Yokogawa MX Ethernet Driver Help

Table of Contents

Table of Contents 2

Yokogawa MX Ethernet Driver Help 3

Overview 3

Device Setup 4

Optimizing Your Ethernet Communications 6

Data Types Description 7

MX100 Addressing 10

Error Descriptions 12

Address Validation 12

Address '<address>' is out of range for the specified device or register 12

Data Type '<type>' is not valid for device address '<address>' 12

Device address '<address>' contains a syntax error 12

Device address '<address>' is Read Only 13

Missing address 13

Device Status Messages 13

Detected Yokogawa MX 100 Unit Style '<version>' with '<number of modules>' modules and a total of '<number of channels>' channels on device '<device name>' 13

Device '<device name>' has no FIFO data for channel (Tag '<address>') 13

Device '<device name>' is not responding 14

Device '<device name>' responded with error '<error code>' (Tag '<address>', Size '<size>') 14

Unable to write to '<address>' on device '<device name>' 14

Driver Error Messages 14

Winsock initialization failed (OS Error = n) 14

Winsock V1.1 or higher must be installed to use the Yokogawa MX Ethernet device driver 15

Index 16

Yokogawa MX Ethernet Driver Help

Help version 1.021

CONTENTS

Overview
What is the Yokogawa MX Ethernet Driver?

Device Setup
How do I configure a device for use with this driver?

Optimizing Your Ethernet Communications
How do I get the best performance from the Yokogawa MX Ethernet driver?

Data Types Description
What data types does this driver support?

Address Descriptions
How do I address a data location on a Yokogawa MX Ethernet driver?

Automatic Tag Database Generation
How can I easily configure tags for the Yokogawa MX Ethernet driver?

Error Descriptions
What error messages does the Yokogawa MX Ethernet driver produce?

Overview

The Yokogawa MX Ethernet Driver provides an easy and reliable way to connect Yokogawa MX Ethernet devices to OPC Client applications, including HMI, SCADA, Historian, MES, ERP and countless custom applications. It is intended for use with Yokogawa MX Data Acquisition devices that support Ethernet TCP communications.

Device Setup

Supported Yokogawa MX Series Devices

MX100

Note: Firmware versions preceding 3.01 are supported.

Connection Timeout

This parameter specifies the time that the driver will wait for a connection to be made with a device. Depending on network load, the connect time may vary with each connection attempt. The valid range is 1 to 30 seconds. The default setting is 3 seconds.

Request Timeout

This parameter specifies the amount of time that the driver will wait for a response from the device before giving up and going on to the next request. Long timeouts will only affect performance if a device is not responding. The default setting is 1000 milliseconds. The valid range is 100 to 30000 milliseconds.

Retry Attempts

This parameter specifies the number of times that the driver will retry a message before giving up and going on to the next message. The default setting is 3 retries. The valid range is 1 to 10.

Note: For more information, refer to the OPC server's help file.

Device Configuration Parameters

Device ID

Yokogawa MX series devices are networked using standard IP addressing. In general, the Device ID has the format YYY.YYY.YYY.YYY, where YYY designates the device IP address. Each YYY byte should be in the range of 0 to 255.

Date & Time

This parameter specifies the origin of the data value of the Date and Time data types which represent the date and time of the latest data. Date & Time options are **Device Time** and **System Time**. If Device Time is selected, the Date and Time tags will return the date and time read from the device. This date and time represents the date and time that the latest data was measured or computed based on the internal device clock. If System Time is selected, the Date and Time tags will return the date and time that the requested data was returned from the device based on the internal system clock.

Date Format

This parameter specifies the format of the return string for the Date data type. Date formats can be specified as **MM/DD/YY** (month/day/year), **YY/MM/DD** (year/month/day) or **DD/MM/YY** (day/month/year).

Stop MX on Shutdown

When checked, this option will inform the driver to send a command to the device to stop measuring when the server is exited.

