Opto 22 Ethernet Driver Help

Table of Contents

Opto 22 Ethernet Driver Help

Help version 1.018

CONTENTS

Overview

The Opto 22 Ethernet Driver provides an easy and reliable way to connect Opto 22 Ethernet devices to OPC Client applications, including HMI, SCADA, Historian, MES, ERP and countless custom applications.

Device Setup

Supported Devices

Supported devices include the following:

SNAP Industrial Controllers
SNAP PAC S-Series
SNAP PAC R-Series
SNAP-LCE

SNAP Brains
SNAP PAC EB-Series
SNAP Ultimate I/O
SNAP Ethernet I/O
SNAP Simple I/O

E1 and E2 Brain Boards

Communications Protocols

MMIO over Ethernet TCP/IP or UDP
CONT over Ethernet TCP/IP

Note 1: This driver requires Winsock V1.1 or higher.

Note 2: Firmware version 8.0 or higher is required for some features. For more information, refer to **Address Descriptions**.

Maximum Number of Channels and Devices

The maximum number of channels is 256. The maximum number of devices is 65535 per channel.

Device ID

The Device ID is used to specify the device's IP address.

Connection Timeout

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

Request Timeout

This parameter specifies the amount of time that the driver will wait on a response from the device before retrying or giving up and moving 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 30000 milliseconds.

Retry Attempts

This parameter specifies the number of times the driver will retry a request before moving on to the next request. The default setting is 3 retries. The valid range is 1 to 10.

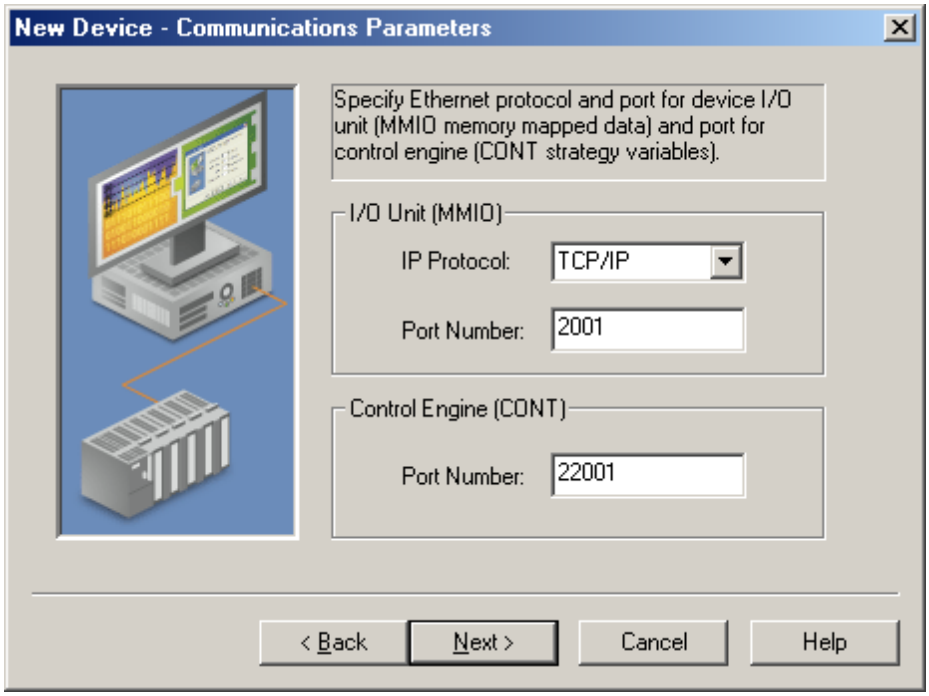
Auto-Demotion

For information on Auto-Demotion, refer to the Server Help documentation.

Database Creation

This function is used in conjunction with the driver tag import feature. For more information, refer to **Import**.

Communications Parameters



I/O Unit (MMIO)

The I/O Unit settings are used to access memory mapped data using the MMIO protocol. Descriptions of the parameters are as follows:

- **IP Protocol:** Options include TCP/IP or UDP. The default setting is TCP/IP.

  Note: Although TCP is a more reliable protocol, it requires more network overhead. Devices will accept a limited number of TCP connections. Once the limit is reached, the driver will not be able to communicate with the device using TCP.
- **Port Number:** This parameter specifies the port number that the device has been configured to use for MMIO communications. The default setting is 2001.

Control Engine (CONT)

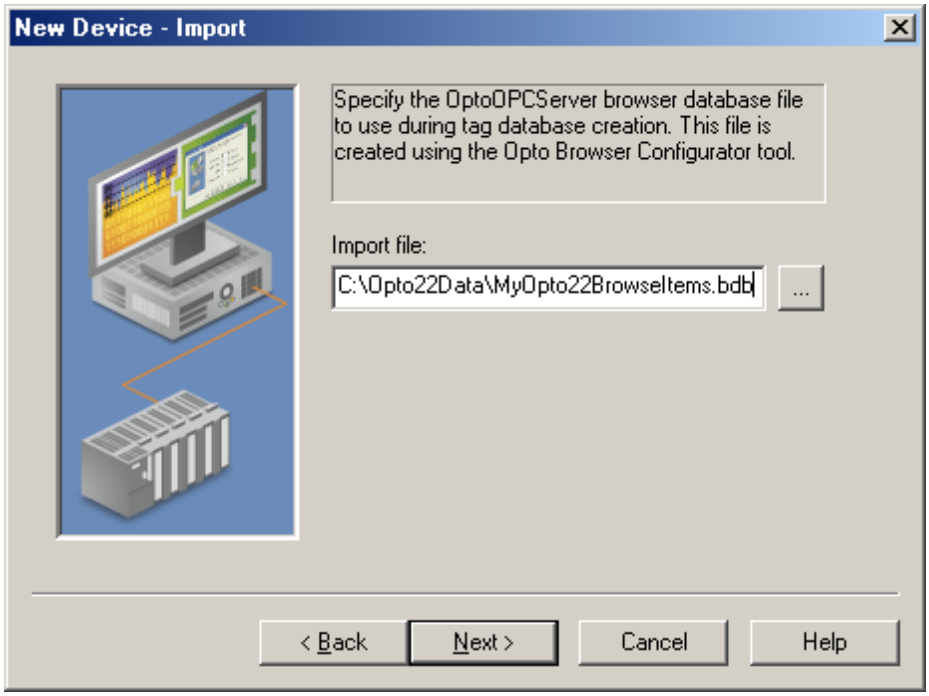
The Control Engine settings are used to access strategy variables using the CONT protocol. Description of the parameter is as follows:

- **Port Number:** This parameter specifies the port number that the device has been configured to use for CONT communications. The default setting is 22001.

