

# **Yaskawa MP Series Serial Driver Help**

© 2012 Kepware Technologies

# Table of Contents

Table of Contents .....	2
Yaskawa MP Series Serial Driver Help .....	3
Overview .....	3
Device Setup .....	4
Block Sizes .....	4
Cable Diagrams .....	4
Modem Setup .....	5
Data Types Description .....	6
Address Descriptions .....	7
MP Series Address Descriptions .....	7
GL Series Address Descriptions .....	8
Error Descriptions .....	9
Address Validation .....	9
Missing address .....	9
Device address '<address>' contains a syntax error .....	9
Address '<address>' is out of range for the specified device or register .....	9
Device address '<address>' is not supported by model '<model name>' .....	10
Data Type '<type>' is not valid for device address '<address>' .....	10
Device address '<address>' is Read Only .....	10
Array size is out of range for address '<address>' .....	10
Array support is not available for the specified address: '<address>' .....	10
Device Status Messages .....	11
Device '<device name>' is not responding .....	11
Unable to write to '<address>' on device '<device name>' .....	11
Device Specific Messages .....	11
Device '<device>' responded with error '<Memobus error code>' (Tag '<tag>', Size '<bytes>') .....	11
Bad received length [<start address> to <end address>] on device '<device>' .....	12
Bad address in block [<start address> to <end address>] on device '<device>' .....	12
Device '<device>' block request [<start address> to <end address>] responded with exception <Mem-... obus error code> .....	12
Index .....	13

## Yaskawa MP Series Serial Driver Help

---

Help version 1.018

### CONTENTS

#### [Overview](#)

What is the Yaskawa MP Series Serial Driver?

#### [Device Setup](#)

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

#### [Data Types Description](#)

What data types does the Yaskawa MP Series Serial Driver support?

#### [Address Descriptions](#)

How do I reference a data location in a Yaskawa MP Series Serial Device?

#### [Error Descriptions](#)

What error messages does the Yaskawa MP Series Serial Driver produce?

### Overview

---

The Yaskawa MP Series Serial Driver provides an easy and reliable way to connect Yaskawa MP Series Serial devices to OPC Client applications, including HMI, SCADA, Historian, MES, ERP and countless custom applications. It supports the Yaskawa MP 900 series CPUs.

## Device Setup

---

### Supported Devices

Yaskawa MP Series  
Yaskawa GL Series

### Communication Protocol

Memobus RTU Protocol.

### Supported Communication Parameters

Baud Rate: 9600, 14400, 19200  
Parity: Even, Odd, None  
Data Bits: 7, 8  
Stop Bits: 1, 2

**Note:** Settings should be chosen to match the hardware's configuration.

### Ethernet Encapsulation

This driver supports Ethernet Encapsulation, which allows the driver to communicate with serial devices attached to an Ethernet network using a terminal server (such as Digi One IA). It may be invoked through the COM ID dialog in Channel Properties. For more information, refer to the OPC server's help documentation.

### 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 will require RTS flow. To determine the converter's flow requirements, refer to its help documentation. An RS485 converter that provides automatic flow control is recommended.

### Device ID

Every device on the network must have a unique network address. Set the Device ID driver setting to match the target device. The Device ID may range from 0 to 63.

### Request Timeout

This parameter specifies the time that the driver will wait on a response from the device before giving up and going on to the next request. Longer timeouts only affect performance if a device is not responding. The default setting is 1000 milliseconds. The valid range is 100 to 9999 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.

## Block Sizes

---

### Block Sizes for Bit addresses

Input bits (IB) and output bits (MB) can be read from 8 to 800 points (bits) at a time.

### Block Sizes for Register addresses

Input registers (IW, IL, IF) and output registers (MW, ML, MF) can be read from 1 to 120 locations (words) at a time.