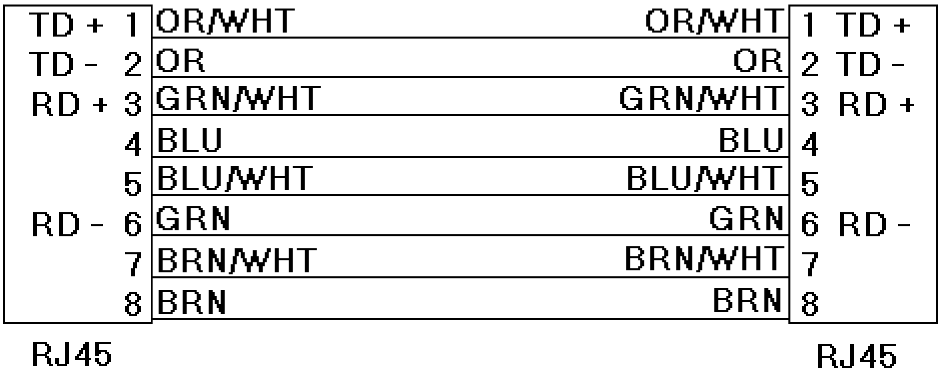
Special Data Handling

This parameter specifies the method of handling data values for channels that return a special condition status, such as data out of range. Special Data Handling options are -INF/+INF or -99999/+99999">. A negative sign indicates an under range condition, and a positive sign is used for all other special conditions including over range.

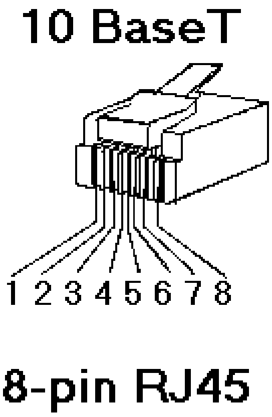
Note: The client software being used may not have the ability to interpret -INF or +INF values. In this case, users will need to select -99999/+99999.

Cable Diagrams

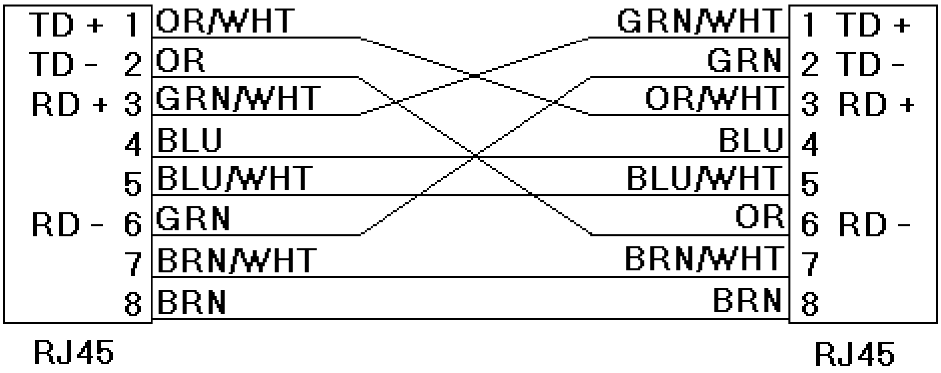
Patch Cable (Straight Through)



TD +	1	OR/WHT	OR/WHT	1	TD +
TD -	2	OR	OR	2	TD -
RD +	3	GRN/WHT	GRN/WHT	3	RD +
	4	BLU	BLU	4	
	5	BLU/WHT	BLU/WHT	5	
RD -	6	GRN	GRN	6	RD -
	7	BRN/WHT	BRN/WHT	7	
	8	BRN	BRN	8	



Crossover Cable

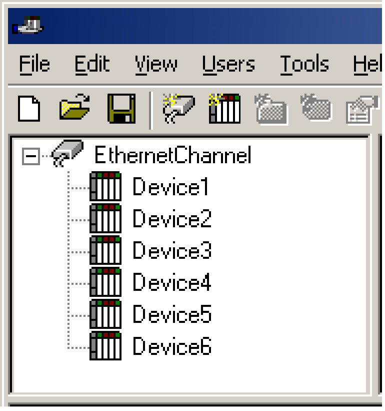


TD +	1	OR/WHT	GRN/WHT	1	TD +
TD -	2	OR	GRN	2	TD -
RD +	3	GRN/WHT	OR/WHT	3	RD +
	4	BLU	BLU	4	
	5	BLU/WHT	BLU/WHT	5	
RD -	6	GRN	OR	6	RD -
	7	BRN/WHT	BRN/WHT	7	
	8	BRN	BRN	8	

Optimizing Your Ethernet Communications

The Yokogawa MX Ethernet driver has been designed to provide the best performance with the least amount of impact on the system's overall performance. While the Yokogawa MX Ethernet driver is fast, there are a couple of guidelines that can be used in order to control and optimize the application and gain maximum performance.