Note: The PAC Control (or other similar tool) can be used to configure devices for MMIO and CONT communications. For instructions and additional information on setting up each device, refer to the Opto 22 User Guides.

Import

The Import parameter is used to specify the browser database file (*.bdb) from which tags will be imported. To browse the file system, click the Browse button.



A browser database file can be created from one or more device configuration files using the Opto Browser Configurator tool. The driver will generate tags for all supported Item IDs in the file with a location IP address that matches the driver's Device ID. The Database Creation options located in Device Properties can be used to initiate the import process. For more information on the tag import procedure, refer to **Automatic Tag Database Generation**.

See Also: **Address Descriptions** and **Device Setup**.

Cable Connections and Diagrams

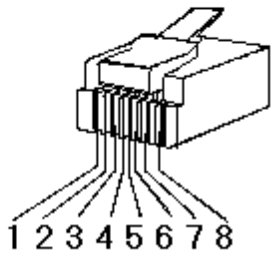
Patch Cable (Straight Through)

TD + 1	OR/WHT	OR/WHT	1	TD +
TD - 2	OR	OR	2	TD -
RD + 3	GRN/WHT	GRN/WHT	3	RD +
4	BLU	BLU	4	
5	BLU/WHT	BLU/WHT	5	
RD - 6	GRN	GRN	6	RD -
7	BRN/WHT	BRN/WHT	7	
8	BRN	BRN	8	
RJ45			RJ45	

Crossover Cable

TD + 1	OR/WHT	GRN/WHT	1	TD +
TD - 2	OR	GRN	2	TD -
RD + 3	GRN/WHT	OR/WHT	3	RD +
4	BLU	BLU	4	
5	BLU/WHT	BLU/WHT	5	
RD - 6	GRN	OR	6	RD -
7	BRN/WHT	BRN/WHT	7	
8	BRN	BRN	8	
RJ45			RJ45	

10 BaseT



8-pin RJ45

Data Types Description

The Opto 22 Ethernet Driver supports the following data types.

Data Type	Description
Boolean	Single 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
String	Null terminated ASCII string

Address Descriptions

In order to address a data location on an Opto 22 Ethernet device, follow the information and instructions below.

Prefix

All addresses may include the following prefix: *[Device|Protocol|Location]*.

Where:
Device = "MMIO" or "CONT"
Protocol = "ip"
Location = "tcp:<IP Address>:<Port>"

Note: This prefix is optional. The ability to recognize the prefix can be useful when migrating OPC client applications from other servers that use the full Opto 22 Item ID syntax. Item IDs in these applications must be modified to either replace the prefix with the appropriate path (<*channel name*>.<*device name*>) or to prepend the prefix with the path. The modified Item IDs will be recognized as valid dynamic tag addresses. For a description of dynamic tags, refer to the OPC Server Help documentation.

Memory Mapped Data (MMIO Protocol)

Basic Address Syntax
ItemName[p]

For I/O:
p = (Module) (PointsPerModule) + Point

where:
Module = zero-based module number.
Point = zero-based point within module.
PointsPerModule = 64 for xxx_4096 ItemNames (see below), and 4 for all others.

For Alarm Addresses
p = zero-based alarm index.

For Scratch Pad Addresses
p = zero-based scratch pad value index.

For PID Addresses
p = zero based PID index.

Array Address Syntax

ItemName[pStart-pEnd]

pStart = start I/O point or scratch pad value index.
pEnd = end I/O point or scratch pad value index.

Note: Arrays are limited to 1024 bytes.

HDD (High Density Digital) Address Syntax

ItemName(Module)[Point]

Module = zero-base module number.
Point = zero-based point within module.

HDD (High Density Digital) Bank Address Syntax

ItemName(Module)

Analog Point

Item Name	Description	Range	Data Type	Access	Arrays
EU[p]	Engineering units value	0-63	Float	Read/Write	Yes
COUNTS[p]	Counts	0-63	Float	Read/Write	Yes
MIN[p]	Minimum value	0-63	Float	Read	Yes
MAX[p]	Maximum value	0-63	Float	Read	Yes
MIN_READCLEAR[p]	Read then clear minimum value*	0-63	Float	Read	Yes
MAX_READCLEAR[p]	Read then clear maximum value*	0-63	Float	Read	Yes
EU_4096[p]	Engineering units value**	0-4095	Float	Read/Write	No

COUNTS_4096[p]	Counts**	0-4095	Float	Read/Write	No
MIN_4096[p]	Minimum value**	0-4095	Float	Read	No
MAX_4096[p]	Maximum value**	0-4095	Float	Read	No
MIN_READCLEAR_4096[p]	Read then clear minimum value*,**	0-4095	Float	Read	No
MAX_READCLEAR_4096[p]	Read then clear maximum value*,**	0-4095	Float	Read	No

*The device clears these values after each read. OPC clients should add these items as inactive so the driver does not poll them.

**These items require firmware 8.0 or higher.

Digital Point (4-Channel Digital Modules)

Item Name	Description	Range	Data Type	Access	Arrays
STATE[p]	On/Off state	0-63	Boolean	Read/Write	No
ONLATCH[p]	On-latch state	0-63	Boolean	Read	No
OFFLATCH[p]	Off-latch state	0-63	Boolean	Read	No
ACTIVECOUNTER[p]	Active state of counter*	0-63	Boolean	Read/Write	No
COUNTERDATA[p]	Counter value*	0-63	**Long**, DWord	Read	Yes
ONLATCH_READCLEAR[p]	Read then clear On-latch state**	0-63	Boolean	Read	Yes
OFFLATCH_READCLEAR[p]	Read then clear Off-latch state**	0-63	Boolean	Read	Yes
COUNTERDATA_READCLEAR[p]	Read then clear counter value**	0-63	**Long**, DWord	Read	Yes

*To use a digital point as a counter, it must be configured as a counter and the counter must be active.

**The device clears these values after each read. OPC clients should add these items as inactive so the driver does not poll them.

