OPTCQ EtherNet/IP Option Card

User Manual

Effective August 2012
New Information

Powering Business Worldwide

Disclaimer of Warranties and Limitation of Liability

The information, recommendations, descriptions and safety notations in this document are based on Eaton Corporation's ("Eaton") experience and judgment and may not cover all contingencies. If further information is required, an Eaton sales office should be consulted. Sale of the product shown in this literature is subject to the terms and conditions outlined in appropriate Eaton selling policies or other contractual agreement between Eaton and the purchaser.

THERE ARE NO UNDERSTANDINGS, AGREEMENTS, WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING WARRANTIES OF FITNESS FOR A PARTICULAR PURPOSE OR MERCHANTABILITY, OTHER THAN THOSE SPECIFICALLY SET OUT IN ANY EXISTING CONTRACT BETWEEN THE PARTIES. ANY SUCH CONTRACT STATES THE ENTIRE OBLIGATION OF EATON. THE CONTENTS OF THIS DOCUMENT SHALL NOT BECOME PART OF OR MODIFY ANY CONTRACT BETWEEN THE PARTIES.

In no event will Eaton be responsible to the purchaser or user in contract, in tort (including negligence), strict liability or other-wise for any special, indirect, incidental or consequential damage or loss whatsoever, including but not limited to damage or loss of use of equipment, plant or power system, cost of capital, loss of power, additional expenses in the use of existing power facilities, or claims against the purchaser or user by its customers resulting from the use of the information, recommendations and descriptions contained herein. The information contained in this manual is subject to change without notice.

Support Services

The goal of Eaton is to ensure your greatest possible satisfaction with the operation of our products. We are dedicated to providing fast, friendly, and accurate assistance. That is why we offer you so many ways to get the support you need. Whether it's by phone, fax, or e-mail, you can access Eaton's support information 24 hours a day, seven days a week. Our wide range of services is listed below.

You should contact your local distributor for product pricing, availability, ordering, expediting, and repairs.

Web Site

Use the Eaton Web site to find product information. You can also find information on local distributors or Eaton's sales offices.

Web Site Address

www.eaton.com/drives

EatonCare Customer Support Center

Call the EatonCare Support Center if you need assistance with placing an order, stock availability or proof of shipment, expediting an existing order, emergency shipments, product price information, returns other than warranty returns, and information on local distributors or sales offices.

Voice: 877-ETN-CARE (877-386-2273) (8:00 a.m.–6:00 p.m. Eastern Time U.S. [UTC –5])

FAX: 800-752-8602

After-Hours Emergency: 800-543-7038
(6:00 p.m.–8:00 a.m. Eastern Time U.S. [UTC –5])

If you are in the U.S. or Canada, and have OI or PLC questions, you can take advantage of our toll-free line for technical assistance with hardware and software product selection, system design and installation, and system debugging and diagnostics. Technical support engineers are available for calls during regular business hours.

Drives Technical Resource Center

Voice: 800-322-4986 or +1 828-651-0984 (8:00 a.m.–5:00 p.m. Central Time U.S. [UTC –6])
Fax: +1 920-262-6070
e-mail: TRCDrives@Eaton.com

Table of Contents

List of Figures

List of Tables

Safety

Definitions and Symbols

WARNING

This symbol indicates high voltage. It calls your attention to items or operations that could be dangerous to you and other persons operating this equipment. Read the message and follow the instructions carefully.

This symbol is the "Safety Alert Symbol." It occurs with either of two signal words: CAUTION or WARNING, as described below.

WARNING

Indicates a potentially hazardous situation which, if not avoided, can result in serious injury or death.

CAUTION

Indicates a potentially hazardous situation which, if not avoided, can result in minor to moderate injury, or serious damage to the product. The situation described in the CAUTION may, if not avoided, lead to serious results. Important safety measures are described in CAUTION (as well as WARNING).

Hazardous High Voltage

WARNING

Motor control equipment and electronic controllers are connected to hazardous line voltages. When servicing drives and electronic controllers, there may be exposed components with housings or protrusions at or above line potential. Extreme care should be taken to protect against shock.

