

Mitsubishi FX Net Driver

©2015 Kepware, Inc.

Table of Contents

Table of Contents	2
Mitsubishi FX Net Driver	3
Overview	3
Device Setup	4
Modem Setup	5
FX3U PLC Parameter Settings	6
Data Types Description	7
Address Descriptions	8
FX Addressing	8
FX2C Addressing	8
FX0N Addressing	9
FX2N Addressing	9
FX3U Addressing	9
FXOpen Addressing	10
Error Descriptions	11
Missing address.	11
Device address <address> contains a syntax error.	11
Address <address> is out of range for the specified device or register.	11
Device address <address> is not supported by model <model name>	11
Data Type <type> is not valid for device address <address>.	12
Device address <address> is read only.	12
COMn does not exist.	12
Error opening COMn.	12
COMn is in use by another application.	12
Unable to set comm parameters on COMn.	13
Communications error on <channel name> [<error mask>].	13
Error Mask Definitions	13
Device <device name> is not responding.	13
Unable to write to <address> on device <device name>.	14
Device <device name> reported an invalid address in the range <start address> to <end address>.	14
Index	15

Mitsubishi FX Net Driver

Help version 1.031

CONTENTS

[Overview](#)

What is the Mitsubishi FX Net Driver?

[Device Setup](#)

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

[FX3U PLC Parameters](#)

What parameter settings are necessary for an FX3U PLC to communicate with this driver?

[Data Types Descriptions](#)

What data types does this driver support?

[Address Descriptions](#)

How do I address a data location on a Mitsubishi FX series device?

[Error Descriptions](#)

What messages does the Mitsubishi FX Net Driver produce?

Overview

The Mitsubishi FX Net Driver provides a reliable way to connect Mitsubishi FX Net devices to OPC client applications; including HMI, SCADA, Historian, MES, ERP, and countless custom applications. It is intended for use with Mitsubishi FX series devices.

Device Setup

Supported Devices

FX
FX2C
FX0N
FFX2N
FFX3U

Notes:

1. FX3U is not supported in Windows CE.
2. FXOpen is a general model that supports the driver's maximum address range. It may be selected for models other than FX, FX2C, FX0N, FX2N, and FX3U. It should not be used if the device is one of the models specifically supported by this driver (such as FX, FX2C, FX0N, FX3U, and FX2N). For example, if the device is FX0N, choose the FX0N model. Selecting FXOpen as the model when the device is FX, FX2C, FX0N, FX2N, or FX3U may result in bad tag reads and incorrect values.

See Also: [FX3U PLC Parameter Settings](#)

Communication Protocol

Format 1, Checksum

Supported Communication Parameters

Programmable

Ethernet Encapsulation

This driver supports Ethernet Encapsulation, which allows the driver to communicate with serial devices attached to an Ethernet network using a Serial-to-Ethernet server. It may be invoked through the Communications tab in Channel Properties. For more information, refer to the OPC server's help documentation.

Note: Ethernet Encapsulation is not supported for the FX3U model.

Maximum Number of Channels and Devices

The maximum number of channels supported by this driver is 100. The maximum number of devices supported is 16.

Device IDs

The valid range for Device IDs is 0 to 15.

Flow Control

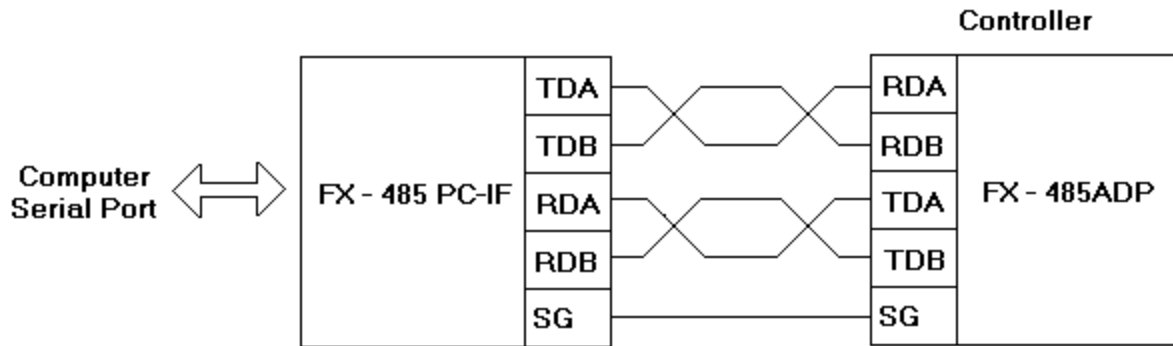
When using an RS232/RS485 converter, the type of flow control that is required depends on the needs of the converter. Some converters do not require any flow control whereas others require RTS flow. Consult the converter's documentation to determine its flow requirements. An RS485 converter that provides automatic flow control is recommended.

