

Micro DCI Driver Help

© 2012 Kepware Technologies

Table of Contents

Table of Contents	2
Micro DCI Driver Help	3
Overview	3
Driver Setup	4
Device Setup	5
Modem Setup	6
Data Types Description	7
Address Descriptions	8
53MC1000 Addressing	8
53MC2000 Addressing	8
53MC2002 Addressing	8
53MC4000 Addressing	8
53MC5000 Addressing	9
Error Descriptions	10
Address Validation	10
Missing address	10
Device address '<address>' contains a syntax error	10
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>'	11
Device address '<address>' is Read Only	11
Received unexpected response for tag '<address>' on device '<device name>'	11
Serial Communications	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	12
Communications error on '<channel name>' [<error mask>]	12
Device Status Messages	13
Device '<device name>' is not responding	13
Unable to write to '<address>' on device '<device name>'	13
Device Specific Messages	13
Bad address in block [<start address> to <end address>] on device '<device name>'	13
Index	15

Micro DCI Driver Help

Help version 1.014

CONTENTS

[Overview](#)

What is the Micro DCI Driver?

[Driver Setup](#)

How do I configure which Micro DCI port is being used?

[Device Setup](#)

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

[Data Types Description](#)

What data types does this driver support?

[Address Descriptions](#)

How do I address a data location on a Micro DCI device?

[Error Descriptions](#)

What error messages does the Micro DCI driver produce?

Overview

The Micro DCI Driver provides an easy and reliable way to connect Micro DCI devices to OPC Client applications, including HMI, SCADA, Historian, MES, ERP and countless custom applications. It is intended for use with Bailey Fischer & Porter Micro DCI devices.

Driver Setup

Controller Port

This parameter is used to specify the port to which the Micro DCI unit will be connected. The **Configuration Port** should be selected when connecting to the front DIMM port on the device. The **Data Link Port** should be selected when using an RS485 connection through the RS485 interface on the back of the device.

Note: When using the back data link port (RS 485), users may need to set the converter to "Echo On".

Device Setup

Supported Devices

53MC1000
53MC2000
53MC2002
53MC4000
53MC5000

Communication Protocol

Micro DCI DataLink Protocol. Binary mode switching through the configuration port is supported.

Supported Communication Parameters

Baud Rate: 300, 600, 1200, 2400, 9600, 19200, or 38400
Parity: None or Even
Data Bits: 8
Stop Bits: 1

Note: Not all devices support the listed configurations.

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. It may be invoked through the COM ID dialog in Channel Properties. For more information, refer to the OPC server's help documentation.

Device IDs

The Micro DCI Driver's supported Device ID range is 0-31 (decimal).

Byte Stuffing

This setting should be enabled if the device is configured to use byte stuffing. For more information, refer to F&P documentation on parameter L258.

Flow Control

When using an RS232/RS485 converter, the type of flow control that is required will depend 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.

Data Link Port (RS 485)

A high quality manufactured converter is recommended. Users may have to set the converter to "Echo On" when using the back data link port (RS 485).

Note: 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.