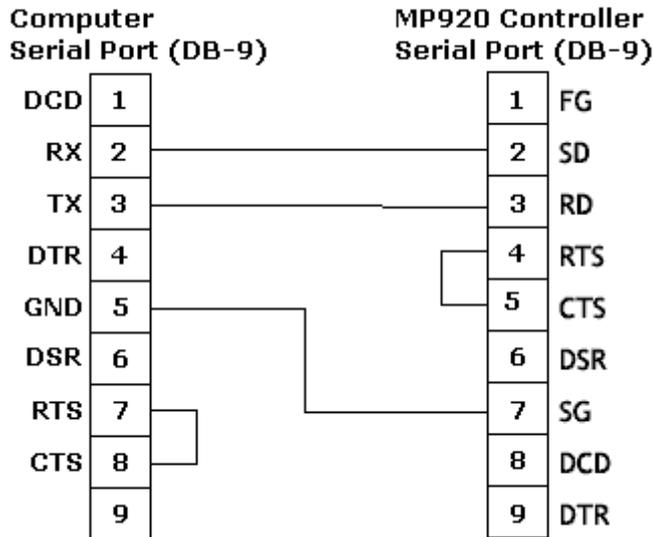
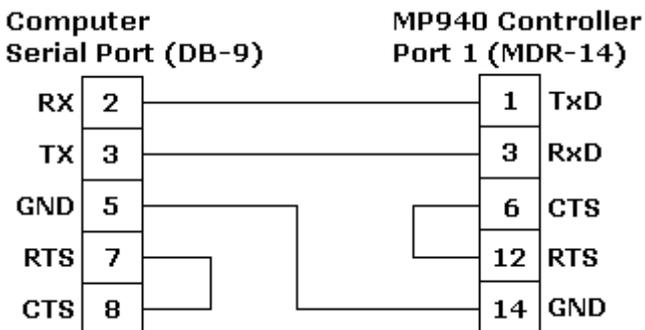
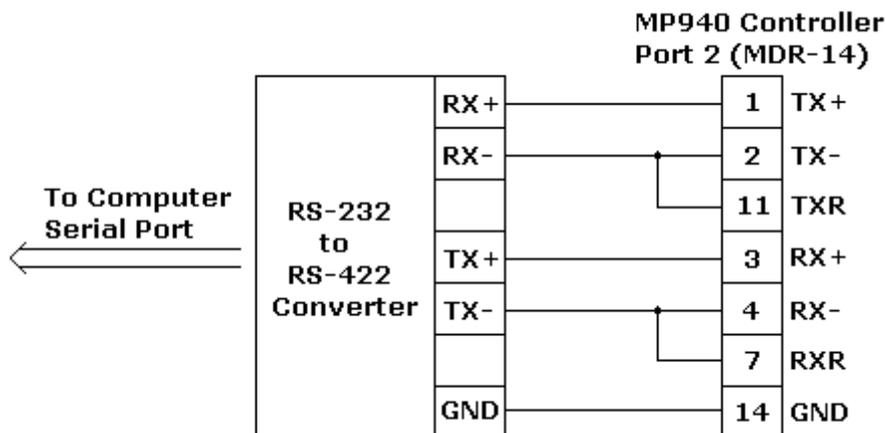
**Note:** The application may benefit by changing the block size. First, that future versions of the device may not support block Read/Write operations of the default size. Second, that the device may contain non-contiguous addresses, such as when a binary space module is used. If this is the case and the driver attempts to read a block of data that encompasses undefined memory, the device will probably reject the request.

## Cable Diagrams

---

The following images are sample cable diagrams for the Yaskawa MP920 and MP940 controllers.

### MP920 (RS-232)

**MP940 (RS-232)****MP940 (RS-422/485)****Modem Setup**

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

## Data Types Description

---

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
BCD	Two byte packed BCD Value range is 0-9999. Behavior is undefined for values beyond this range.
LBCD	Four byte packed BCD Value range is 0-99999999. Behavior is undefined for values beyond this range.
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.  When accessing data from a thermocouple or resistance sensor input module, the driver will use a single 16-bit register as a floating point value.
Float Example	If register 40001 is specified as a float, bit 0 of register 40001 would be bit 0 of the 32 bit word, and bit 15 of register 40002 would be bit 31 of the 32 bit 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.