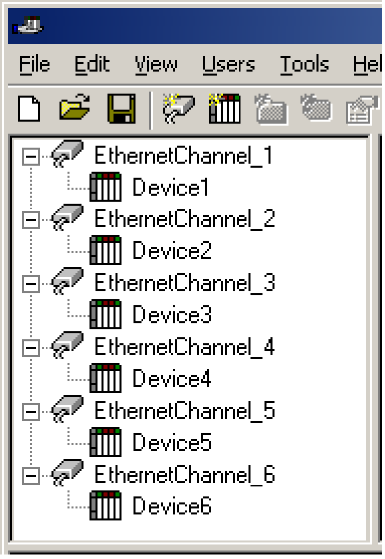
Our server refers to communications protocols like Yokogawa MX Ethernet Device as a channel. Each channel defined in the application represents a separate path of execution in the server. Once a channel has been defined, a series of devices must then be defined under that channel. Each of these devices represents a single Yokogawa MX Ethernet Device from which data will be collected. While this approach to defining the application will provide a high level of performance, it won't take full advantage of the Yokogawa MX Ethernet driver or the network. An example of how the application may appear when configured using a single channel is shown below.



EthernetChannel
Device1
Device2
Device3
Device4
Device5
Device6

Each device appears under a single Yokogawa MX Ethernet Device channel. In this configuration, the driver must move from one device to the next as quickly as possible in order to gather information at an effective rate. As more devices are added or more information is requested from a single device, the overall update rate begins to suffer.

If the Yokogawa MX Ethernet driver could only define one single channel, then the example shown above would be the only option available; however, the Yokogawa MX Ethernet driver can define up to 16 channels. Using multiple channels distributes the data collection workload by simultaneously issuing multiple requests to the network. An example of how the same application may appear when configured using multiple channels to improve performance is shown below.



EthernetChannel_1
Device1
EthernetChannel_2
Device2
EthernetChannel_3
Device3
EthernetChannel_4
Device4
EthernetChannel_5
Device5
EthernetChannel_6
Device6

Each device has now been defined under its own channel. In this new configuration, a single path of execution is dedicated to the task of gathering data from each device. If the application has 16 or fewer devices, it can be optimized exactly how it is shown here.

The performance will improve even if the application has more than 16 devices. While 16 or fewer devices may be ideal, the application will still benefit from additional channels. Although by spreading the device load across all channels will cause the server to move from device to device again, it can now do so with far less devices to process on a single channel.

Data Types Description

Data Type	Description
Boolean	Single bit
Byte	Unsigned 8 bit value bit 0 is the low bit bit 7 is the high bit
Word	Unsigned 16 bit value bit 0 is the low bit bit 15 is the high bit
Short	Signed 16 bit value bit 0 is the low bit bit 14 is the high bit bit 15 is the sign bit
Long	Signed 32 bit value
Float	32 bit floating point value
Double	64 bit floating point value
String	Null terminated ASCII string