Notes:

1. When using the manufacturer's supplied communications cable, it is sometimes necessary to choose a flow control setting of **RTS** or **RTS Always** under the Channel Properties.
2. The FX-485PC-IF RS485 Interface Unit is configured via special memory locations within the FX PLC. Users should consult the FX-485PC-IF manual for memory locations and settings to properly configure the unit (and select matching settings for the Mitsubishi FX Net Driver).

Cable Connections

A null modem cable is required when connecting the RS-232 port of the computer to the FX-485PC-IF.



Modem Setup

This driver supports modem functionality. For more information, please refer to the topic "Modem Support" in the OPC server help documentation.

FX3U PLC Parameter Settings

The FX3U PLC needs the following parameter settings to communicate with the Mitsubishi FX Net Driver.

1. To start, open the GX Developer-FX software application. Then, click **Parameter | PLC Parameter**.
2. Next, select the **PLC system(2)** tab. Change the settings as needed in order to match those shown in the image below.

The screenshot shows the 'FX parameter' dialog box with the following settings:

- CH1 (dropdown)
- Operate communication setting (checked)
- Protocol: Dedicated protocol (dropdown)
- Data length: 7bit (dropdown)
- Parity: Odd (dropdown)
- Stop bit: 1bit (dropdown)
- Transmission speed: 9600 (bps) (dropdown)
- Header: (unchecked)
- Terminator: (unchecked)
- Control line: (unchecked)
- H/W type: Regular/RS-232C (dropdown)
- Control mode: Invalid (dropdown)
- Sum check: (checked)
- Transmission control procedure: Form1(without CR,LF) (dropdown)
- Station number setting: 00 H (00H-0FH) (text input)
- Time out judge time: 1 X10ms (1-255) (text input)

Buttons at the bottom: Default, Check, End, Cancel.

3. Next, click **End | Online | Write to PLC**.
4. Click the **Clear PLC** memory button.

Note: It is important to clear the PLC memory before writing the new settings to the PLC. If the PLC memory isn't cleared, the new settings will not take effect.

5. To write the settings to the PLC, click **Execute**.
6. Once finished, cycle the power on the PLC so that it will be ready to communicate with the Mitsubishi FX Net Driver.

Data Types Description

The Mitsubishi FX Net Driver supports the following data types.

Data Type	Description
Boolean	Single 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
DWord	Unsigned 32-bit value bit 0 is the low bit bit 31 is the high bit
Long	Signed 32-bit value bit 0 is the low bit bit 30 is the high bit bit 31 is the sign bit
Float	32-bit floating point value. The driver interprets two consecutive registers as a floating point value by making the second register the high word and the first register the low word.

Address Descriptions

Address specifications vary depending on the model in use. Select a link from the following list to obtain specific address information for the model of interest.

[FX Addressing](#)

[FX2C Addressing](#)

[FX0N Addressing](#)

[FX2N Addressing](#)

[FX3U Addressing](#)

[FXOpen Addressing](#)

FX Addressing

The default data types for dynamically defined tags are shown in **bold**.