Cable Connections

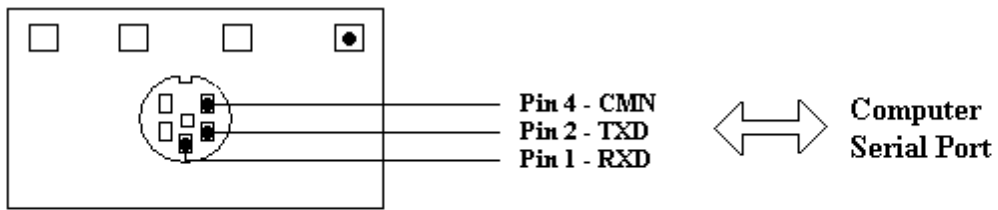


Figure 1. Configuration Port Cable Connection

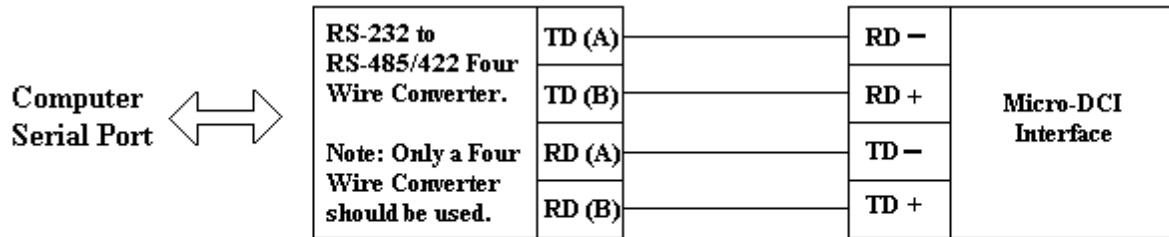


Figure 2. Data Link Port Cable Connection

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
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
Float	32 bit floating point value.
Double	64 bit double precision value
String	Null terminated character array

Address Descriptions

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

[53MC1000](#)

[53MC2000](#)

[53MC2002](#)

[53MC4000](#)

[53MC5000](#)

Note: The actual number of addresses of each type depends on the Micro DCI device in use. Refer to the device documentation for address ranges.

53MC1000 Addressing

The following addresses are supported for this model. The default data types are shown in **bold**.

Description	Syntax	Access	Data Type
Logical Data Points	L0-L63	Read/Write	Bool , Byte or Word*
Byte Data	B0-B11	Read/Write	Byte
Real Data (3-byte Float)	C0-C46	Read/Write	Float
Real Data (5-byte Float)	H0-H10	Read/Write	Float, Double

*Addressing L type data for Bytes and Words are aligned on even Byte/Word boundaries respectively. This means that addressing Byte data is acceptable at addresses L0, L8, L16, L24 (8 bits per byte); Word data at addresses L0, L16, L32 (16 bits per word).

53MC2000 Addressing

The following addresses are supported for this model. The default data types are shown in **bold**.

Description	Syntax	Access	Data Type
Logical Data Points	L0-L255	Read/Write	Bool , Byte or Word*
Byte Data	B0-B127	Read/Write	Byte
Real Data (3-byte Float)	C0-C255	Read/Write	Float
Real Data (5-byte Float)	H0-H36	Read/Write	Float, Double

*Addressing L type data for Bytes and Words are aligned on even Byte/Word boundaries respectively. This means that addressing Byte data is acceptable at addresses L0, L8, L16, L24 (8 bits per byte); Word data at addresses L0, L16, L32 (16 bits per word).

53MC2002 Addressing

The following addresses are supported for this model. The default data types are shown in **bold**.

Description	Syntax	Access	Data Type
Logical Data Points	L0-L255	Read/Write	Bool , Byte or Word*
Byte Data	B0-B255	Read/Write	Byte
Real Data (3-byte Float)	C0-C255	Read/Write	Float
Real Data (5-byte Float)	H0-H127	Read/Write	Float, Double

*Addressing L type data for Bytes and Words are aligned on even Byte/Word boundaries respectively. This means that addressing Byte data is acceptable at addresses L0, L8, L16, L24 (8 bits per byte); Word data at addresses L0, L16, L32 (16 bits per word).

53MC4000 Addressing

The following addresses are supported for this model. The default data types are shown in **bold**.

Description	Syntax	Access	Data Type
Logical Data Points	L0-L119	Read/Write	Bool , Byte or Word*

Byte Data	B0-B49	Read/Write	Byte
Real Data (3-byte Float)	C0-C199	Read/Write	Float
Real Data (5-byte Float)	H0-H49	Read/Write	Float, Double
Text String (10 character)	A0-A9	Read/Write	String

*Addressing L type data for Bytes and Words are aligned on even Byte/Word boundaries respectively. This means that addressing Byte data is acceptable at addresses L0, L8, L16, L24 (8 bits per byte); Word data at addresses L0, L16, L32 (16 bits per word).