[MP Series](#)

[GL Series](#)

## MP Series Address Descriptions

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

Memory Type	Range	Data Type	Access
Input Bits	IB0000.b-IBFFFE.b (b is bit number 0x0-0xF)	<b>Boolean</b>	Read Only
Output Bits	MB00000.b-MB65534.b (b is bit number 0x0-0xF)	<b>Boolean</b>	Read/Write
Input Registers	IW0000-IWFFFE	<b>Short</b> , Word, BCD	Read Only
	IW0000-IWFFFD	<b>Long</b> , DWord, LBCD, Float	
	IW0000.b-IWFFFE.b (b is bit number 0x0-0xF)	<b>Boolean</b>	
	IL0000-ILFFFD	<b>Long</b> , DWord, LBCD	
	IF0000-IFFFFD	<b>Float</b>	
Output Registers	MW00000-MW65534	<b>Short</b> , Word, BCD	Read/Write
	MW00000-MW65533	<b>Long</b> , DWord, LBCD, Float	
	MW00000.b-MW65534.b (b is bit number 0x0-0xF)	<b>Boolean</b>	
	ML00000-ML65533	<b>Long</b> , DWord, LBCD	
	MF00000-MF65533	<b>Float</b>	

## Arrays

Arrays are also supported for register addresses (IW, IL, IF, MW, ML, MF). The syntax for declaring an array is MMxxxx[cols] with assumed row count of 1 and MMxxxx[rows][cols], where "MM" is the memory type mnemonic and "xxxx" is the base address of the array data.

- For Word, Short, and BCD arrays, the base address + (rows \* cols) - 1 cannot exceed 65534.
- For Float, DWord, Long, and Long BCD arrays, the base address + (rows \* cols \* 2) - 1 cannot exceed 65533.
- For all arrays, the total number of registers being requested cannot exceed the register block size that was specified for this device.

**Note:** Arrays are not supported for Boolean types (Ibxxxx.b, MBxxxx.b, IWxxxx.b, MWxxxx.b).

## Input Addresses

Input addresses (IB, IW, IL, IF) are in hex. Bit numbers (b) are always in hex. Array "rows" and "cols" are always in decimal.

## Output Addresses

Output addresses (MB, MW, ML, MF) are in decimal and map to the same memory area. For example, MB00001.F is the same as MW00001.F. ML00001 and MF00001 both map to the same memory as MW00001 and MW00002. The same is true for input addresses.

**Note:** Writes to MB00000-MB04095 are faster than writes to MB04096-MB65534 because they can take advantage of direct bit access Memobus commands. Writes to the higher bits require the driver to perform a Read/Modify/Write operation, taking approximately twice as long.

**Important:** The actual range of valid addresses is hardware specific and may be smaller than the range allowed by this driver.

## GL Series Address Descriptions

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

Memory Type	Range	Data Type	Access
Input Bits	IB0000.b-IBFFFE.b (b is bit number 0x0-0xF)	<b>Boolean</b>	Read Only
Output Bits	MB00000.b-MB65534.b (b is bit number 0x0-0xF)	<b>Boolean</b>	Read/Write
Input Registers	IW0000-IWFFFE IW0000-IWFFFD IW0000.b-IWFFFE.b (b is bit number 0x0-0xF) IL0000-ILFFFD IF0000-IFFFFD	<b>Short</b> , Word, BCD <b>Long</b> , DWord, LBCD, Float <b>Boolean</b> <b>Long</b> , DWord, LBCD <b>Float</b>	Read Only
Output Registers	MW00000-MW65534 MW00000-MW65533 MW00000.b-MW65534.b (b is bit number 0x0-0xF) ML00000-ML65533 MF00000-MF65533	<b>Short</b> , Word, BCD <b>Long</b> , DWord, LBCD, Float <b>Boolean</b> <b>Long</b> , DWord, LBCD <b>Float</b>	Read/Write
Constant Registers	CW00000-CW65534 CW00000-CW65533	Short, <b>Word</b> , BCD <b>Long</b> , DWord, LBCD, Float	Read/Write

## Arrays

Arrays are also supported for register addresses (IW, IL, IF, MW, ML, MF, CW). The syntax for declaring an array is MMxxxx[cols] with assumed row count of 1 and MMxxxx[rows][cols], where "MM" is the memory type mnemonic and "xxxx" is the base address of the array data.