Device Type	Range	Data Type	Access
Inputs	X000-X377*	Boolean	Read Only
Outputs	Y000-Y377*	Boolean	Read/Write
Auxiliary Relays	M0000-M1535	Boolean	Read/Write
Special Aux. Relays	M8000-M8255	Boolean	Read/Write
States	S000-S999	Boolean	Read/Write
Timer Contacts	TS000-TS255	Boolean	Read Only
Counter Contacts	CS000-CS255	Boolean	Read Only
Timer Value	T000-T255	Short, Word	Read/Write
Counter Value	C000-C199	Short, Word	Read/Write
32 Bit Counter Value**	C200-C255	Long, DWord	Read/Write
Data Registers**	D000-D999 D000-D998	Short , Word, Long, DWord, Float	Read/Write
Special Data Registers**	D8000-D8255 D8000-D8254	Short , Word, Long, DWord, Float	Read/Write

*Octal.

**Users can specify a Long data type by appending a space and an "L" to the address. For example, "D000" would be entered as "D000 L". This does not apply to arrays or bit accessed registers.

FX2C Addressing

The default data types for dynamically defined tags are shown in **bold**.

Device Type	Range	Data Type	Access
Inputs	X000-X377*	Boolean	Read Only
Outputs	Y000-Y377*	Boolean	Read/Write
Auxiliary Relays	M0000-M1535	Boolean	Read/Write
Special Aux. Relays	M8000-M8255	Boolean	Read/Write
States	S000-S999	Boolean	Read/Write
Timer Contacts	TS000-TS255	Boolean	Read Only
Counter Contacts	CS000-CS255	Boolean	Read Only
Timer Value	T000-T255	Short, Word	Read/Write
Counter Value	C000-C199	Short, Word	Read/Write
32 Bit Counter Value**	C200-C255	Long, DWord	Read/Write
Data Registers**	D000-D999 D000-D998	Short , Word Long, DWord, Float	Read/Write
Special Data Registers**	D8000-D8255 D8000-D8254	Short , Word Long, DWord, Float	Read/Write

*Octal.

**Users can specify a Long data type by appending a space and an "L" to the address. For example, "D000" would be entered as "D000 L". This does not apply to arrays or bit accessed registers.

FX0N Addressing

The default data types for dynamically defined tags are shown in **bold**.

Device Type	Range	Data Type	Access
Inputs	X000-X177*	Boolean	Read Only
Outputs	Y000-Y177*	Boolean	Read/Write
Auxiliary Relays	M0000-M0511	Boolean	Read/Write
Special Aux. Relays	M8000-M8255	Boolean	Read/Write
States	S000-S127	Boolean	Read/Write
Timer Contacts	TS00-TS63	Boolean	Read Only
Counter Contacts	CS00-CS31 CS235-CS254	Boolean	Read Only
Timer Value	T00-T63	Short, Word	Read/Write
Counter Value	C00-C31	Short, Word	Read/Write
32 Bit Counter Value**	C235-C254	Long, DWord	Read/Write
Data Registers**	D000-D255 D000-D254	Short , Word Long, DWord, Float	Read/Write
Special Data Registers**	D8000-D8255 D8000-D8254	Short , Word Long, DWord, Float	Read/Write

*Octal.

**Users can specify a Long data type by appending a space and an "L" to the address. For example, "D000" would be entered as "D000 L". This does not apply to arrays or bit accessed registers.

FX2N Addressing

The default data types for dynamically defined tags are shown in **bold**.

Device Type	Range	Data Type	Access
Inputs	X000-X377*	Boolean	Read Only
Outputs	Y000-Y377*	Boolean	Read/Write
Auxiliary Relays	M0000-M3071	Boolean	Read/Write
Special Aux. Relays	M8000-M8255	Boolean	Read/Write
States	S000-S999	Boolean	Read/Write
Timer Contacts	TS000-TS255	Boolean	Read Only
Counter Contacts	CS000-CS255	Boolean	Read Only
Timer Value	T000-T255	Short, Word	Read/Write
Counter Value	C000-C199	Short, Word	Read/Write
32 Bit Counter Value**	C200-C255	Long, DWord	Read/Write
Data Registers**	D000-D7999 D000-D7998	Short , Word Long, DWord, Float	Read/Write
Special Data Registers**	D8000-D8255 D8000-D8254	Short , Word Long, DWord, Float	Read/Write

*Octal.

**Users can specify a Long data type by appending a space and an "L" to the address. For example, "D000" would be entered as "D000 L". This does not apply to arrays or bit accessed registers.