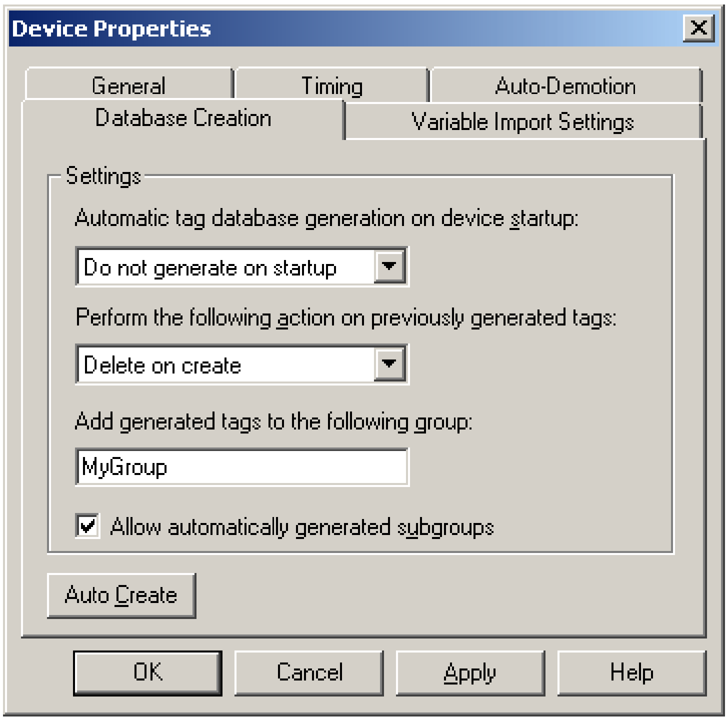
Automatic Tag Database Generation

The automatic OPC tag database generation features of this driver have been designed to make setting up the OPC application a Plug and Play operation. This driver can be configured to automatically build a list of OPC tags within the OPC Server that correspond to device specific data. The automatically generated OPC tags can then be browsed from the OPC client. The OPC tags that are generated depend on the nature of the driver.

If the target device supports its own local tag database, the driver will read the device's tag information and use this data to generate OPC tags within the OPC Server. If the device does not natively support its own named tags, the driver will create a list of tags based on information specific to the driver. An example of these two conditions may be as follows:

1. A data acquisition system that supports its own local tag database. The driver will use the tags names found in the device to build the OPC Server's OPC tags.
2. An Ethernet I/O system that supports the detection of an I/O module type. The driver in this case will automatically generate OPC tags in the OPC Server that are based on the types of I/O modules plugged into the Ethernet I/O rack.

The mode of operation for automatic tag database generation is completely configurable. The following dialog is used to configure how the OPC Server and the associated communications driver will handle automatic OPC tag database generation:



Automatic tag database generation on device startup: Do not generate on startup

Perform the following action on previously generated tags: Delete on create

Add generated tags to the following group: MyGroup

☑ Allow automatically generated subgroups

The **Automatic tag database generation on device startup** selection is used to configure when OPC tags will be automatically generated. There are three possible selections:

- **Do not generate on startup**, the default condition, prevents the driver from adding any OPC tags to tag space of the OPC Server.
- **Always generate on startup** causes the driver to always evaluate the device for tag information and to add OPC tags to the tag space of the server each time the server is launched.
- **Generate on first startup** causes the driver to evaluate the target device for tag information the first time this OPC Server project is run and to add any OPC tags to the server tag space as needed.

When the automatic generation of OPC tags is selected, any tags that are added to the server's tag space must be saved with the project. The OPC Server project can be configured to auto save from the **Tools** | **Options** menu.

When automatic tag generation is enabled, the server needs to know what to do with OPC tags that it may have added from a previous run or with OPC tags that may have been added or modified after the communications driver added them. **Perform the following action** is used to control how the server will handle OPC tags that were automatically generated and currently exist in the OPC Server project. This feature prevents automatically generated tags from piling up in the server. This would occur if, using the Ethernet I/O example above, users continued to change the I/O modules in the rack with the OPC Server configured to always generate new OPC tags on startup. Under this condition every time the communications driver detected a new I/O module, the tags would be added to the server. If the old tags are not removed, a number of unused tags could accumulate in the server's tag space. **Perform the following action** is used to tailor the server's operation to best fit the application's needs. Descriptions of the selections are as follows:

1. **Delete on create**, the default condition, allows the server to remove any tags that had previous been added to the tag space before the communications driver can add any new tags.