Stand on an insulating pad and make it a habit to use only one hand when checking components. Always work with another person in case an emergency occurs. Disconnect power before checking controllers or performing maintenance. Be sure equipment is properly grounded. Wear safety glasses whenever working on electronic controllers or rotating machinery.

Warnings and Cautions

WARNING

Internal components and circuit boards are at high potential when the frequency converter is connected to the power source. This voltage is extremely dangerous and may cause death or severe injury if you come into contact with it.

WARNING

MAKE SURE THAT THE FREQUENCY CONVERTER IS SWITCHED OFF BEFORE AN OPTION OR FIELDBUS BOARD IS CHANGED OR ADDED!

Notice

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

General Information

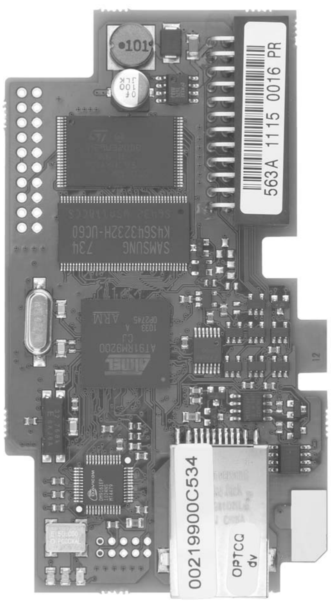
Eaton 9000X variable frequency drives can be connected to Ethernet using an EtherNet/IP fieldbus board OPTCQ.

OPTCQ EtherNet/IP Communication Interface Option Card

Features

- Provides a means to control, configure and collect data over an Ethernet network
- 10/100 Mbps, full duplex operation
- Explicit messaging (for example, parameter read/write)
- Diagnostics, device items and events

The OPTCQ EtherNet/IP communication interface option card features standard EtherNet/IP communication, allowing you to easily manage drive control and data over EtherNet/IP networks.



Flash Upgradeable

The OPTCQ EtherNet/IP communication interface option card can be flash updated in the field to take advantage of new firmware features as they are made available.

The OPTCQ can be installed in the card slots D or E.

Every appliance connected to an Ethernet network has two identifiers: a MAC address and an IP address. The MAC address (address format: 00:21:99:xx:yy:zz) is unique to the appliance and cannot be changed. The EtherNet/IP board's MAC address can be found on the sticker attached to the board. Please find the software installation at www.Eaton.com/drives.

In a local network, IP addresses are determined by the network server using DHCP protocol. The user can also manually define the network address for the OPTCQ as long as all units connected to the network are given the same network portion of the address. For more information about IP addresses, contact your network administrator. Overlapping IP addresses can cause conflicts between appliances. For more information about setting IP addresses, see Installation on **Page 5**.

Note: EtherNet/IP is a trademark of the Open DeviceNet Vendor Association (ODVA).

WARNING

Internal components and circuit boards are at high potential when the frequency converter is connected to the power source. This voltage is extremely dangerous and may cause death or severe injury if you come into contact with it.

