

Yokogawa MW Ethernet Driver Help

© 2012 Kepware Technologies

Table of Contents

Table of Contents	2
Yokogawa MW Ethernet Driver Help	3
Overview	3
Device Setup	4
Communication Parameters	4
Optimizing Your Ethernet Communications	7
Data Types Description	8
Automatic Tag Database Generation	9
MW100 Addressing	12
Error Descriptions	15
Address Validation	15
Address '<address>' is out of range for the specified device or register.....	15
Data Type '<type>' is not valid for device address '<address>'.....	15
Device address '<address>' contains a syntax error.....	15
Device address '<address>' is Read Only.....	16
Missing address.....	16
Device Status Messages	16
Device '<device name>' login failed. Check username and password.....	16
Device '<device>' responded with error '<errnum>' (Tag '<address>').....	16
Device '<device>' responded with error '<error>'.....	17
Device '<device>' returned error '2 <Value exceeds setting range>' (Tag '<address>').....	17
Device '<device name>' is not responding.....	17
Device '<device name>' login failed. No more logins at this user level.....	17
Unable to set clock for device '<device>'.....	17
Unable to start math for device '<device>'.....	18
Unable to start measuring for device '<device>'.....	18
Unable to write to '<address>' on device '<device name>'.....	18
Driver Error Messages	18
Winsock initialization failed (OS Error = n).....	18
Winsock V1.1 or higher must be installed to use the Yokogawa MW Ethernet Device Driver.....	19
Index	20

Yokogawa MW Ethernet Driver Help

Help version 1.021

CONTENTS

[Overview](#)

What is the Yokogawa MW 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 MW Ethernet Driver?

[Data Types Description](#)

What data types does this driver support?

[Address Descriptions](#)

How do I address a data location on a Yokogawa MW Ethernet Device?

[Automatic Tag Database Generation](#)

How can I easily configure tags for the Yokogawa MW Ethernet Driver?

[Error Descriptions](#)

What error messages does the Yokogawa MW Ethernet Driver produce?

Overview

The Yokogawa MW Ethernet Driver provides an easy and reliable way to connect Yokogawa MW Ethernet devices to OPC Client applications, including HMI, SCADA, Historian, MES, ERP and countless custom applications.

Device Setup

Supported Devices

MW100

Device ID

Yokogawa MW Ethernet 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.

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 valid range is 100 to 30000 milliseconds. The default setting is 1000 milliseconds.

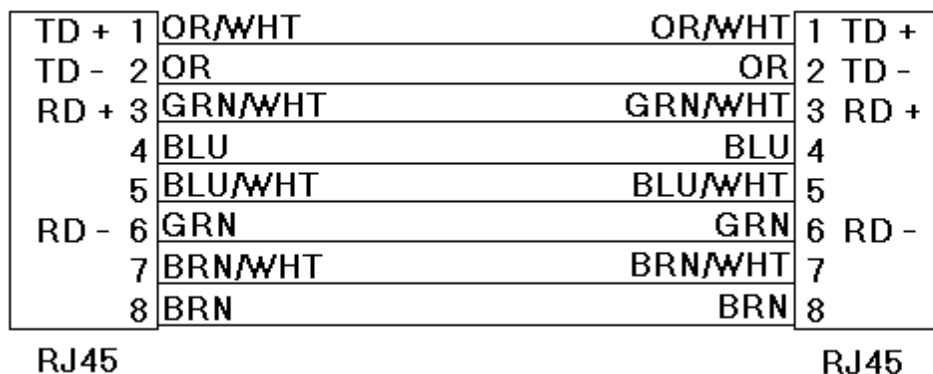
Fail After ... Successive Timeouts

This parameter specifies how many times the driver will retry a communications request before considering the request to have failed and the device to be in error. The valid range is 1 to 10 retries. The default setting is 3.

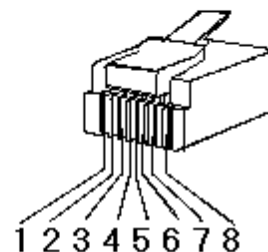
Note: For more information, refer to the OPC server's help documentation.