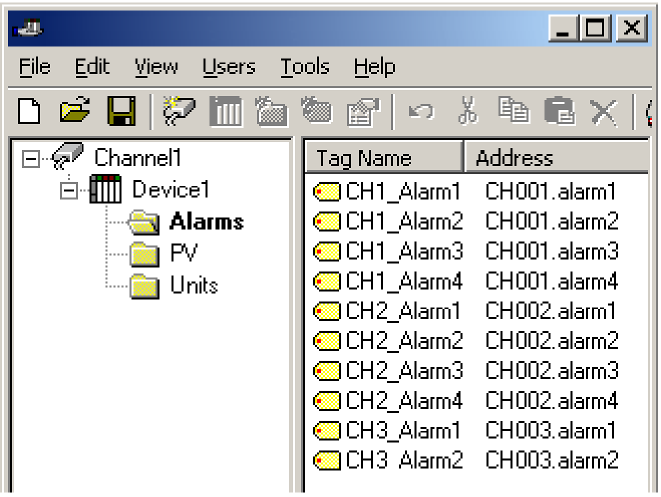
2. **Overwrite as necessary** allows the server to remove only tags the communications driver is replacing with new tags. Any tags that are not being overwritten will remain in the server's tag space.

3. **Do not overwrite** prevents the server from removing any tags that had been previous generated or may have already existed in the server. With this selection, the communications driver can only add tags that are completely new.

4. **Do not overwrite, log error**, has the same effect as the third with an addition: an error message will be posted to the OPC Server's event log when a tag overwrite would have occurred.

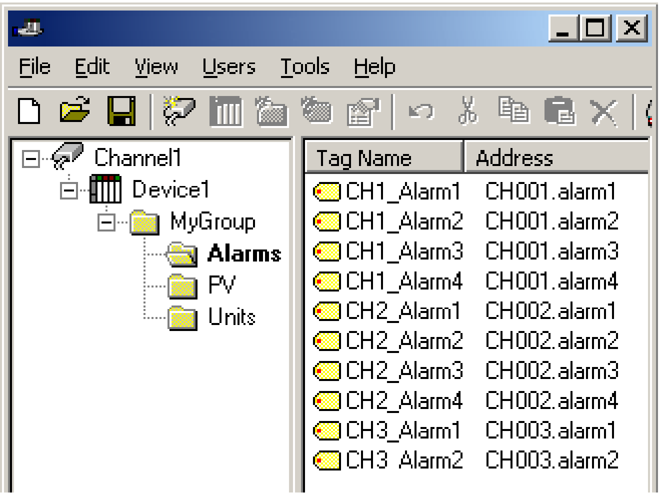
Note: The removal of OPC tags effects tags that have been automatically generated by the communications driver and any tags that have been added using names that match generated tags. It is recommended that users avoid adding own tags to the server using names that match tags that may be automatically generated by the driver.

Add generated tags to the following groupcan be used to keep automatically generated tags from mixing with tags that have been entered manually. This parameter is used to specify a sub group that will be used when adding all automatically generated tags for this device. The name of the sub group can be up to 31 characters in length. The following displays demonstrate how this parameter effects where automatically generated tags are placed in the server's tag space. As shown below, this parameter provides a root branch to which all automatically generated tags will be added.



Tag Name	Address
CH1_Alarm1	CH001.alarm1
CH1_Alarm2	CH001.alarm2
CH1_Alarm3	CH001.alarm3
CH1_Alarm4	CH001.alarm4
CH2_Alarm1	CH002.alarm1
CH2_Alarm2	CH002.alarm2
CH2_Alarm3	CH002.alarm3
CH2_Alarm4	CH002.alarm4
CH3_Alarm1	CH003.alarm1
CH3 Alarm2	CH003.alarm2

No sub group specified.



Tag Name	Address
CH1_Alarm1	CH001.alarm1
CH1_Alarm2	CH001.alarm2
CH1_Alarm3	CH001.alarm3
CH1_Alarm4	CH001.alarm4
CH2_Alarm1	CH002.alarm1
CH2_Alarm2	CH002.alarm2
CH2_Alarm3	CH002.alarm3
CH2_Alarm4	CH002.alarm4
CH3_Alarm1	CH003.alarm1
CH3 Alarm2	CH003.alarm2

Sub group namedMyGroup specified.

Auto Create is used to manually initiate the creation of automatically generated OPC tags. It can be used to make the driver to reevaluate the device for possible tag changes. Auto Create can also be accessed from the System Tags for this device, allowing the OPC client application to initiate tag database creation.