EtherNet/IP Board Technical Data

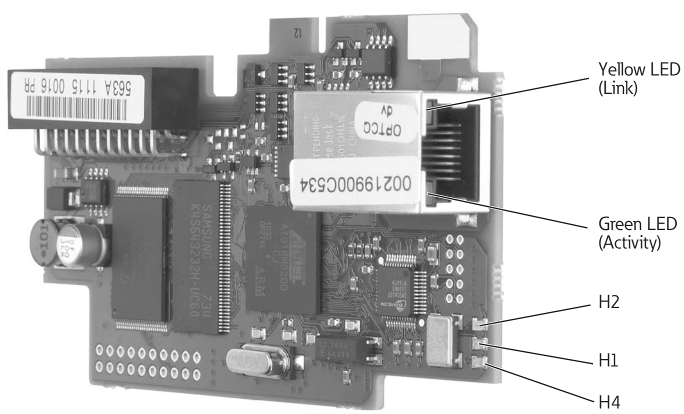
Overview

EtherNet/IP Board Technical Data

Description		Specification
General	Card Name	OPTCQ
Ethernet connections	Interface	RJ-45 connector
Communications	Transfer cable	Shielded twisted pair
	Speed	10/100 Mb
	Duplex	Half/full
	Default IP–address	192.168.0.10
Protocols	EtherNet/IP	—
Environment	Ambient operating temperature	–10°C to 50°C
	Storing temperature	–40°C to 70°C
	Humidity	<95%, no condensation allowed
	Altitude	Max. 1000m
	Vibration	0.5G at 9 to 200 Hz
Safety		Fulfills EN50178 standard

LED Indications

OPTCQ EtherNet/IP Communication Interface Option Card



LED Description

LED	Meaning
H4	LED in ON when board is powered
H1	Blinking 0.25s ON/0.25s OFF when board firmware is corrupted (see note on **Page 37**). OFF when board is operational.
H2	Blinking 2.5s ON/2.5s OFF when board is ready for external communication. OFF when board is not operational.
Activity	Flashes with Ethernet message activity
Link	Indicates connected in 100 Mbps port

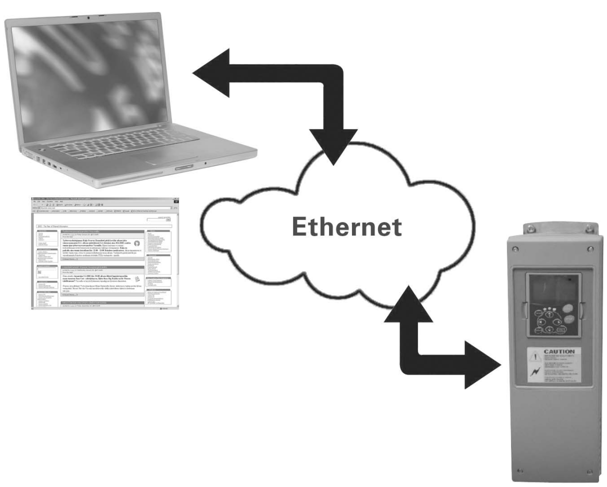
EtherNet/IP

EtherNet/IP was introduced in 2001 and today is the most developed, proven, and complete industrial Ethernet network solution available for manufacturing automation. EtherNet/IP is a member of a family of networks that implements the Common Industrial Protocol (CIP™) at its upper layers. CIP encompasses a comprehensive suite of messages and services for a variety of manufacturing automation applications, including control, safety, synchronization, motion, configuration, and information. As a truly media-independent protocol that is supported by hundreds of vendors around the world, CIP provides users with a unified communication architecture throughout the manufacturing enterprise.

There are two common use cases of Ethernet—devices are “human to machine” and “machine to machine.” Basic features are presented in the pictures below.

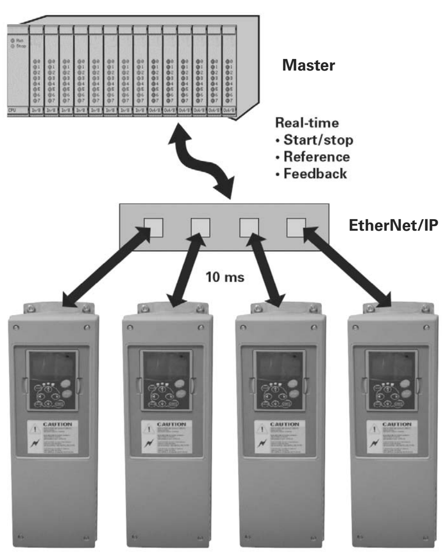
1. Human to machine (graphical user interface, relatively slow communication)

User Interface



2. Machine to machine (industrial environment, fast communication)

Industrial Environment



Connections and Wiring

The EtherNet/IP board supports 10/100 Mb speeds in both full and half-duplex modes. The boards must be connected to the Ethernet network with a shielded CAT-5e cable. A crossover cable (at least CAT-5e cable with STP, shielded twisted pair) may be needed if you want to connect the EtherNet/IP option board directly to the master appliance.