Cable Diagrams

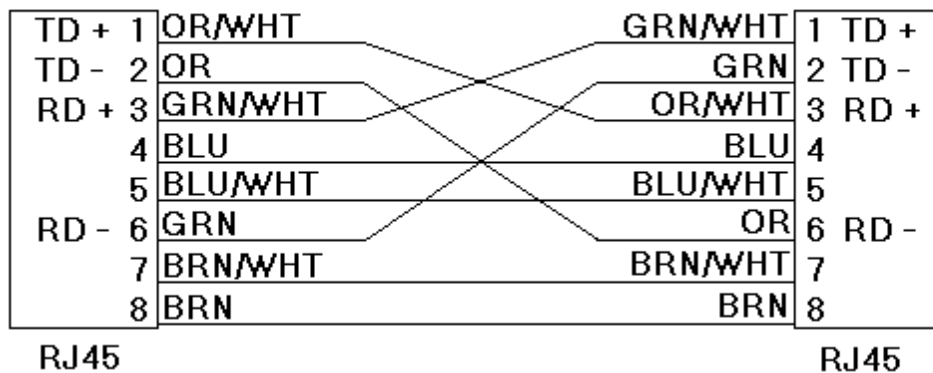
Patch Cable (Straight Through)



10 BaseT



Crossover Cable



8-pin RJ45

Communication Parameters

New Device - Communications Parameters

General
 Port: Ethernet
 Data Handling: None
 Start Math when start
 Start Measuring when start

Time Settings
 Date & Time: Device Time
 Date Format: MM/DD/YY
 Set clock when start

Tagnames
 Generate tag database using: Physical Channel Number

Login
 Username: admin Password: *

< Back Next > Cancel Help

Descriptions of the parameters are as follows:

- **Port:** This parameter specifies the port that is supported by the driver. The default setting is Ethernet.
- **Data Handling:** This parameter allows the driver to be configured to return specific data values for the numerical out of range and error conditions returned from the device. Options include None, +INF, and -INF. The default setting is None. For more information, refer to [Data Handling](#).
- **Start Math when start:** When checked, this option will inform the driver to send a command to the device at communication startup that will start the math computation. The default setting is unchecked.
- **Start Measuring when start:** When checked, this option will inform the driver to send a command to the device at communication startup that will start the measuring option. The default setting is unchecked.
- **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). Options include Device Time and System Time. The default setting is Device Time. Descriptions of the options are as follows:
 - **Device Time:** When selected, the Date and Time tags will return the date and time read from the device. This represents the date and time of the latest data that was measured or computed based on the internal device clock.
 - **System Time:** When selected, the Date and Time tags will return the date and time of the requested data that 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. Options include MM/DD/YY (month/day/year), YY/MM/DD (year/month/day), or DD/MM/YY (day/month/year). The default setting is MM/DD/YY.
- **Set clock when start:** When checked, this option will inform the driver to send a command to the device at communication startup that will set the device clock to the date and time settings of the system clock. The default setting is unchecked.
- **Generate tag database using:** This parameter specifies the origin of the tag name that will be used when a tag database is automatically generated. Options include Physical Channel Number and Device Tagname. The default setting is Physical Channel Number. Descriptions are as follows:
 - **Physical Channel Number:** When selected, the driver will generate tag names based on the channel number of an item. For example, "CH01" or "CH01_alarm1".
 - **Device Tagname:** When selected, the driver will generate tag names using the tag name returned by the device for a channel. Special characters (such as slash and pound sign) are not allowed in the tag name. For example, "Flow" or "Flow_alarm1".

- **Username:** This parameter specifies the registered username, which is required by Yokogawa MW devices when the Login function is enabled. This username may be up to 16 alphanumeric characters, and is case-sensitive.

Note: When the Login function is enabled, users must specify both a registered username and a password. When the device is disabled, users must specify the user level in order to communicate with the Yokogawa MW device. To indicate the user level, enter the username "admin" or "user". This does not require a password.

- **Password:** This parameter specifies the registered password, which may be up to 6 alphanumeric characters. Password entry will not be displayed on the Device Configuration screen.

