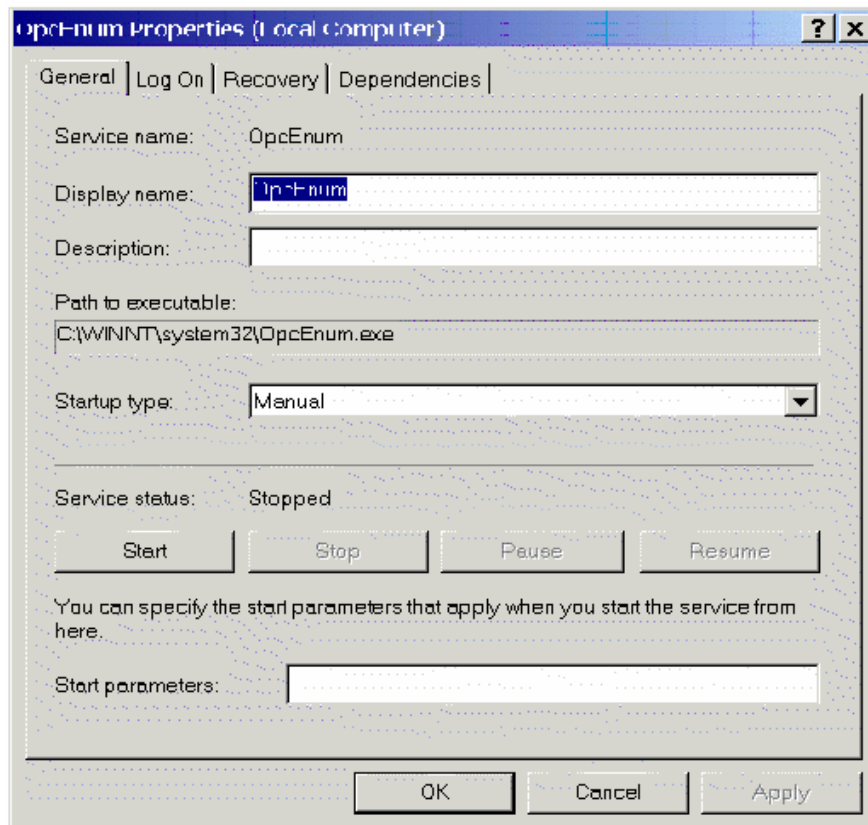
To open the 'services' page, select 'Start' -> 'Run', enter 'Services.msc' in the dialog that appears and click 'OK'.



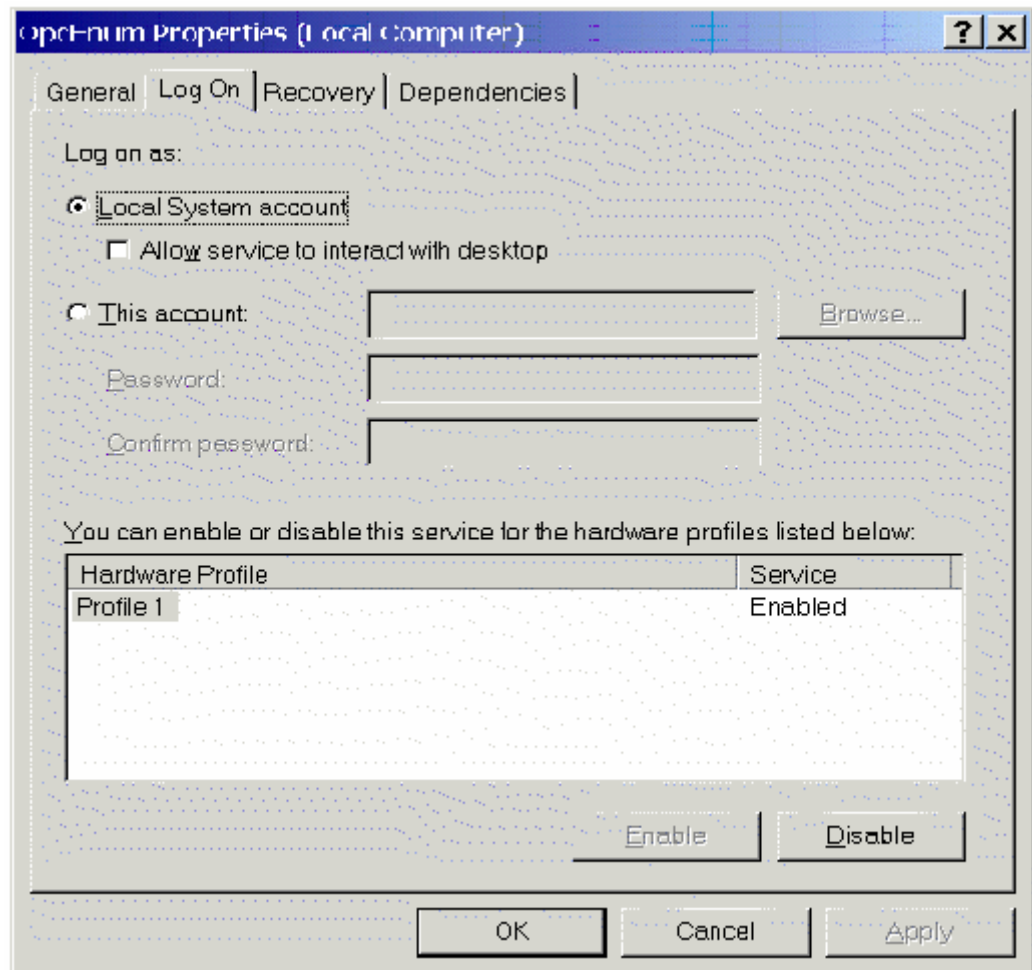
Find the service to be changed in the list. In the sample screen below, the 'MX OPC server 5.0' and 'OPC enum' services are both visible.



Right click on the service to be changed, and select 'Properties' from the pop-up menu that appears. The service configuration window is shown below for the 'OPC enum' service.



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



Finally, select 'OK' then close the services window.