Use only industrial standard components in the network and avoid complex structures to minimize the length of response time and the amount of incorrect dispatches.

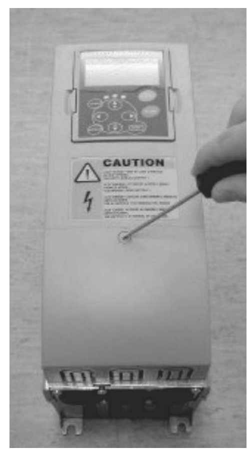
It is often a good practice to use a subnet that is different from other devices not related to the drive control.

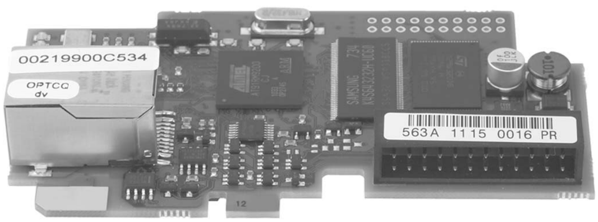
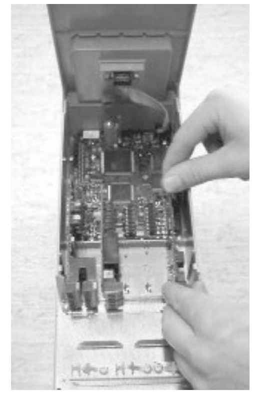
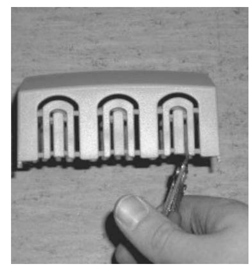
Installation

Installing the EtherNet/IP Option Board in an Eaton 9000X Variable Frequency Drive

⚠ WARNING

MAKE SURE THAT THE FREQUENCY CONVERTER IS SWITCHED OFF BEFORE AN OPTION OR FIELDBUS BOARD IS CHANGED OR ADDED!

Step	Example
1. Eaton 9000X variable frequency drives.	
2. Remove the cable cover.	
3. Open the cover of the control unit.	

Step		Example
4. Install EtherNet/IP option board in slot D or E on the control board of the frequency converter. Make sure that the grounding plate (right) fits tightly in the clamp.		
5. Make an opening that is wide enough for your cable by cutting the grid as wide as necessary.		
6. Close the cover of the control unit and re-install the cable cover.		

Commissioning

The OPTCQ EtherNet/IP board is commissioned with the control keypad by giving values to appropriate parameters in menu M7 (or with NCIPConfig tool, read chapter IP Tool NCIPConfig). Keypad commissioning is only possible with SVX9000- and SPX9000-type frequency converters.

Expander Board Menu (M6)

The Expander board menu makes it possible for the user to see what expander boards are connected to the control board and to reach and edit the parameters associated with the expander board.

Enter the following menu level (G#) with the Menu Button Right. At this level, you can browse through slots A to E with the Browser buttons to see what expander boards are connected. On the lowermost line of the display, you can see the number of parameter groups associated with the board. If you still press the Menu Button Right once, you will reach the parameter group level where there is one group in the EtherNet/IP board case: Parameters. A further press on the Menu Button Right takes you to Parameter group.

EtherNet/IP Parameters

Number	Name	Default	Range	Description
1	Comm. Timeout	10s	0–255s	0 = Not used
2	IP Part 1	192	1–223	IP Address Part 1 ①
3	IP Part 2	168	0–255	IP Address Part 2 ①
4	IP Part 3	0	0–255	IP Address Part 3 ①
5	IP Part 4	10	0–255	IP Address Part 4 ①
6	SubNet Part 1	255	0–255	Subnet Mask Part 1 ①
7	SubNet Part 2	255	0–255	Subnet Mask Part 2 ①
8	SubNet Part 3	0	0–255	Subnet Mask Part 3 ①
9	SubNet Part 4	0	0–255	Subnet Mask Part 4 ①
10	DefGW Part 1	192	0–255	Default Gateway Part 1 ①
11	DefGW Part 2	168	0–255	Default Gateway Part 2 ①
12	DefGW Part 3	0	0–255	Default Gateway Part 3 ①
13	DefGW Part 4	1	0–255	Default Gateway Part 4 ①
14	InputAssembly	71	0–255	See Assembly Instances Implemented by the OPTCQ Option Board on **Page 20**
15	OutputAssembly	21	0–255	

Note

① These values are selected by DHCP from the network server.