Data Handling

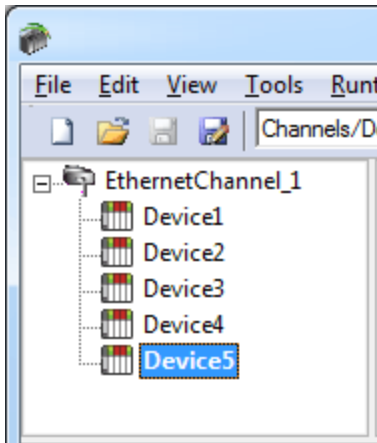
Data Handling specifies how to return specific data values for the numerical out of range and error conditions that are returned from the device. Options include None, +INF, and -INF. Descriptions of the options are as follows:

- **None:** When selected, special data values will be returned with the actual data value received from the device. For example, the data value of a measuring channel Over Range would be returned as 32,767 and the data value of a math channel Over Range would be returned as 2,147,450,879.
- **+INF:** When selected, special data values will be returned as a numerical representation of positive infinity (#INF). The exception is an Under Range condition that is always returned as negative infinity.
- **-INF:** When selected, special data values will be returned as a numerical representation of negative infinity (-#INF). The exception is an Over Range condition that is always returned as positive infinity.

Optimizing Your Ethernet Communications

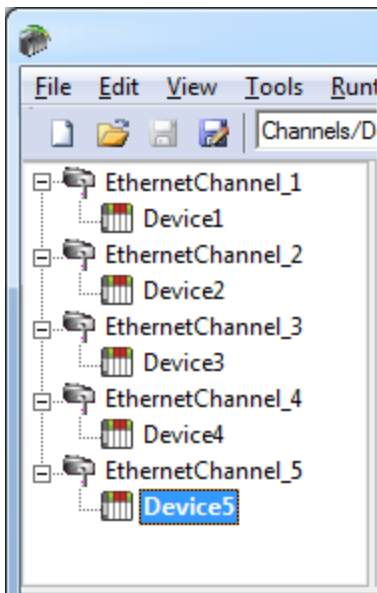
The Yokogawa MW Ethernet Driver has been designed to provide the best performance with the least amount of impact on the system's overall performance. While the Yokogawa MW Ethernet Driver is fast, there are a couple of guidelines that can be implemented to control and optimize the application and gain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

This server refers to communications protocols like Yokogawa MW Ethernet Driver 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 MW 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 MW Ethernet Driver or the network. An example of how the application may appear when configured using a single channel is shown below.



Each device appears under a single Yokogawa MW Ethernet 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 MW Ethernet Driver could only define one single channel, then the example shown above would be the only option available; however, the Yokogawa MW 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.



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 spreading the device load across all channels will still cause the server to move from device to device, 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
Word	Unsigned 16 bit value
Short	Signed 16 bit value
Long	Signed 32 bit value
Float	32 bit floating point value
Double	64 bit floating point value
String	Null terminated ASCII string