MX100 Addressing

The driver supports the following addresses for this device. The default data type for each address type is shown in **bold**.

Measured Channels

Process Value of Channel	CHxxxxx or CHxxxxx.PV	00001-00060	**Double**, Float	Read Only
Alarm Summary of Channel	CHxxxxx.Alarm	00001-00060	**Short**, Word, Byte	Read Only
Alarm Level1 Status of Channel	CHxxxxx.Alarm1	00001-00060	**Short**, Word, Byte	Read Only
Alarm Level2 Status of Channel	CHxxxxx.Alarm2	00001-00060	**Short**, Word, Byte	Read Only
Alarm Level1 Setpoint*	CHxxxxx.ASP1	00001-00060	**Double**, Float	Read Only
Alarm Level2 Setpoint*	CHxxxxx.ASP2	00001-00060	**Double**, Float	Read Only
Scan interval of Channel*	CHxxxxx.interval	00001-00060	**Short**, Word, Byte	Read Only
Upper Scale Value of Channel*	CHxxxxx.scale_Hi	00001-00060	**Double**, Float	Read Only
Lower Scale Value of Channel*	CHxxxxx.scale_Lo	00001-00060	**Double**, Float	Read Only
Status of Channel	CHxxxxx.status	00001-00060	**Short**, Word, Byte	Read Only
Tag name of Channel*	CHxxxxx.tag	00001-00060	**String**	Read Only
Tag comment of Channel*	CHxxxxx.comment	00001-00060	**String**	Read Only
Unit String of Channel*	CHxxxxx.unit	00001-00060	**String**	Read Only
Precision of Channel*	CHxxxxx.Precision	00001-00060	**Short**, Word, Byte	Read Only
Digital Output on Channel	CHxxxxx.DO	00001-00060	**Boolean**, Short, Word, Byte	Read/Write
Analog Output (Volt) on Channel	CHxxxxx.AOVolt	00001-00060	**Double**, Float	Read/Write
Analog Output (mA) on Channel	CHxxxxx.AOmA	00001-00060	**Double**, Float	Read/Write
Pulse Width Modulation on Channel	CHxxxxx.PWM	00001-00060	**Double**, Float	Read/Write
Set Retransmission Start/Stop for Channel	CHxxxxx.Retransmission	00001-0060	**Boolean**, Short, Word, Byte	Read/Write
Lowest Measuring Channel*	CH.Low		**Short**, Word, Byte	Read Only
Highest Measuring Channel*	CH.High		**Short**, Word, Byte	Read Only

*Initialized Data

Data associated with these addresses are read from the device only at the start of a communications session. Once read, the values will not be refreshed until the server has been restarted or the "Reset" tag has been invoked. To invoke a reset, a non zero value must be written to the Reset tag. Once the Reset tag has been invoked, the driver will reinitialize all startup data from the device.

Note: This will result in a pause of the update of process values and alarms until all initialized data has been read from the device.

Channel Addressing

The Yokogawa MX Ethernet Device is arranged as a 6 slot (numbers 0-5) unit. Each slot may hold a module consisting of up to 10 channels (1-10). The first slot has possible channel numbers of CH00001-00010, the second slot CH000011-00020, the third slot CH00021-00030, the fourth slot CH00031-00040, the fifth slot CH00041-00050, and the last slot has possible channel numbers of 00051-00060. There is potential for gaps in channel numbering due to the fact that not all slots must contain a module, and not all modules are equipped with 10 channels (as in the case of a 4-channel module.)

Tag Names

For channels that have unspecified tag names, the driver will construct an internal tag name based on the channel number. For example, the tag name of address 'CH00001' will be returned as 'CH00001'.

Digital Output

'CHxxxxx.DO' is only valid for channel numbers in slots containing a Digital Output Module.