DHCP

The OPTCQ EtherNet/IP communication option card supports DHCP for easier network configuration. Dynamic Host Configuration Protocol (DHCP) is a network protocol that is used to configure network devices so that they can communicate on an IP network. As a DHCP client, the OPTCQ option card negotiates with the DHCP server to determine its IP address and obtain any other initial configuration details it needs for network operation.

IP Address

IP is divided into four parts. (Part = Octet) Default IP Address is 192.168.0.10.

Communication Timeout

Defines how much time can pass from the last received message from the client device before fieldbus fault is generated. Communication timeout is disabled when given the value 0. Communication timeout value can be changed from the keypad or with NCIPConfig tool (read chapter IP Tool NCIPConfig).

Note: If the network cable is broken from OPTCQ EtherNet/IP board end, a fieldbus error is generated immediately.

All EtherNet/IP parameters are saved to the OPTCQ EtherNet/IP board (not to the control board). If a new OPTCQ EtherNet/IP board is installed in the control module, you must configure the new OPTCQ EtherNet/IP board. OPTCQ board parameters can be saved to the keypad, with NCIPConfig tool or with 9000X drive.

Static IP Address

In most cases the user may want to establish a Static IP Address for the OPTCQ EtherNet/IP option card based on their network configuration. The user can manually define the network address for the OPTCQ as long as all units connected to the network are given the same network portion of the address. This is often the case when replacing the previous generation OPTCK EtherNet/IP option card.

In these situations the user will need to manually set the IP Address in the OPTCQ option card by using the 9000X drive keypad or the PC Software tool "9000XDRIVE.EXE" (available on the Eaton website). Be aware that overlapping IP addresses can cause conflicts between devices on the network. For more information about selecting IP addresses, contact your network administrator.

Manual IP Address Configuration

Using the 9000X Drive Keypad

Using the 9000X Drive Keypad to set the IP Address manually in the OPTCQ EtherNet/IP option card:

1. Make note of the desired IP Address (or the address of the existing card) for future use.
2. Power off the drive control and wait three minutes for the internal voltages to dissipate.
3. Install the new OPTCQ EtherNet/IP option card in slot D or E of the 9000X drive

   or

   Replace the OPTCK EtherNet/IP card with the new OPTCQ EtherNet/IP card
4. Power on the drive control. The drive will report a "Device Change". Press "Reset" to clear the fault.
5. Using the keypad, set the IP address in the OPTCQ card to the desired address setting or to match the old card address by:
   a. Hold the "Enter" key until the display shows "Parameters"
   b. Press the up arrow until the display shows "Expander Boards"
   c. Press the right arrow, then the up arrow until the display shows "OPTCQ"
   d. Press the right arrow to access the parameters, then the right arrow to view the parameters
   e. Step through each parameter to verify or make any changes needed. Press "Enter" after adjustment

   The IP Address settings can be found at the Expander Board Menu (M5 or M6, slot D or E respectively).

   The IP Address can be set in the menu level (G6.x.1.2) through (G6.x.1.15).
6. Check that the Output Assembly Instance is set correctly, typically "101".
7. Check that the Input Assembly Instance is set correctly, typically "127".

EtherNet/IP

Overview

EtherNet/IP (Ethernet/Industrial Protocol) is a communication system suitable for use in industrial environments. EtherNet/IP allows industrial devices to exchange time-critical application information. These devices include simple I/O devices such as sensors/actuators, as well as complex control devices such as robots, programmable logic controllers, welders, and process controllers. EtherNet/IP uses CIP (Control and Information Protocol), the common network, transport, and application layers also shared by ControlNet and EtherNet/IP. EtherNet/IP then makes use of standard Ethernet and TCP/IP technology to transport CIP communications packets. The result is a common, open application layer on top of open and highly popular Ethernet and TCP/IP protocols.