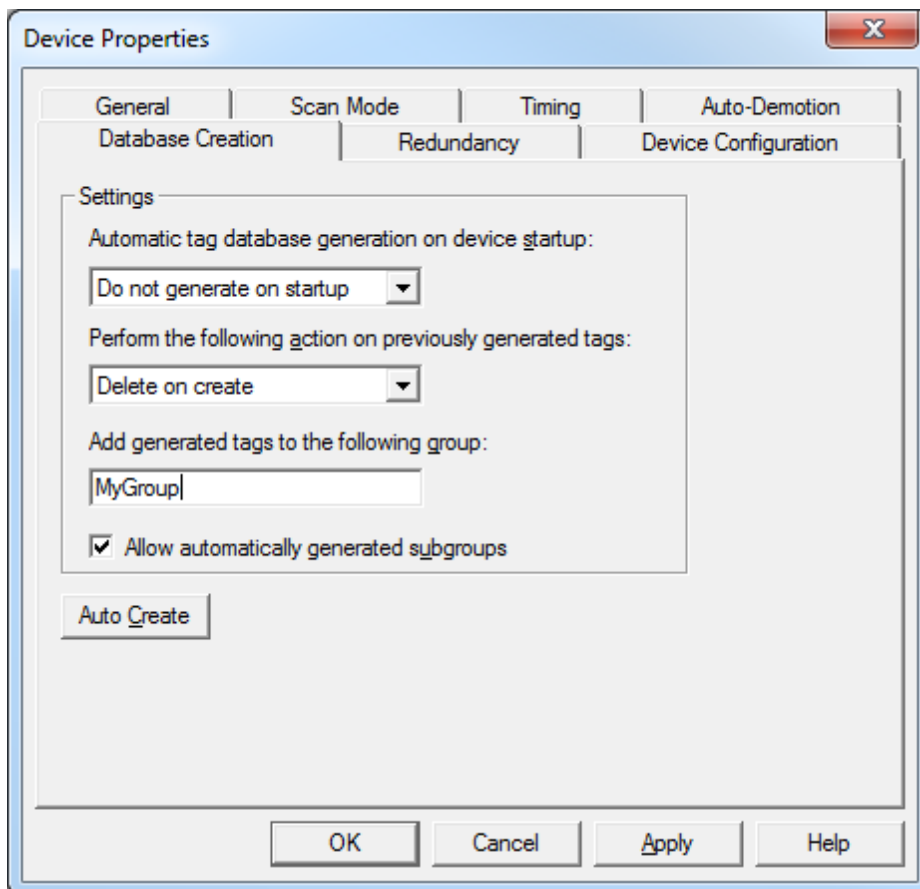
Automatic Tag Database Generation

This driver's Automatic OPC Tag Database Generation features have been designed to make configuring 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 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 device-specific information. An example of these two conditions is as follows:

1. A data acquisition system that supports its own local tag database. The driver will use the tag 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 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.

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 Tag Database Generation:



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.

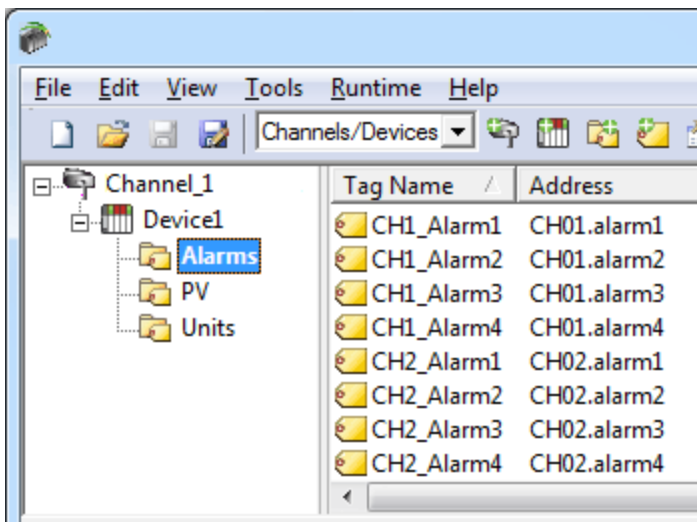
Note: Any tags that are added to the server's tag space must be saved with the project. The OPC Server project can be configured to automatically save from the **Tools | Options** menu.

When automatic tag generation is enabled, the server needs to know what to do with the OPC tags that were added from previous runs (or with the OPC tags that have been added or modified after being added by the communications driver originally). The selection **Perform the following action** controls 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. In the Ethernet I/O example above, this would occur if users continued to change the I/O modules in the rack while the OPC Server was configured to always generate new OPC tags on startup. Under this condition, tags would be added to the server every time the communications driver detected a new I/O module. If the old tags are not removed, they will accumulate in the server's tag space. Descriptions of the selections are as follows:

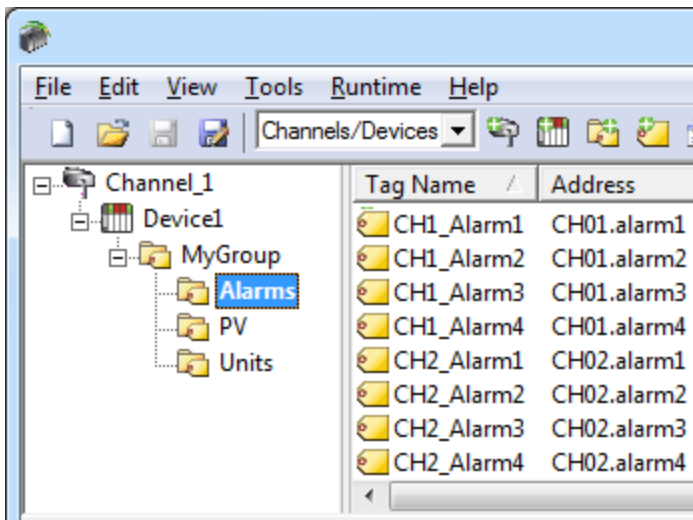
1. **Delete on create:** This option allows the server to remove any tags that have previously been added to the tag space before any new tags can be added by the communications driver. This is the default setting.
2. **Overwrite as necessary:** This option allows the server to only remove the tags that the communications driver will replace with new tags. Any tags that are not being overwritten will remain in the server's tag space.
3. **Do not overwrite:** This option prevents the server from removing any tags that had previously been generated or that already exist in the server. With this selection, the communications driver can only add tags that are completely new.
4. **Do not overwrite, log error:** This option has the same effect as the third option, but also posts an error message to the OPC Server's Event Log when a tag overwrite would have occurred.

Note: The removal of OPC tags not only affects tags that have been automatically generated by the communications driver, but also any tags that have been added using names that match generated tags. It is recommended that users avoid adding tags to the server using names that match tags that may be automatically generated by the driver.

Add generated tags to the following group can be used to keep automatically-generated tags from mixing with tags that have been entered manually. This parameter specifies 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 image demonstrates how this parameter affects where automatically-generated tags are placed in the server's tag space. It provides a root branch to which all automatically-generated tags will be added.



No sub group specified.



Sub group named MyGroup specified.

Auto Create manually initiates the creation of automatically-generated OPC tags, and also forces the driver to reevaluate the device for possible tag changes. It can also be accessed from the System Tags, thus allowing the OPC client application to initiate tag database creation.

MW100 Addressing

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

Measured Channels

Address Type	Format	Range	Data Types	Access
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 Level3 Status of Channel	CHxxxxx.Alarm3	00001-00060	Short , Word, Byte	Read Only
Alarm Level4 Status of Channel	CHxxxxx.Alarm4	00001-00060	Short , Word, Byte	Read Only
Alarm Level1 Setpoint*	CHxxxxx.ASP1	00001-00060	Double , Float	Read/Write
Alarm Level2 Setpoint*	CHxxxxx.ASP2	00001-00060	Double , Float	Read/Write
Alarm Level3 Setpoint*	CHxxxxx.ASP3	00001-00060	Double , Float	Read/Write
Alarm Level4 Setpoint*	CHxxxxx.ASP4	00001-00060	Double , Float	Read/Write
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	String	Read Only
Tag name of Channel*	CHxxxxx.tag	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	Read Only
Precision of Channel*	CHAxxxxx.Precision	001-300	Short , Word	Read Only
Digital Input	CHxxxxx.DI	00001-00060	Boolean	Read Only
Alarm Type 1 for Channel (Num)	CHxxxxx.AlarmType1.Num	00001-00060	Short , Word, Byte	Read Only
Alarm Type 2 for Channel (Num)	CHxxxxx.AlarmType2.Num	00001-00060	Short , Word, Byte	Read Only
Alarm Type 3 for Channel (Num)	CHxxxxx.AlarmType3.Num	00001-00060	Short , Word, Byte	Read Only
Alarm Type 4 for Channel (Num)	CHxxxxx.AlarmType4.Num	00001-00060	Short , Word, Byte	Read Only
Alarm Type 1 for Channel (String)	CHxxxxx.AlarmType1.String	00001-00060	String	Read Only
Alarm Type 2 for Channel (String)	CHxxxxx.AlarmType2.String	00001-00060	String	Read Only
Alarm Type 3 for Channel (String)	CHxxxxx.AlarmType3.String	00001-00060	String	Read Only
Alarm Type 4 for Channel (String)	CHxxxxx.AlarmType4.String	00001-00060	String	Read Only
Process Value of Math	CHAxxxxx	001-300	Double , Float	Read Only
Process Value of Math	CHAxxxxx.PV	001-300	Double , Float	Read Only
Alarm Summary for Channel	CHAxxxxx.alarm	001-300	Short , Word, Byte	Read Only