- For Word, Short, and BCD arrays, the base address + (rows \* cols)-1 cannot exceed 65534.
- For Float, DWord, Long, and Long BCD arrays, the base address + (rows \* cols \* 2)-1 cannot exceed 65533.
- For all arrays, the total number of registers being requested cannot exceed the register block size that was specified for this device.

**Note:** Arrays are not supported for Boolean types (Ibxxxx.b, MBxxxx.b, IWxxxx.b, MWxxxx.b).

## Input Addresses

Input addresses (IB, IW, IL, IF) are in hex. Bit numbers (b) are always in hex. Array "rows" and "cols" are always in decimal.

## Output Addresses

Output addresses (MB, MW, ML, MF) are in decimal and map to the same memory area. For example, MB00001.F is the same as MW00001.F. ML00001 and MF00001 both map to the same memory as MW00001 and MW00002. The same is true for input addresses.

**Note:** Writes to MB00000-MB04095 are faster than writes to MB04096-MB65534 because they can take advantage of direct bit access Memobus commands. Writes to the higher bits require the driver to perform a Read/Modify/Write operation, taking approximately twice as long.

**Important:** The actual range of valid addresses is hardware specific and may be smaller than the range allowed by this driver.

## Error Descriptions

---

The following error/warning 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](#)

[Array size is out of range for address '<address>'](#)

[Array support is not available for the specified address: '<address>'](#)

### Device Status Messages

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

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

### Device Specific Messages

[Device '<device>' responded with error '<Memobus error code>' \(Tag '<tag>', Size '<bytes>'\)](#)

[Bad received length \[<start address> to <end address>\] on device '<device>'](#)

[Bad address in block \[<start address> to <end address>\] on device '<device>'](#)

[Device '<device>' block request \[<start address> to <end address>\] responded with exception '<Memobus error code>'](#)

## Address Validation

---

The following error/warning 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](#)

[Array size is out of range for address '<address>'](#)

[Array support is not available for the specified address: '<address>'](#)

## 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 address '<address>' contains a syntax error

---

### Error Type:

Warning

### Possible Cause:

An invalid tag address has been specified in a dynamic request.

### 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 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.

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

---

**Error Type:**

Warning

**Possible Cause:**

A tag address that has been specified statically 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 statically 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 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.

**Array size is out of range for address '<address>'**

---

**Error Type:**

Warning

**Possible Cause:**

A tag address that has been specified statically is requesting an array size that is too large for the address type or block size of the driver.

**Solution:**

Re-enter the address in the client application to specify a smaller value for the array or a different starting point.

**Array support is not available for the specified address: '<address>'**

---

**Error Type:**

Warning

**Possible Cause:**

A tag address that has been specified statically contains an array reference for an address type that doesn't support arrays.

**Solution:**

Re-enter the address in the client application to remove the array reference or correct the address type.

**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 name>' is not responding](#)

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

**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 communication parameters for the 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 device.
2. Verify that the specified communication 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 named device may not be connected to the network.
2. The named device may have been assigned an incorrect Network ID.
3. The named device is not responding to write requests.
4. The address does not exist in the PLC.

**Solution:**

1. Check the PLC network connections.
2. Verify that the Network ID given to the named device matches that of the actual device.