High Density Digital (HDD) Point

Item Name	Description	Range	Data Type	Access	Arrays
HDD_STATE(m)[p]	On/Off state	m: 0-15 p: 0-31	Boolean	Read/Write	No
HDD_ONLATCH(m)[p]	On-latch state	m: 0-15 p: 0-31	Boolean	Read	No
HDD_OFFLATCH(m)[p]	Off-latch state	m: 0-15 p: 0-31	Boolean	Read	No
HDD_ONLATCH_CLEAR (m)[p]	Clear On-latch state*	m: 0-15 p: 0-31	Boolean	Write	No
HDD_OFFLATCH_CLEAR (m)[p]	Clear Off-latch state*	m: 0-15 p: 0-31	Boolean	Write	No
HDD_COUNTER(m)[p]	Counter value	m: 0-15 p: 0-31	**Long**, DWord	Read	No
HDD_COUNTER_READCLEAR (m)[p]	Read then clear counter value**	m: 0-15 p: 0-31	**Long**, DWord	Read	No
HDD_BANK_STATE(m)	On/Off state for all 32 points in module***	m: 0-15 p: n/a	**Long**, DWord	Read/Write	No
HDD_BANK_ONLATCH(m)	On-latch state for all 32 points in module***	m: 0-15 p: n/a	**Long**, DWord	Read	No
HDD_BANK_OFFLATCH(m)	Off-latch state for all 32 points in module***	m: 0-15 p: n/a	**Long**, DWord	Read	No
HDD_BANK_ONLATCH_CLEAR (m)	Clear On-latch state of all 32 points of module*,****	m: 0-15 p: n/a	**Long**, DWord	Write	No
HDD_BANK_OFFLATCH_CLEAR (m)	Clear Off-latch state of all 32 points of module*,****	m: 0-15 p: n/a	**Long**, DWord	Write	No

*These items are Write Only and not a "Read and Clear." Item may be set active by client. Driver will always return zero on client read requests. Write TRUE (non-zero) to clear respective latch.

**Unlike latches, the device clears these values after each read. OPC clients should add these items as inactive so the driver does not poll them.

***The value sent is a 32 bit mask where high bits equal On and low bits equal Off.

****The value sent is a 32 bit mask where high bites equal "Clear respective latch" and low bits equal "Do nothing."

Point Configuration

Item Name	Description	Range	Data Type	Access	Arrays
MODULETYPE[p]	Module type	0-63	**Long**, DWord	Read	No
POINTTYPE[p]	Point type	0-63	**Long**, DWord	Read/Write	No
FEATURE[p]	Point feature	0-63	**Long**, DWord	Read/Write	No
OFFSET[p]	Offset value (analog calibration)	0-63	Float	Read/Write	No
GAIN[p]	Gain value (analog calibration)	0-63	Float	Read/Write	No
HISCALE[p]	Analog high scaling factor	0-63	Float	Read/Write	No
LOSCALE[p]	Analog low scaling factor	0-63	Float	Read/Write	No
MODULETYPE_4096[p]	Module type*	0-4095	**Long**, DWord	Read	No
POINTTYPE_4096[p]	Point type*	0-4095	**Long**, DWord	Read/Write	No
FEATURE_4096[p]	Point feature*	0-4095	**Long**, DWord	Read/Write	No
OFFSET_4096[p]	Offset value (analog calibration)*	0-4095	Float	Read/Write	No
GAIN_4096[p]	Gain value (analog calibration)*	0-4095	Float	Read/Write	No
HISCALE_4096[p]	Analog high scaling factor*	0-4095	Float	Read/Write	No
LOSCALE_4096[p]	Analog low scaling factor*	0-4095	Float	Read/Write	No

*These items require firmware 8.0 or higher.

Alarm

Item Name	Description	Range	Data Type	Access	Arrays
ALARM_HI_STATE[p]	High alarm state	0-63	Boolean	Read	No
ALARM_HI_ENABLE[p]	High alarm enabled state	0-63	Boolean	Read/Write	No
ALARM_HI_SETPOINT[p]	High alarm setpoint	0-63	Float	Read/Write	No
ALARM_HI_DEADBAND[p]	High alarm deadband	0-63	Float	Read/Write	No
ALARM_LO_STATE[p]	Low alarm state	0-63	Boolean	Read	No
ALARM_LO_ENABLE[p]	Low alarm enabled state	0-63	Boolean	Read/Write	No
ALARM_LO_SETPOINT[p]	Low alarm setpoint	0-63	Float	Read/Write	No
ALARM_LO_DEADBAND[p]	Low alarm deadband	0-63	Float	Read/Write	No

Scratch Pad (SNAP PAC Controllers and Ultimate Brains Only)

Item Name	Description	Range	Data Type	Access	Arrays
SP_BIT[p]	Scratch pad bit	0-63	Boolean	Read/Write	Yes
SP_INTEGER[p]	Scratch pad integer	0-1023	**Long**, DWord	Read/Write	Yes
SP_INTEGER_EXT[p]	Scratch pad integer	1024-10239	**Long**, DWord	Read/Write	Yes
SP_FLOAT[p]	Scratch pad float	0-1023	Float	Read/Write	Yes
SP_FLOAT_EXT[p]	Scratch pad float	1024-10239	Float	Read/Write	Yes
SP_STRING[p]	Scratch pad string*	0-63	String	Read/Write	No

*Scratch pad string values are limited to 128 characters. The driver will truncate write values that exceed this length.

PID

Item Name	Description	Range	Data Type	Access	Arrays
PID_CV_IN[p]	Current value: Input	0-127	Float	Read	No
PID_CV_SP[p]	Current value: Setpoint	0-127	Float	Read	No
PID_CV_OUT[p]	Current value: Output	0-127	Float	Read/Write	No
PID_CV_FF[p]	Current value: Feed forward	0-127	Float	Read/Write	No