General Device Data

Address Type	Format	Range	Data Types	Access
Date of Last Data	Date		**String**	Read Only
Time of Last Data	Time		**String**	Read Only
7 segment display pattern	Dispx	1-2	**Long**, Short, Word	Read/Write
Set display blink	Disp.Blink		**Boolean**, Short, Word, Byte	Write Only
Set interval for display blink (mSec)	Disp.Interval		**Integer**	Write Only
Serial Number of Unit	Unit.Serial		**String**	Read Only
Serial Number of Module	Modulex.Serial	0-5	**String**	Read Only
IP Address of Device	IP		**String**	Read Only
Set backup	Backup		**Boolean**, Short, Word, Byte	Read/Write
Reset Alarms	AlarmReset		**Boolean**, Short, Word, Byte	Write Only
Set Retransmission Start/Stop for all channels	RetransAll		**Boolean**, Short, Word, Byte	Write Only
Direct Reloading of Configuration	Reset		**Boolean**, Short, Word, Byte	Write Only
Unit Style	Unit.Style		**Long**	Read Only
Unit firmware version in xx.xx.xx.xx format	Unit.Firmware		**String**	Read Only

7 Segment Display Pattern

'Disp1' refers to the left indicator of the display pattern on the front of the unit. 'Disp2' refers to the right indicator.

Set Display Blink and Set Interval for Display Blink

Setting display blink on (non-zero value) will cause the display on the front of the unit to blink for the set interval.

Write Only Items

Addresses that have Write Only access are assigned a default access of Read/Write. However, data values are unreadable for these addresses and the associated tags are not included in the scan list. The current data value for these tags will always be 0 for numeric data types with the exception of 'Disp.Interval which has a default value of 1000.

Note: The actual number of addresses available for of each type is dependent on the configuration of the Yokogawa device. If the driver finds that an address is not present in the device at Runtime, it will post an error message and remove the tag from its scan list.

Error Descriptions

The following error/warning messages may be generated. Click on the link for a description of the message.

Address Validation

Address '<address>' is out of range for the specified device or register
Data Type '<type>' is not valid for device address '<address>'
Device address '<address>' contains a syntax error
Device address '<address>' is Read Only
Missing address

Device Status Messages

Detected Yokogawa MX 100 Unit Style '<version>' with '<number of modules>' modules and a total of '<number of channels>' channels on device '<device name>'
Device '<device name>' has no FIFO data for channel (Tag '<address>')
Device '<device name>' is not responding
Device '<device name>' responded with error '<error code>' (Tag '<address>', Size '<size>'
Unable to write to '<address>' on device '<device name>

Driver Error Messages

Winsock initialization failed (OS Error = n)
Winsock V1.1 or higher must be installed to use the Yokogawa MX Ethernet device driver

Address Validation

The following error/warning messages may be generated. Click on the link for a description of the message.

Address Validation

Address '<address>' is out of range for the specified device or register
Data Type '<type>' is not valid for device address '<address>'
Device address '<address>' contains a syntax error
Device address '<address>' is Read Only
Missing address

Address '<address>' is out of range for the specified device or register

Error Type:
Warning

Possible Cause:
A tag address that has been specified statically references a location that is beyond the range of supported locations for the device.

Solution:
Verify that the address is correct; if it is not, re-enter it in the client application.

Data Type '<type>' is not valid for device address '<address>'

Error Type:
Warning

Possible Cause:
A tag address that has been specified statically has been assigned an invalid data type.

Solution:
Modify the requested data type in the client application.

Device address '<address>' contains a syntax error

Error Type:
Warning

Possible Cause:
A tag address that has been specified statically contains one or more invalid characters.

Solution:
Re-enter the address in the client application.

Device address '<address>' is Read Only

Error Type:
Warning

Possible Cause:
A tag address that has been specified statically has a requested access mode that is not compatible with what the device supports for that address.

Solution:
Change the access mode in the client application.

Missing address

Error Type:
Warning

Possible Cause:
A tag address that has been specified statically has no length.

Solution:
Re-enter the address in the client application.

Device Status Messages

The following error/warning messages may be generated. Click on the link for a description of the message.

Device Status Messages
Detected Yokogawa MX 100 Unit Style '<version>' with '<number of modules>' modules and a total of '<number of channels>' channels on device '<device name>'
Device '<device name>' has no FIFO data for channel (Tag '<address>')
Device '<device name>' is not responding
Device '<device name>' responded with error '<error code>' (Tag '<address>', Size '<size>'
Unable to write to '<address>' on device '<device name>

Detected Yokogawa MX 100 Unit Style '<version>' with '<number of modules>' modules and a total of '<number of channels>' channels on device '<device name>'

Error Type:
Informational

Possible Cause:
Communications have occurred with the device, and the unit and channel information requests have completed successfully.