Alarm 1 for Channel	CHxxxxx.alarm1	001-300	Short , Word, Byte	Read Only
Alarm 2 for Channel	CHxxxxx.alarm2	001-300	Short , Word, Byte	Read Only
Alarm 3 for Channel	CHxxxxx.alarm3	001-300	Short , Word, Byte	Read Only
Alarm 4 for Channel	CHxxxxx.alarm4	001-300	Short , Word, Byte	Read Only
Alarm Setpoint 1 for Channel	CHxxxxx.ASP1	001-300	Double , Float	Read/Write
Alarm Setpoint 2 for Channel	CHxxxxx.ASP2	001-300	Double , Float	Read/Write
Alarm Setpoint 3 for Channel	CHxxxxx.ASP3	001-300	Double , Float	Read/Write
Alarm Setpoint 4 for Channel	CHxxxxx.ASP4	001-300	Double , Float	Read/Write
Upper Scale Value for Channel	CHxxxxx.scale_Hi	001-300	Double , Float	Read Only
Lower Scale Value for Channel	CHxxxxx.scale_Lo	001-300	Double , Float	Read Only
Unit String of Channel	CHxxxxx.unit	001-300	String	Read Only
Tagname of Channel	CHxxxxx.tag	001-300	String	Read Only
Status of Channel	CHxxxxx.status	001-300	String	Read Only
Last Math Channel	CHA.High	001-300	Short , Word	Read Only
First Math Channel	CHA.Low	001-300	Short , Word	Read Only
Alarm Type 1 for Channel (Numeric)	CHxxxxx.AlarmType1.Num	001-300	Short	Read Only
Alarm Type 1 for Channel (String)	CHxxxxx.AlarmType1.String	001-300	String	Read Only
Digital Output on Channel	CHxxxxx.DO	00001-00060	Boolean	Read/Write
Analog Output (Volt) on Channel**	CHxxxxx.AOVolt	00001-00060	Float	Read/Write
Analog Output (mA) on Channel	CHxxxxx.AOMa	00001-00060	Float	Read/Write
Pulse Width Modulation on Channel	CHxxxxx.PWM	00001-00060	Float	Read/Write
Lowest Measuring Channel*	CH.Low		Short , Word	Read Only
Highest Measuring Channel*	CH.High		Short , Word	Read Only

*The data associated with these addresses will only be read from the device 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. This will result in a pause of the update of process values and alarms until all initialized data has been read from the device.

*The Yokogawa MW 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 CH00011-00020, 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.

Analog Output (Volt)

Depending upon how fast multiple writes are written to this tag, the response time during multiple and frequent writes might be slow.

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
SetTime	SetTime		Boolean	Write Only
Model String	Model		String	Read Only
Math Communication Data	CDxxx	001-300	Float	Write Only

MathControl	Math Control		Short	Write Only
OperationCtrl	Operation Control		Short	Write Only
Serial Number of Unit	SerialNumber		String	Read Only
IP Address of Device	IP		String	Read Only
Reset Alarms	AlarmReset		Boolean, Short, Word, Byte	Write Only
Direct Reloading of Configuration	Reset		Boolean, Short, Word, Byte	Write Only
ClearError	Clear Error		Boolean	Write Only
Control Command and Response*	Command		String	Read/Write

*The Control Command and Response address type allows users to send a string command and receive a string response to and from the device. As such, users can send commands to the device that are not directly supported by the driver. This tag is only available to users logged in at the Administrator level; otherwise, write operations will return an error. Binary data is not supported.

Caution: It is recommended that users be careful when performing write operations using the Command address.

Note 1: 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 2: The actual number of addresses available for of each type depends 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

[Device '<device>' login failed. Check username and password](#)

[Device '<device>' responded with error '<errnum>' \(Tag '<address>'\)](#)

[Device '<device>' responded with error '<error>'](#)

[Device '<device>' returned error '2 <Value exceeds setting range>'. \(Tag '<address>'\)](#)

[Device '<device name>' is not responding](#)

[Device <device name> login failed. No more logins at this user level](#)

[Unable to set clock for device '<device>'](#)

[Unable to start math for device '<device>'](#)

[Unable to start measuring for device '<device>'](#)