PID_CV_ERROR[p]	Current value: Error	0-127	Float	Read	No
PID_CV_P[p]	Current value: Gain contribution	0-127	Float	Read	No
PID_CV_I[p]	Current value: Integral contribution	0-127	Float	Read	No
PID_CV_D[p]	Current value: Derivative contribution	0-127	Float	Read	No
PID_CV_INTEGRAL[p]	Current value: Integral	0-127	Float	Read	No
PID_LSV_IN[p]	Last scanned value: Input	0-127	Float	Read/Write	No
PID_LSV_SP[p]	Last scanned value: Setpoint	0-127	Float	Read/Write	No
PID_STATUS[p]	Status flags	0-127	**Long**, DWord	Read/Write	No
PID_STATUS_ON[p]	Status flags On mask	0-127	**Long**, DWord	Read/Write	No
PID_STATUS_OFF[p]	Status flags Off mask	0-127	**Long**, DWord	Read/Write	No
PID_TUNE_P[p]	Tuning: Proportional value	0-127	Float	Read/Write	No
PID_TUNE_I[p]	Tuning: Integral value	0-127	Float	Read/Write	No
PID_TUNE_D[p]	Tuning: Derivative value	0-127	Float	Read/Write	No
PID_TUNE_FF[p]	Tuning: Feed forward gain	0-127	Float	Read/Write	No
PID_CFG_MAX_OUT[p]	Configuration: Max output change allowed	0-127	Float	Read/Write	No
PID_CFG_MIN_OUT[p]	Configuration: Min output change allowed	0-127	Float	Read/Write	No
PID_CFG_SCAN_TIME[p]	Configuration: Scan time in seconds	0-127	Float	Read/Write	No
PID_CFG_LOW_RANGE[p]	Configuration: Output when input is low	0-127	Float	Read/Write	No
PID_CFG_HI_RANGE[p]	Configuration: Output when input is high	0-127	Float	Read/Write	No
PID_CFG_ALG[p]	Configuration: Algorithm	0-127	**Long**, DWord	Read/Write	No
PID_CFG_MAN_MODE[p]	Configuration: Manual mode 1=Yes, 0=No	0-127	**Long**, DWord	Read/Write	No
PID_CFG_FLAGS[p]	Configuration: Flags	0-127	**Long**, DWord	Read/Write	No
PID_CFG_FLAGS_ON[p]	Configuration: Flags On mask	0-127	**Long**, DWord	Read/Write	No
PID_CFG_FLAGS_OFF[p]	Configuration: Flags Off mask	0-127	**Long**, DWord	Read/Write	No
PID_CFG_MM_IN[p]	Configuration: Input mem map address	0-127	**Long**, DWord	Read/Write	No
PID_CFG_MM_SP[p]	Configuration: Setpoint mem map address	0-127	**Long**, DWord	Read/Write	No
PID_CFG_MM_OUT[p]	Configuration: Output mem map address	0-127	**Long**, DWord	Read/Write	No
PID_SCALE_IN_LOW[p]	Scaling: Input low range	0-127	Float	Read/Write	No
PID_SCALE_IN_HI[p]	Scaling: Input high range	0-127	Float	Read/Write	No
PID_SCALE_OUT_LOW[p]	Scaling: Output lower clamp	0-127	Float	Read/Write	No
PID_SCALE_OUT_HI[p]	Scaling: Output upper clamp	0-127	Float	Read/Write	No
PID_SCAN_COUNTER[p]	Scan counter	0-127	**Long**, DWord	Read/Write	No

Examples

Address	Description
[MMIO|ip|tcp:10.10.110.126:2001]EU[4]	EU value of point 0 in module 1. Address includes optional prefix.
EU[4]	EU value of point 0 in module 1.
EU_4096[64]	EU value of point 0 in module 1.
EU[4-7]	Array with EU value of points 0, 1, 2, 3 in module 1.

STATE[9]	State of point 1 in module 2.
HDD_STATE(3)[0]	State of point 0 in HDD module 3.
HDD_BANK_STATE(3)	State of points 0 through 31 in HDD module 3. Value returned as a 32 bit integer, LSB = point 0.
ALARM_HI_STATE[5]	High state of alarm 5.
SP_INTEGER[4]	Scratch pad integer value 4.
PID_CV_IN[10]	Current value input of PID 10.

Strategy Variables (CONT Protocol)

Basic Address Syntax
DataType;Property;VariableName

Table Element Address Syntax
DataType;Property[ElementNumber]

Table Element Array Address Syntax
DataType;Property[StartElementNumber-EndElementNumber]

Note: Arrays are limited to 1024 bytes.

Bit Address Syntax
DataType;Property.BitNumber;VariableName

Note: This syntax may be used for 32 bit integer values (I32) and single 32 bit integer table elements (I32T) only. Arrays of bit in integer values are not allowed.

Variables

Data Type (Mnemonic)	Property	Description	Data Type	Access
I32	VALUE	32 bit integer value	**Long**, DWord	Read/Write
I64	VALUE	64 bit integer value*	**Long**, DWord	Read/Write
F	VALUE	Float value	Float	Read/Write
S	VALUE	String value**	String	Read/Write
T	VALUE	Timer value	Float	Read/Write

*64 bit integer values are represented as a 2-element Long or DWord array.

**String values are limited to 1024 characters. The driver will truncate write values that exceed this length.

Tables

Data Type (Mnemonic)	Property	Description	Data Type	Access
I32T	VALUE	Table of 32-bit integer values	**Long**, DWord	Read/Write
I64T	VALUE	Table of 64 bit integer values*	**Long**, DWord	Read/Write
FT	VALUE	Table of float values	Float	Read/Write
ST	VALUE	Table of string values**,***	String	Read/Write

*64 bit integer values are represented as a 2-element Long or DWord array. An n-element array of 64 bit integer values is represented as a 2n-element Long or DWord array.

**String values are limited to 1024 characters. The driver will truncate write values that exceed this length.

***This driver does not support arrays of string table elements.

Analog Points

Data Type (Mnemonic)	Property	Description	Data Type	Access
APOINT	EU	Engineering Units value*	Float	Read/Write

*Read/Write to XVAL (external value) if communications are enabled for I/O unit and point. Read/Write to IVAL (internal value) otherwise.

Digital Points

Data Type (Mnemonic)	Property	Description	Data Type	Access
DPOINT	STATE	On/Off State*	Boolean	Read/Write

*Read/Write to XVAL (external value) if communications are enabled for I/O unit and point. Read/Write to IVAL (internal value) otherwise.