EtherNet/IP messaging forms:

- Unconnected messaging is used for connection establishment and for infrequent, low-priority messages
- Connected messaging uses resources that are dedicated in advance to a particular purpose such as real-time I/O data transfer

EtherNet/IP messaging connections:

- Explicit messaging connections are general purpose point-to-point connections. Messages are sent through TCP protocol
- Implicit (I/O data) connections are established to move application-specific I/O data at regular intervals. They are often set up as one-to-many relationships in order to take full advantage of the producer-consumer multicast model. Implicit messages are sent through UDP protocol

AC/DC Drive Profile

In order to provide interoperability between devices from different manufacturers, there must be a defined “standard” in which those devices:

- Exhibit the same behavior
- Produce and/or consume the same basic set of I/O data
- Contain the same basic set of configurable attributes

The formal definition of this information is known as a device profile.

EDS File

EDS—Is the abbreviation for Electronic Data Sheet, a file on disk that contains configuration data for specific device types.

You can provide configuration support for your device by using a specially formatted ASCII file, referred to as the EDS. An EDS provides information about the device configuration data’s:

- Context
- Content
- Format

The information in an EDS allows configuration tools to provide informative screens that guide a user through the steps necessary to configure a device. An EDS provides all of the information necessary to access and alter the configurable parameters of a device. This information matches the information provided by instances of the parameter object class. The CIP object library describes the parameter object class in detail.

Explicit Messaging

Explicit Messaging is used in commissioning and parameterizing of the EtherNet/IP board. Explicit messages provide multipurpose, point-to-point communication paths between two devices. They provide the typical request/ response-oriented network communication used to perform node configuration and problem diagnosis. Explicit messages typically use low priority identifiers and contain the specific meaning of the message right in the data field. This includes the service to be performed and the specific object attribute address.

Note: If Class 1 connection (cyclic data) has been established, then explicit messages cannot be used to control output data. However, this restriction doesn’t apply for IO Data reading.

List of Object Classes

The communication interface supports the following object classes.

Object Classes

Class	Object
0x01	Identity objects
0x04	Assembly object
0x06	Connection manager object
0x28	Motor data object
0x29	Control supervisor object
0x2A	AC/DC drive object
0xA0	Vendor parameters object
0xBE	Assembly instance selector object
0xF5	TCP/IP interface object
0xF6	Ethernet link object

List of Services

The services supported by these object classes are shown below.

Services Supported by Object Classes

Service Code (in hex)	Service Name	Identity Object		Connection Manager		TCP/IP Interface		Ethernet Link		Assembly		Motor Data		Control Supervisor		AC/DC Drive		Vendor Parameter		Assembly Instance Selector	
		Class	Inst	Class	Inst	Class	Inst	Class	Inst	Class	Inst	Class	Inst	Class	Inst	Class	Inst	Class	Inst	Class	Inst
01	Get_Attributes_All		Y	Y	Y	Y	Y	Y													
05	Reset (Type 0)		Y												Y						
0E	Get_Attribute_Single	Y	Y	Y	Y	Y	Y	Y	Y		Y		Y		Y		Y		Y		Y
10	Set_Attribute_Single						Y				Y		Y		Y		Y		Y		Y
4E	Forward Close				Y																
52	Unconnected_Send				Y																
54	Forward_Open				Y																

Note: See Common Industrial Objects Implemented by the OPTCQ Option Board on **Page 12**.

List of Data Types

The attribute list that follows includes information on the data type of each attribute. The following tables explain the data, structure, and array type codes used in the data type column.