Solution:
N/A.

Note:
This message provides a summary of the device configuration.

Device '<device name>' has no FIFO data for channel (Tag '<address>')

Error Type:
Serious

Possible Cause:
Requested FIFO data is not available from the device.

Solution:
N/A

Device '<device name>' is not responding

Error Type:
Serious

Possible Cause:
1. The connection between the device and the Host PC is broken.
2. The IP address assigned to the device is incorrect.
3. The device Ethernet port is already in use.
4. The connection cannot be established in the specified timeout period.
5. The response from the device took longer to receive than the amount of time specified in the "Request Timeout" device setting.

Solution:
1. Verify the cabling between the PC and the PLC device.
2. Verify that the IP address given to the named device matches that configured in the actual device. If applicable, verify the subnet mask and default gateway settings configured in the actual device.
3. Check for another connected application (such as MX Standard software) and disconnect. The MX device's TCP/IP port only supports a single connection.
4. Increase the Connect Timeout value in the Timeout page of Device Properties.
5. Increase the Request Timeout setting so that the entire response can be handled.

Device '<device name>' responded with error '<error code>' (Tag '<address>', Size '<size>')

Error Type:
Serious

Possible Cause:
The Yokogawa MX Ethernet device responded with the indicated error code.

Solution:
The solution will depend upon the device error code.

See Also:
The Yokogawa MX Ethernet documentation of the specific error code.

Unable to write to '<address>' on device '<device name>'

Error Type:
Serious

Possible Cause:
1. The connection between the device and the Host PC is broken.
2. The named device may have been assigned an incorrect IP address.
3. The address specified may be Read Only or may not exist in the current device.

Solution:
1. Verify the cabling between the PC and the PLC device.
2. Verify the IP address given to the named device matches that of the actual device.
3. Check address availability for the device.

Driver Error Messages

The following error/warning messages may be generated. Click on the link for a description of the message.

Driver Error Messages
Winsock initialization failed (OS Error = n)
Winsock V1.1 or higher must be installed to use the Yokogawa MX Ethernet device driver

Winsock initialization failed (OS Error = n)

Error Type:

Fatal

OS Error	Indication	Possible Solution
10091	Indicates that the underlying network subsystem is not ready for network communication.	Wait a few seconds and restart the driver.
10067	Limit on the number of tasks supported by the Windows Sockets implementation has been reached.	Close one or more applications that may be using Winsock and restart the driver.

Winsock V1.1 or higher must be installed to use the Yokogawa MX Ethernet device driver

Error Type:
Fatal

Possible Cause:
The version number of the Winsock DLL found on the system is less than 1.1.

Solution:
Upgrade Winsock to version 1.1 or higher.

Index

A

Address '<address>' is out of range for the specified device or register 12

Address Validation 12

Automatic Tag Database Generation 7

B

Boolean 7

D

Data Type '<type>' is not valid for device address '<address>' 12

Data Types Description 7

Detected Yokogawa MX 100 Unit Style '<version>' with '<number of modules>' modules and a total of '<number of channels>' channels on device '<device name>' 13

Device '<device name>' has no FIFO data for channel (Tag '<address>') 13

Device '<device name>' is not responding 14

Device '<device name>' responded with error '<error code>' (Tag '<address>', Size '<size>') 14

Device address '<address>' contains a syntax error 12

Device address '<address>' is Read Only 13

Device ID 4

Device Setup 4

Device Status Messages 13

Driver Error Messages 14

E

Error Descriptions 12

F

Float 7

L

Long 7

M

Missing address 13
MX100 Addressing 10

N

Network 4

O

Optimizing Your Ethernet Communications 6
Overview 3

S

Short 7

U

Unable to write tag '<address>' on device '<device name>' 14

W

Winsock initialization failed (OS Error = n) 14
Winsock V1.1 or higher must be installed to use the Yokogawa MX Ethernet device driver 15
Word 7