Pointers

The driver can read/write data referenced by pointer variables by pre-pending "PTR_" to the DataType. Pointer tables are not supported. Supported pointer types include the following:

PTR_I32
PTR_I64
PTR_F
PTR_S
PTR_T
PTR_APOINT
PTR_DPOINT

Examples

Address	Description
[CONT|ip|tcp:10.10.110.126:22001]F;VALUE;MyFloatVariable	Read/Write value of a floating point variable named "MyFloatVariable." Address includes optional prefix.
F;VALUE;MyFloatVariable	Read/Write value of a floating point variable named "MyFloatVariable."
I32;VALUE;MyIntegerVariable	Read/Write value of a 32-bit integer variable named "MyIntegerVariable."
I32;VALUE.0;MyIntegerVariable	Read/Write bit 0 (LSB) of 32-bit integer variable named "MyIntegerVariable."
I32T;VALUE[0];MyIntegerTable	Read/Write value of element 0 of a 32-bit integer table called "MyIntegerTable."
I32T;VALUE[1].0;MyIntegerTable	Read/Write bit 0 of table element 1 value.
I32T;VALUE[0-3];MyIntegerTable	Read/Write value of table elements 0, 1, 2, and 3.
PTR_I32;VALUE;pMyIntegerPointer	Read/Write value of 32-bit integer referenced by the pointer variable named "pMyIntegerPointer."

Automatic Tag Database Generation

The Opto 22 Ethernet Driver can import items from a browser database file (*.bdb). Items must meet the following criteria for import:

- The IP must match the Device ID.
- The item must have supported protocol (MMIO or CONT).
- The address must be supported.*
- The array size must be compatible.*
- The data type must be compatible.**
- Read and Clear MMIO addresses are not imported, but can be created manually.

*For more information, refer to **Address Descriptions**.

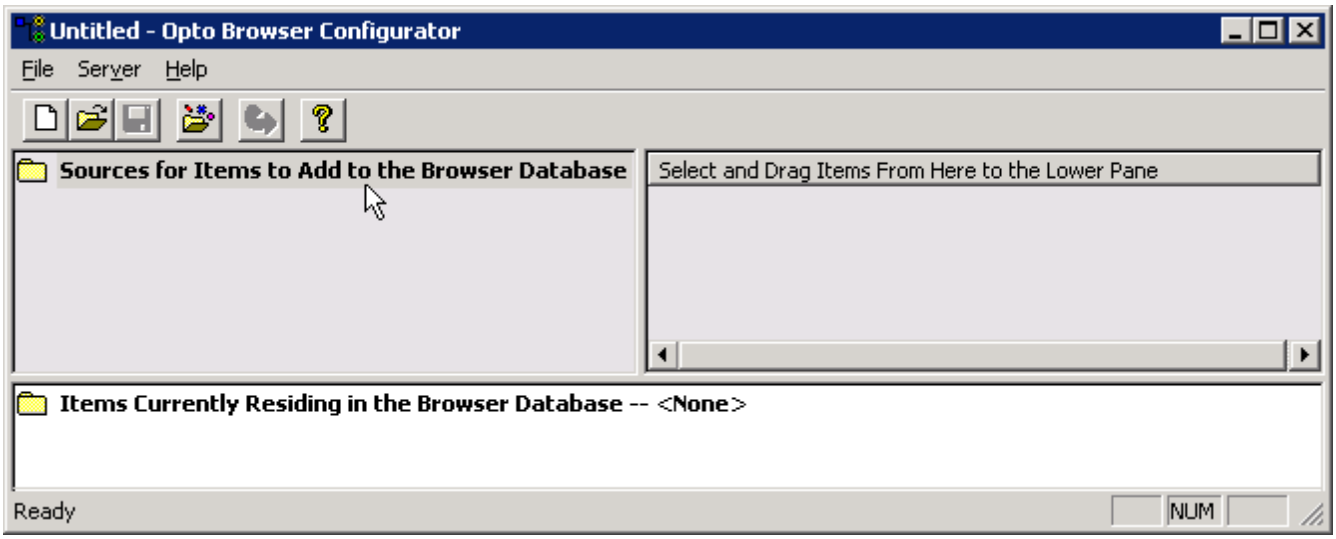
**For example, the driver does not support string arrays.

Creating a Browser Database File

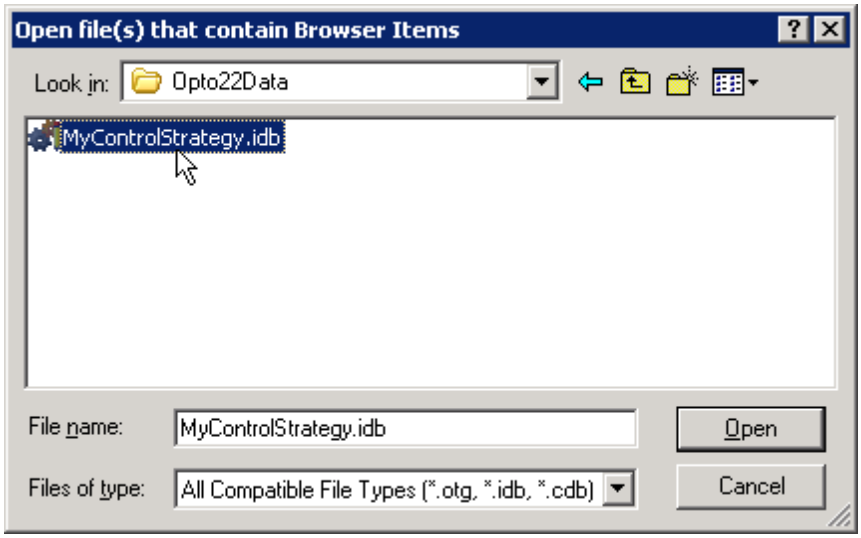
A browser database file can be created from one or more device configuration files using the Opto Browser Configurator tool. Follow the instructions below for directions on how to create a browser database file.

Note: The images below are from the Opto Browser Configurator tool, version R8.2a.

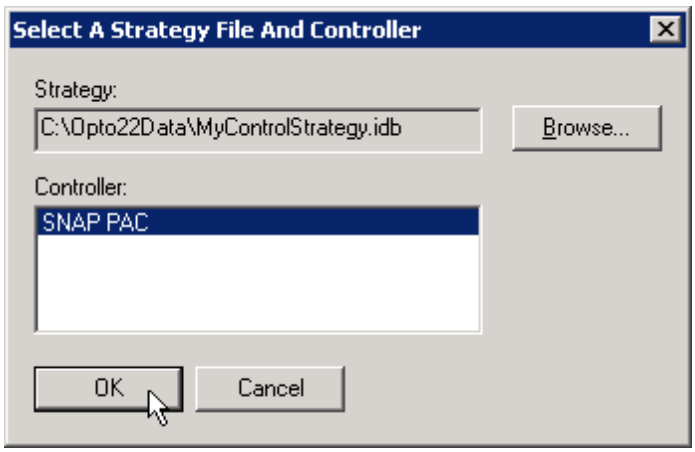
1. Start a new project in the **Browser Configurator**.