[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 MW 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 dynamically 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 dynamically 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 dynamically 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 dynamically 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 dynamically 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

[Device '<device>' login failed. Check username and password](#)

[Device '<device>' responded with error '<errnum>' \(Tag '<address>'\)](#)

[Device '<device>' responded with error '<error>'](#)

[Device '<device>' returned error '2 <Value exceeds setting range>'. \(Tag '<address>'\)](#)

[Device '<device name>' is not responding](#)

[Device <device name> login failed. No more logins at this user level](#)

[Unable to set clock for device '<device>'](#)

[Unable to start math for device '<device>'](#)

[Unable to start measuring for device '<device>'](#)

[Unable to write to '<address>' on device '<device name>'](#)

Device '<device name>' login failed. Check username and password

Error Type:

Warning

Possible Cause:

An invalid username has been specified.

Solution:

Retry with a valid username and password.

Device '<device>' responded with error '<errnum>' (Tag '<address>')

Error Type:

Warning

Possible Cause:

An error has been returned by the device that is not specified in the driver.

Solution:

Look up the error number in the device manual.

Device '<device>' responded with error '<error>'

Error Type:

Warning

Possible Cause:

The device has failed to automatically generate a tag.

Solution:

Look up the error number in the device manual.

Device '<device>' returned error '2 <Value exceeds setting range>'. (Tag '<address>')

Error Type:

Warning

Possible Cause:

The driver is attempting to write a value to a tag that exceeds its limit.

Solution:

Retry writes that are within the accepted range.

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 MW Standard software) and disconnect.
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.

Note:

The MW device's TCP/IP port supports a single connection.

Device '<device name>' login failed. No more logins at this user level

Error Type:

Warning

Possible Cause:

The device is already connected to the maximum number of clients that is allowed at a time.

Solution:

Wait until a connection becomes available for that device.

Unable to set clock for device '<device>'

Error Type:

Warning

Possible Cause:

The driver has failed to set the clock in the device.

Solution:

1. Check connections to the device.
2. The device may have been busy performing other functions. Retry.

Unable to start math for device '<device>'**Error Type:**

Warning

Possible Cause:

1. The driver has failed to start math on the device.
2. The math option is missing from the device.

Solution:

1. Check connections to the device.
2. Math cannot be started when certain other features are already on. Check the device manual for details.
3. Use a device with the math option.

Unable to start measuring for device '<device>'**Error Type:**

Warning

Possible Cause:

The driver has failed to start measuring on the device.

Solution:

1. Check connections to the device.
2. Measuring cannot be started when certain other features are already on. Check the device manual for details.

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 that 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 MW 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 com-	Wait a few seconds and restart the driver.

	munication.	
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 MW 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.....	15
Address Validation.....	15
Automatic Tag Database Generation.....	9

B

Boolean.....	8
--------------	---

C

Communication Parameters.....	4
-------------------------------	---

D

Data Type '<type>' is not valid for device address '<address>'.....	15
Data Types Description.....	8
Device '<device name>' is not responding.....	17
Device '<device>' login failed. Check username and password.....	16
Device '<device>' responded with error '<errnum>' (Tag '<address>').....	16
Device '<device>' responded with error '<error>'.....	17
Device '<device>' returned error '2 <Value exceeds setting range>' (Tag '<address>').....	17
Device <device name> login failed. No more logins at this user level.....	17
Device address '<address>' contains a syntax error.....	15
Device address '<address>' is Read Only.....	16
Device ID.....	4
Device Setup.....	4
Device Status Messages.....	16
Driver Error Messages.....	18

E

Error Descriptions..... 15

F

Float..... 8

L

Long..... 8

M

Missing address..... 16

MW100 Addressing..... 12

N

Network..... 4

O

Optimizing Your Ethernet Communications..... 7

Overview..... 3

S

Short..... 8

U

Unable to set clock for device '<device>'.....	17
Unable to start math for device '<device>'.....	18
Unable to start measuring for device '<device>'.....	18
Unable to write tag '<address>' on device '<device name>'.....	18

W

Winsock initialization failed (OS Error = n).....	18
Winsock V1.1 or higher must be installed to use the Yokogawa MX Ethernet Device Driver..	19
Word.....	8