FX3U Addressing

The default data types for dynamically defined tags are shown in **bold**.

Note: The FX3U model is not supported in Windows CE. Also, note that the FX3U model does not support Ethernet Encapsulation.

Device Type	Range	Data Type	Access
Inputs	X000-X377*	Boolean	Read Only
Outputs	Y000-Y377*	Boolean	Read/Write
Auxiliary Relays	M0000-M7679	Boolean	Read/Write

Device Type	Range	Data Type	Access
Special Aux. Relays	M8000-M8511	Boolean	Read/Write
States	S0000-S4095	Boolean	Read/Write
Timer Contacts	TS000-TS511	Boolean	Read Only
Counter Contacts	CS000-CS255	Boolean	Read Only
Timer Value	T000-T511	Short, Word	Read/Write
Counter Value	C000-C199	Short, Word	Read/Write
32 Bit Counter Value**	C200-C255	Long, DWord	Read/Write
Data Registers**	D000-D7999 D000-D7998	Short , Word Long, DWord , Float	Read/Write
Special Data Registers**	D8000-D8511 D8000-D8510	Short , Word Long, DWord , Float	Read/Write

*Octal.

**Users can specify a Long data type by appending a space and an "L" to the address. For example, "D000" would be entered as "D000 L". This does not apply to arrays or bit accessed registers.

FXOpen Addressing

The default data types for dynamically defined tags are shown in **bold**.

Note: When adding a device to the OPC server project, do not select FXOpen if the device is one of the models that is specifically supported by this driver (such as FX, FX2C, FX0N, and FX2N). For example, if the device is FX0N, select the FX0N model. Selecting FXOpen as the model when the device is FX, FX2C, FX0N, or FX2N may result in bad tag reads and incorrect values.

Device Type	Range	Data Type	Access
Inputs	X000-X777*	Boolean	Read Only
Outputs	Y000-Y777*	Boolean	Read/Write
Auxiliary Relays	M0000-M9999***	Boolean	Read/Write
Special Aux. Relays	M0000-M9999***	Boolean	Read/Write
States	S000-S999	Boolean	Read/Write
Timer Contacts	TS000-TS999	Boolean	Read Only
Counter Contacts	CS000-CS999	Boolean	Read Only
Timer Value	T000-T999	Short, Word	Read/Write
Counter Value	C000-C999****	Short, Word	Read/Write
32 Bit Counter Value**	C000-C998****	Long, DWord	Read/Write
Data Registers**	D000-D9999 D000-D9998	Short , Word Long, DWord , Float	Read/Write
Special Data Registers**	Please refer to the device manual.	Short , Word Long, DWord , Float	Read/Write

*Octal.

**Users can specify a Long data type by appending a space and an "L" to the address. For example, "D000" would be entered as "D000 L". This does not apply to arrays or bit accessed registers.

***For Auxiliary Relays and Special Aux. Relays, the driver can accommodate the fullest range. For information on the model to determine the separation between Auxiliary Relays and Special Aux. Relays, refer to the manual.

****For Counter Value and 32 Bit Counter Value, the driver can accommodate the fullest range. For information on the model to determine the separation between Counter Value and 32 Bit Counter Value, refer to the manual.

Error Descriptions

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

Address Validation

[Missing address.](#)

[Device address <address> contains a syntax error.](#)

[Address <address> is out of range for the specified device or register.](#)

[Device address <address> is not supported by model <model name>.](#)

[Data type <type> is not valid for device address <address>.](#)

[Device address <address> is read only.](#)

Serial Communications

[COMn does not exist.](#)

[Error opening COMn.](#)

[COMn is in use by another application.](#)

[Unable to set comm parameters on COMn.](#)

[Communications error on <channel name> \[<error mask>\].](#)

Device-Status Messages

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

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

Device-Specific Messages

[Device <device name> reported an invalid address in the range <start address> to <end address>.](#)

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 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.

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.

Device address <address> is not supported by model <model name>

Error Type:

Warning

Possible Cause:

A tag address that has been specified dynamically references a location that is valid for the communications protocol but not supported by the target device.