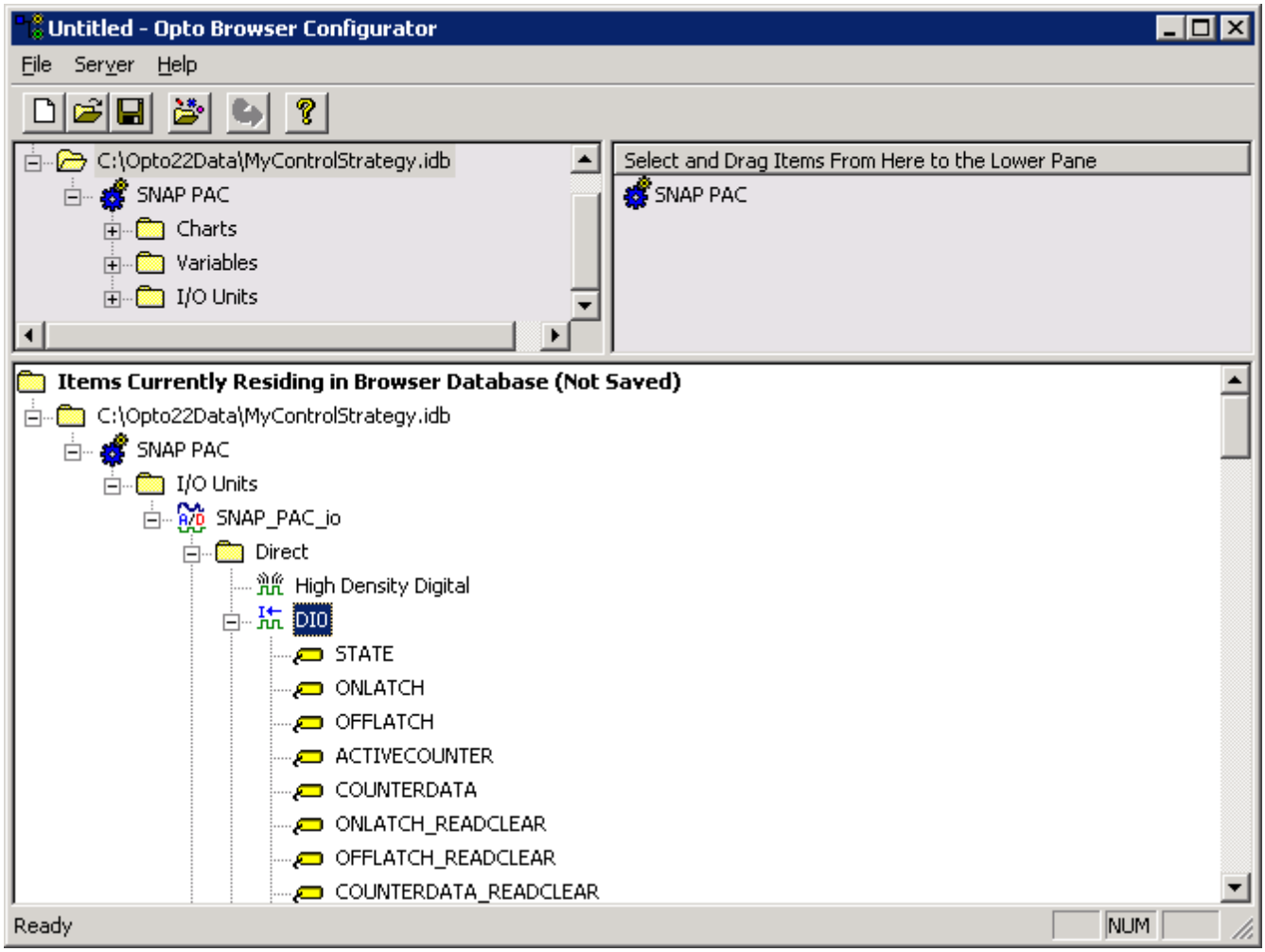
2. Open one or more Browser Item Files created with **PAC Manager** (*.otg), **PAC Control** (*.idb), or **Opto-Control** (*.cdb).



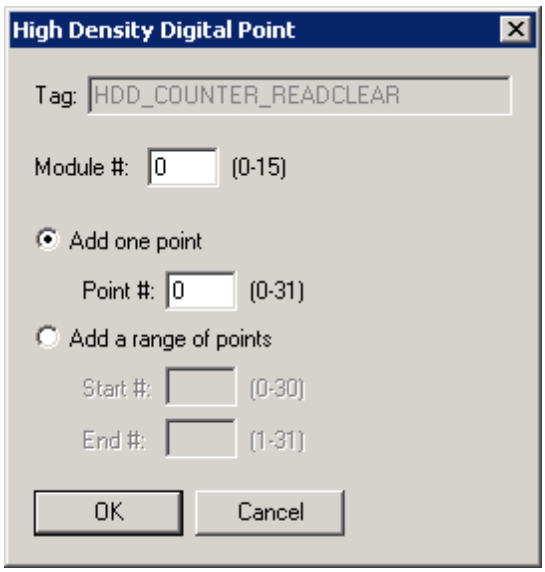
3. If an .idb or .cdb file was opened, the controllers will need to be specified.



4. At this point, the browse items should be visible in the upper-left panel. The right panel will show which specific tags are associated with the browser items selected on the left panel. To include tags in the database, drag and drop them from the right panel to the bottom panel.



5. If including High Density Digital (HDD) module items, additional information must be specified. Right-click on the item and select **Add to Browser Database...**.



6. Next, save the browser database.

Importing Browser Items

The driver can automatically generate tags for items in the browser database file. To do so, follow the instructions below.

1. In the specific device's Device Properties, click on the **Import** tab.

2. Specify the name and path to the database file. Then, click **Apply**.

3. In Device Properties, click on the **Database Creation** tab. The driver can be configured to import these tags every time the driver starts, thus automatically keeping the driver configuration up to date with changes made to the database file. It is, however, usually sufficient to import the tags once by clicking **Auto Create** | **OK**. For more information on database creation options, refer to the OPC Server Help documentation.