**Device Specific Messages**

---

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

**Device Specific Messages**

[Device '<device>' responded with error '<Memobus error code>' \(Tag '<tag>', Size '<bytes>'\)](#)

[Bad received length \[<start address> to <end address>\] on device '<device>'](#)

[Bad address in block \[<start address> to <end address>\] on device '<device>'](#)

[Device '<device>' block request \[<start address> to <end address>\] responded with exception <Memobus error code>](#)

**Device '<device>' responded with error '<Memobus error code>' (Tag '<tag>', Size '<bytes>')**

---

**Error Type:**

Fatal

**Possible Cause:**

The driver attempted to read a block of memory in the PLC. The PLC responded with the specified Memobus error code.

**Solution:**

Ensure that the range of memory exists for the PLC.

**Bad received length [<start address> to <end address>] on device '<device>'****Error Type:**

Fatal

**Possible Cause:**

The driver attempted to read a block of memory in the PLC. The PLC responded with no error, but did not provide the driver with the requested block size of data.

**Solution:**

Ensure that the range of memory exists for the PLC.

**Bad address in block [<start address> to <end address>] on device '<device>'****Error Type:**

Fatal

**Possible Cause:**

The driver attempted to read a location that does not exist in a PLC. For example, this error would be generated in a PLC that only has input registers IW00000 to IW10000 but requests address IW10001. Once this error has been generated, the driver will not request the specified block of data from the PLC again. Any other addresses being requested that are in the same block will also be invalid.

**Solution:**

Ensure that the range of memory exists for the PLC.

**Device '<device>' block request [<start address> to <end address>] responded with exception <Memobus error code>****Error Type:**

Fatal

**Possible Cause:**

The driver attempted to read a location that does not exist in a PLC. For example, this error would be generated in a PLC that only has input registers IW00000 to IW10000 but requests address IW10001. Once this error has been generated, the driver will not request the specified block of data from the PLC again. Any other addresses being requested that are in the same block will also be invalid.

**Solution:**

Ensure that the range of memory exists for the PLC.

# Index

## A

Address '<address>' is out of range for the specified device or register.....	9
Address Descriptions.....	7
Address Validation.....	9
Array size is out of range for address '<address>'.....	10
Array support is not available for the specified address: '<address>'.....	10

## B

Bad address in block [<start address> to <end address>] on device <device>.....	12
Bad received length [<start address> to <end address>] on device <device>.....	12
BCD.....	6
Block Sizes.....	4
Boolean.....	6

## C

Cable Diagrams.....	4
Constant Registers.....	8

## D

Data Type '<type>' is not valid for device address '<address>'.....	10
Data Types Description.....	6
Device '<device name>' is not responding.....	11
Device <device> block request [<start address> to <end address>] responded with excep- tion <Memobus error code>.....	12
Device <device> responded with error <Memobus error code> Tag <tag> Size <bytes>...	11
Device address '<address>' contains a syntax error.....	9
Device address '<address>' is not supported by model '<model name>'.....	10
Device address '<address>' is Read Only.....	10
Device Setup.....	4
Device Specific Messages.....	11

<b>Device Status Messages</b> .....	<b>11</b>
<b>DWord</b> .....	<b>6</b>
 <b>E</b>	
<b>Error Descriptions</b> .....	<b>9</b>
 <b>F</b>	
<b>Float</b> .....	<b>6</b>
 <b>G</b>	
<b>GL Series Address Descriptions</b> .....	<b>8</b>
 <b>H</b>	
<b>Holding Register</b> .....	<b>7</b>
 <b>L</b>	
<b>LBCD</b> .....	<b>6</b>
<b>Long</b> .....	<b>6</b>
 <b>M</b>	
<b>Missing address</b> .....	<b>9</b>
<b>Modem Setup</b> .....	<b>5</b>
<b>MP Series Address Descriptions</b> .....	<b>7</b>
 <b>O</b>	
<b>Overview</b> .....	<b>3</b>

**S**

Short ..... 6

**U**

Unable to write to '<address>' on device '<device name>' ..... 11

**W**

Word ..... 6