Solution:

Verify that the address is correct; if it is not, re-enter it in the client application. Also verify that the selected model name for the device is correct.

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> 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.

COMn does not exist.

Error Type:

Fatal

Possible Cause:

The specified COM port is not present on the target computer.

Solution:

Verify that the proper COM port has been selected.

Error opening COMn.

Error Type:

Fatal

Possible Cause:

The specified COM port could not be opened due an internal hardware or software problem on the target computer.

Solution:

Verify that the COM port is functional and may be accessed by other Windows applications.

COMn is in use by another application.

Error Type:

Fatal

Possible Cause:

The serial port assigned to a device is being used by another application.

Solution:

Verify that the correct port has been assigned to the channel.

Unable to set comm parameters on COMn.

Error Type:

Fatal

Possible Cause:

The serial parameters for the specified COM port are not valid.

Solution:

Verify the serial parameters and make any necessary changes.

Communications error on <channel name> [<error mask>].

Error Type:

Serious

Possible Cause:

1. The serial connection between the device and the host PC is bad.
2. The communications parameters for the serial connection are incorrect.

Solution:

1. Verify the cabling between the PC and the PLC device.
2. Verify that the specified communications parameters match those of the device.

See Also:[Error Mask Definitions](#)

Error Mask Definitions

B = Hardware break detected**F** = Framing error**E** = I/O error**O** = Character buffer overrun**R** = RX buffer overrun**P** = Received byte parity error**T** = TX buffer full

Device <device name> is not responding.

Error Type:

Serious

Possible Cause:

1. The serial connection between the device and the Host PC is broken.
2. The communications parameters for the serial connection are incorrect.
3. The named device may have been assigned an incorrect Network ID.
4. 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 specified communications parameters match those of the device.

3. Verify that the Network ID given to the named device matches that of the actual device.
4. Increase the Request Timeout setting so that the entire response can be handled.

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

Error Type:

Serious

Possible Cause:

1. The serial connection between the device and the host PC is broken.
2. The communications parameters for the serial connection are incorrect.
3. The named device may have been assigned an incorrect network ID.

Solution:

1. Verify the cabling between the PC and the PLC device.
2. Verify that the specified communications parameters match those of the device.
3. Verify that the network ID given to the named device matches that of the actual device.

Device <device name> reported an invalid address in the range <start address> to <end address>.

Error Type:

Serious

Possible Cause:

An attempt has been made to reference a nonexistent location in the specified device.

Solution:

Verify that the tags assigned to addresses in the specified range on the device and eliminate those that reference invalid locations.

Index

A

Address <address> is out of range for the specified device or register. 11
Address Descriptions 8
Address Validation 11

B

Boolean 7

C

Cable Connections 4
Character buffer overrun 13
Communication Parameters 4
Communications error on <channel name> [<error mask>] 13
COMn does not exist. 12
COMn is in use by another application. 12

D

Data Type <type> is not valid for device address <address>. 12
Data Types Description 7
Device-Specific Messages 11
Device-Status Messages 11
Device <device name> is not responding. 13
Device <device name> reported an invalid address in the range <start address> to <end address>. 14
Device address <address> contains a syntax error. 11
Device address <address> is not supported by model <model name>. 11
Device address <address> is read only. 12
Device ID 4
Device Setup 4
DWord 7

E

Error Descriptions 11
Error Mask Definitions 13
Error opening COMn. 12
Ethernet Encapsulation 4

F

Float 7
Flow Control 4
Framing 13
FX Addressing 8
FX0N Addressing 9
FX2C Addressing 8
FX2N Addressing 9
FX3U Addressing 9
FX3U Parameter Settings 6
FXOpen Addressing 10

G

GX Developer-FX 6

H

Hardware break 13

L

Long 7

M

Mask. 13
Missing address. 11
Mitsubishi FX
 series devices 3
Modem Setup 5
Modem Support 5

O

Overview 3

P

Parity 13

Protocol 4

R

RTS 4

RX buffer overrun 13

S

Serial Communications 11

Short 7

Signed 7

Supported 4

T

TX buffer 13

U

Unable to set comm parameters on COMn. 13

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

Unsigned 7

W

Word 7