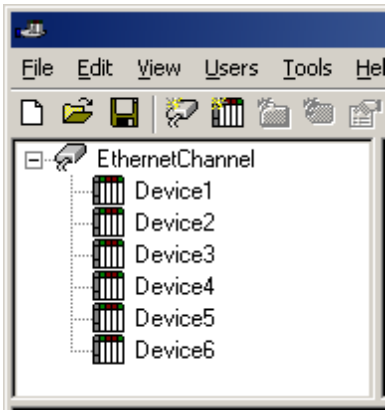
4. Once the automatic tag generation feature is triggered, the driver will open the specified browser database file and create a tag for each supported item with location IP addresses that match the driver's Device ID.

See Also: **Address Descriptions**

Optimizing Your Opto 22 Ethernet Communications

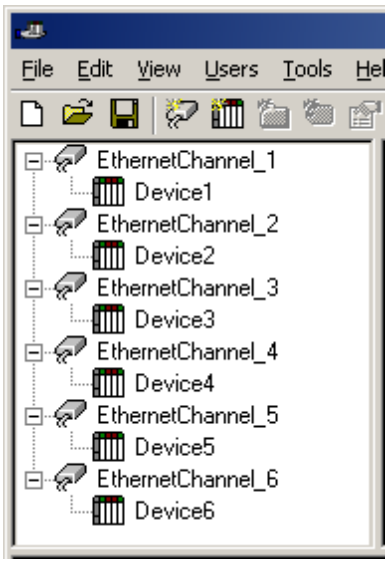
The Opto 22 Ethernet Driver has been designed to provide the best performance with the least amount of impact on the system's overall performance. While the Opto 22 Ethernet Driver is fast, there are a couple of guidelines that can be used in order to control and optimize the application and gain maximum performance.

This server refers to communications protocols like Opto 22 Ethernet Device as a channel. Each channel defined in the application represents a separate path of execution in the server. Once a channel has been defined, a series of devices must then be defined under that channel. Each of these devices represents a single 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 Opto 22 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 Opto 22 Ethernet Device channel. In this configuration, the driver must move from one device to the next as quickly as possible in order to gather information at an effective rate. As more devices are added or more information is requested from a single device, the overall update rate begins to suffer.

If the Opto 22 Ethernet Driver could only define one single channel, then the example shown above would be the only option available; however, the Opto 22 Ethernet Driver can define up to 256 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 256 or fewer devices, it can be optimized exactly how it is shown here.

The performance will improve even if the application has more than 256 devices. While 256 or fewer devices may be ideal, the application will still benefit from additional channels. Although by spreading the device load across all channels will cause the server to move from device to device again, it can now do so with far less devices to process on a single channel.

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's 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>'
Array support is not available for the specified address: '<address>'
Data Type'<type>' is not valid for the device address '<address>
Device address '<address>' is Read Only

Device Status Messages

Device '<channel.device>' not responding to requests on I/O Unit (MMIO) port
Device '<channel.device>' not responding to requests on Control Engine (CONT) port
Unable to write to '<address>' on device '<device name>'

Device Specific Messages

Winsock initialization failed (OS Error = <n>)
Winsock V1.1 or higher must be installed to use the Opto 22 Ethernet Driver
Unable to bind to adapter: '<adapter name>'. Connect failed
Powerup clear failed for device '<device name>'. <Protocol> response code: <n>
Write failed for tag '<tag name>' on device '<device name>'. <Protocol> response code: <n>
Read failed for tag '<tag name>' on device '<device name>'. <Protocol> response code: <n>
Block read failed for <n> bytes starting at <memory map offset> on device '<device name>'. <Protocol> response code: <n>
Read failed for tag '<tag name>' on device '<device name>'. Object appears to be invalid in current strategy
Read failed for tag '<tag name>' on device '<device name>'. Unexpected CONT data format
Read request failed for multiple tags on device '<device name>'. CONT response code: <n>

Import Error Messages

Did not import one or more items with incompatible protocol or IP
No compatible items found in import file
Did not import item '<address>' at record <record> - arrays not supported for specified address
Did not import item '<address>' at record <record> - Read and Clear tags must be manually created
Error parsing import file record number n, field f
Did not import item '<address>' at record <record> - unsupported array size

Response Codes

MMIO Response Codes
CONT Response Codes

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's 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>'
Array support is not available for the specified address: '<address>'
Data Type'<type>' is not valid for the device address '<address>
Device address '<address>' is Read Only

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 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 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.

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.

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.

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 '<channel.device>' not responding to requests on I/O Unit (MMIO) port
Device '<channel.device>' not responding to requests on Control Engine (CONT) port
Unable to write to '<address>' on device '<device name>'

Device '<channel.device>' not responding to requests on I/O Unit (MMIO) port

Error Type:
Serious

Possible Cause:
1. The MMIO port number configured in the device may not match the port configured in the hardware.
2. The device cannot accept any more TCP connections.

Solution:
1. Verify the port configuration and change as needed.
2. Eliminate other applications and hardware that have made unnecessary TCP connections to the device. Alternatively, configure the driver to UDP.

See Also:
Communications Parameters

Device '<channel.device>' not responding to requests on Control Engine (CONT) port

Error Type:
Serious

Possible Cause:
The CONT port number configured in the device may not match the port configured in the hardware.

Solution:
Verify the port configuration and change as needed.

See Also:
Communications Parameters

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

Error Type:
Serious

Possible Cause:
1. The network connection between the device and the host PC is broken.
2. The communications parameters configured for the device and driver do not match.

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

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 communications.	Wait a few seconds and restart the driver.
10067	Limit on the number of tasks supported by the Windows Sockets implementation has been reached.	Close one or more applications that may be using Winsock and restart the driver.

Winsock V1.1 or higher must be installed to use the Opto 22 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.

Unable to bind to adapter: '<adapter name>'. Connect failed

Error Type:
Fatal

Possible Cause:
The driver was unable to bind to the specified network adapter, which is necessary for communications with the device.

Reasons:
1. Adapter is disabled or no longer exists.
2. Network system failure, such as Winsock or network adapter failure.
3. No more available ports.

Solution:
1. Check the Channel Properties | Network Interface | Network Adapter list in the Communications Server application for network adapters available on the system. If '<adapter>' is not in this list, steps should be taken to make it available to the system. This includes but is not limited to: verifying that the network connection is enabled and connected in the PC's Network Connections.