53MC5000 Addressing

The following addresses are supported for this model. The default data types are shown in **bold**.

Description	Syntax	Access	Data Type
Logical Data Points	L0-L999	Read/Write*	Bool , Byte or Word**
Byte Data	B0-B639	Read/Write**	Byte
Real Data (3-byte Float)	C0-C639	Read/Write	Float
Real Data (5-byte Float)	H0-H127	Read/Write	Float, Double
Text String (10 character)	A0-A317	Read/Write	String
Text String (5 character)	F0-F447	Read/Write	String

*Addresses L256, L257, L258, B1 and B2 are Read Only addresses. These addresses affect communication parameters in the device.

**Addressing L type data for Bytes and Words are aligned on even Byte/Word boundaries respectively. This means that addressing Byte data is acceptable at addresses L0, L8, L16, L24 (8 bits per byte); Word data at addresses L0, L16, L32 (16 bits per word).

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](#)

[Received unexpected response for tag '<address>' on device '<device name>'](#)

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

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

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](#)

[Received unexpected response for tag '<address>' on device '<device name>'](#)

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.

Received unexpected response for tag '<address>' on device '<device name>'

Error Type:

Warning

The Error Could Be:

1. Unexpected characters in response.
2. Invalid checksum.
3. Invalid response command returned from device.

Possible Cause:

1. Misalignment of packets due to connection/disconnection between the PC and device.
2. Bad cabling connecting the devices may be causing noise.

Solution:

The driver will recover from this error without intervention. If this error occurs frequently, there may be an issue with the cabling or with the device itself.

Serial Communications

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

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>' \[<error mask>\]](#)

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 in the Channel Properties.

Error opening COMn

Error Type:

Fatal

Possible Cause:

The specified COM port could not be opened due to 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

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.

Possible Cause:

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

Solution:

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

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 serial connection between the device and the Host PC is broken.
2. The communication 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 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 serial connection between the device and the Host PC is broken.
2. The communication 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 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.

Device Specific Messages

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

Device Specific Messages

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

Bad address in block [<start address> to <end address>] on device '<device name>'

Error Type:

Serious

Possible Cause:

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

Solution:

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

Index

5

53MC1000 Addressing.....	8
53MC2000 Addressing.....	8
53MC2002 Addressing.....	8
53MC4000 Addressing.....	8
53MC5000 Addressing.....	9

A

Address '<address>' is out of range for the specified device or register.....	11
Address Descriptions.....	8
Address Validation.....	10

B

Bad address in block [<start address> to <end address>] on device '<device name>'.....	13
Boolean.....	7

C

Communications error on '<channel name>' [<error mask>].....	12
COMn does not exist.....	12
COMn is in use by another application.....	12

D

Data Type '<type>' is not valid for device address '<address>'.....	11
Data Types Description.....	7
Device '<device name>' is not responding.....	13
Device address '<address>' contains a syntax error.....	10
Device address '<address>' is not supported by model '<model name>'.....	11
Device address '<address>' is Read Only.....	11

Device ID.....	5
Device Setup.....	5
Device Specific Messages.....	13
Device Status Messages.....	13
Driver Setup.....	4

E

Error Descriptions.....	10
Error opening COMn.....	12

F

Float.....	7
Framing.....	12

M

Mask.....	12
Missing address.....	10
Modem Setup.....	6

N

Network.....	5
--------------	---

O

Overrun.....	13
Overview.....	3

P

Parity.....	13
-------------	----

R

Received unexpected response for tag '<address>' on device '<device name>'..... 11

S

Serial Communications..... 12

String..... 7

U

Unable to set comm parameters on COMn..... 12

Unable to write tag '<address>' on device '<device name>'..... 13

W

Word..... 7