2. Determine how many channels are using the same '<adapter>' in the communications server application. Reduce this number so that only one channel is referencing '<adapter>'. If the error still occurs, check to see if other applications are using that adapter and shut down those applications

Powerup clear failed for device'<device name>'. <Protocol> response code: <n>

Error Type:
Serious

Possible Cause:
The device was restarted and requires the driver to send a "Powerup Clear' message before data access is permitted.

Solution:
The problem may correct itself as the driver issues new requests. Additional actions will depend on the reported error code. Refer to the response code for the specific reason of failure.

Note:
If the Protocol is MMIO, refer to **MMIO Response Codes**. If Protocol is CONT, refer to **CONT Response Codes**.

Write failed for tag '<tag name>' on device '<device name>'. <Protocol> response code: <n>

Error Type:
Warning

Possible Cause:
The device rejected a write request. Refer to the response code for the specific reason why.

Solution:
The problem may correct itself as the driver issues new requests. Additional actions will depend on reported error code.

Note:
If Protocol is MMIO, see **MMIO Response Codes**. If Protocol is CONT, see **CONT Response Codes**.

Read failed for tag '<tag name>' on device '<device name>'. <Protocol> response code: <n>

Error Type:
Warning

Possible Cause:
The device rejected a read request for a single tag. Refer to the response code for the specific reason why.

Solution:
The problem may correct itself as the driver issues new requests. Additional actions will depend on reported error code.

Note:
If Protocol is MMIO, see **MMIO Response Codes**. If Protocol is CONT, see **CONT Response Codes**.

Block read failed for <n> bytes starting at <memory map offset> on device '<device name>'. <Protocol> response code: <n>

Error Type:
Warning

Possible Cause:
The device rejected a read request for a block of data that would update multiple tags. Refer to the response code for the specific reason why.

Solution:

The problem may correct itself as the driver issues new requests. Additional actions will depend on reported error code.

Note:
If Protocol is MMIO, see **MMIO Response Codes**. If Protocol is CONT, see **CONT Response Codes**.

Read failed for tag '<tag name>' on device '<device name>'. Object appears to be invalid in current strategy

Error Type:
Warning

Possible Cause:
The tag addresses a strategy variable that is not defined in the device control strategy.

Solution:
1. Correct the address or remove the tag.
2. Define a new strategy variable.

Read failed for tag '<tag name>' on device '<device name>'. Unexpected CONT data format

Error Type:
Serious

Possible Cause:
The driver received a response to a strategy variable Read or Write that did not have the expected length or format.

Solution:
Contact Technical Support.

Read request failed for multiple tags on device '<device name>'. CONT response code: <n>

Error Type:
Warning

Possible Cause:
The device rejected a read request for multiple tags that address strategy variables. Refer to the response code for specific reason why.

Solution:
The problem may correct itself as the driver issues new requests. Additional actions will depend on the reported error code.

See Also:
CONT Response Codes

Import Error Messages

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

Import Error Messages
Did not import one or more items with incompatible protocol or IP
No compatible items found in import file
Did not import item '<address>' at record <record> - arrays not supported for specified address
Did not import item '<address>' at record <record> - Read and Clear tags must be manually created
Error parsing import file record number n, field f
Did not import item '<address>' at record <record> - unsupported array size

Did not import one or more items with incompatible protocol or IP

Error Type:

Warning

Possible Cause:
1. The import file contained items for a device with an IP other than that configured for the Device ID.
2. The import file contained items for a protocol that is not supported by this driver.

Solution:
This driver will only import those items with a matching IP address. Make sure the IP specified as the Device ID matches the desired items in the import file.

Note:
It is possible for bdb files to contain data for more than one device. In these cases, it is normal for the driver to log this message.

See Also:
Device Setup

No compatible items found in import file

Error Type:
Warning

Possible Cause:
An item must meet the following criteria for import:

1. The IP must match the Device ID.
2. The item must have supported protocol (MMIO or CONT).
3. The address must be supported.
4. The array size must be compatible.
5. The data type must be compatible.
6. Read and Clear MMIO addresses are not imported, but can be created manually.

Solution:
Check each of the import criteria above.

Note:
The most common reason when tags are not imported is an IP mismatch.

See Also:
Address Descriptions
Automatic Tag Database Generation

Did not import item '<address>' at record <record> - arrays not supported for specified address

Error Type:
Warning

Possible Cause:
The driver does not support arrays for certain types (such as strings).

Solution:
Create tags for individual array elements. For example, although the driver would not allow an array of CONT string table elements, individual tags for each table element are permitted.

Did not import item '<address>' at record <record> - Read and Clear tags must be manually created

Error Type:
Warning

Possible Cause:
The driver will not import Read and Clear MMIO items since they must be used cautiously and must not be set active in OPC clients. These include the following:

MIN_READCLEAR
MAX_READCLEAR
MIN_READCLAR_4096
MAX_READCLEAR_4096
ONLATCH_READCLEAR
OFFLATCH_READCLEAR
COUNTERDATA_READCLEAR
HDD_COUNTER_READCLEAR

Solution:
Tags for these addresses may be created manually.

Error parsing import file record number n, field f

Error Type:
Warning

Possible Cause:
1. The import file may have been modified and an invalid entry was created.
2. The import file format may have changed since the driver was created.

Solution:
1. Correct or recreate the import file.
2. Contact Technical Support.

Did not import item '<address>' at record <record> - unsupported array size

Error Type:
Warning

Possible Cause:
The driver does not support arrays larger than 1024 bytes.

Solution:
Create multiple arrays that are smaller in size.

Response Codes

For more information on a specific type of Response Code, click a link from the list below.

MMIO Response Codes
CONT Response Codes

MMIO Response Codes

Code	Meaning
1	Undefined command
2	Invalid point type
3	Invalid float value
4	Powerup Clear expected
5	Invalid memory address or invalid data for the memory address
6	Invalid command length
7	Reserved
8	Busy
9	Cannot erase flash
10	Cannot program flash
11	Downloaded images too small
12	Image CRC mismatch
13	Image length mismatch
14	Feature is not yet implemented
15	Communications watchdog timeout

CONT Response Codes

Code	Meaning
-3	Buffer overrun or invalid length
-4	Powerup clear expected
-5	Operation failed
-6	Data field error
-7	Communications watchdog timeout
-8	Invalid data
-12	Invalid table index
-14	Invalid number
-17	Port locked - strategy download in progress
-20	Device busy
-22	Command not valid on specified I/O unit
-28	Object not found
-29	Wrong object type – most likely pointer type mismatch
-30	Pointer not initialized

Index