Elementary Data Types

Data Type Name	Data Type Code (in hex)	Data Type Description
BOOL	C1	Logical boolean with values true and false
SINT	C2	Signed 8-bit integer value
INT	C3	Signed 16-bit integer value
USINT	C6	Unsigned 8-bit integer value
UINT	C7	Unsigned 16-bit integer value
UDINT	C8	Unsigned 32-bit integer value
BYTE	D1	Bit string—8-bits
WORD	D2	Bit string—16-bits
SHORT_STRING	DA	Character sting (1 byte per character, 1 byte length indicator)

Constructed Data Types

Type Code	Description
A1	Abbreviated array type encoding
A2	Formal structure type encoding

Reset Service

The following table lists the different types of resets supported by the identity object.

Resetting the OPTCQ interface to its out-of-box configuration will set all attributes to their default values and change the response of the drive to a loss of communications with the OPTCQ. The device will have to be re-configured for your application before resuming normal operation.

Reset

Value	Reset Type
0	Emulate as closely as possible the cycling of power to the OPTCQ EtherNet/IP Interface. This value is the default if this parameter is omitted. The 9000X drive shall be stopped if it is running.

Common Industrial Objects Implemented by the OPTCQ Option Board

CIP Common Required Objects

Identity Object, Class 0x01

Identity	Description	Data Type	Access Rule
Class Attributes			
01h	Revision		Get
02h	Maximum instances	UINT	Get
Class Services			
0Eh	Get_Attribute_Single		
Instance Attributes			
01h	Vendor ID	UINT	Get
02h	Device type	UINT	Get
03h	Product code	UINT	Get
04h	Revision	STRUCT of:	Get
	Major revision	USINT	
	Minor revision	USINT	
05h	Status	WORD	Get
06h	Serial number	UDINT	Get
07h	Product name	SHORT_STRING	Get
Instance Services			
01h	Get_Attributes_All		
05h	Reset ①		
0Eh	Get_Attribute_Single		

Note

① Only reset type 0—reset of the option board.

Connection Manager Object, Class 0x06

Identity	Description	Data Type	Access Rule
Class Attributes			
01h	Revision		Get
02h	Maximum instance		Get
Class Services			
01h	Get_Attributes_All		
0Eh	Get_Attribute_Single		
Instance Attributes			
01h	Open requests	UINT	Get
02h	Open format rejects	UINT	Get
03h	Open resource rejects	UINT	Get
04h	Open other rejects	UINT	Get
05h	Close requests	UINT	Get
06h	Close format requests	UINT	Get
07h	Close other requests	UINT	Get
08h	Connection timeouts	UINT	Get
Instance Services			
01h	Get_Attributes_All		
0Eh	Get_Attribute_Single		
4Eh	Forward close		
52h	Unconnected_Send		
54h	Forward_Open		

TCP/IP Interface Object, Class 0xF5

Identity	Description	Data Type	Access Rule
Class Attributes			
01h	Revision		Get
02h	Maximum Instance	UINT	Get
Class Services			
01h	Get_Attributes_All		
0Eh	Get_Attribute_Single		
Instance Attributes			
01h	Status	DWORD	Get
02h	Configuration capability	DWORD	Get
03h	Configuration control	DWORD	Get/set
04h	Physical link	STRUCT of:	Get
	Path size	UINT	
	Path	Padded EPATH	
05h	Interface configuration	STRUCT of:	Get/set
	IP address	UDINT	
	Network mask	UDINT	
	Gateway address	UDINT	
	Name server	UDINT	
	Name server 2	UDINT	
	Domain name	STRING	
06h	Host name	STRING	Get/set
Instance Services			
01h	Get_Attributes_All		
0Eh	Get_Attribute_Single		
10h	Set_Attribute_Single		

Note: Attribute configuration control supports only value 0 (device is using configuration values that are stored in non-volatile memory). Attribute host name is used just for information purposes.

Ethernet Link Object, Class 0xF6

Identity	Description	Data Type	Access Rule
Class Attributes			
01h	Revision	UINT	Get
02h	Maximum instance	UINT	Get
03h	Number of instances	UINT	Get
Class Services			
01h	Get_Attributes_All		
0Eh	Get_Attribute_Single		
Instance Attributes			
01h	Interface speed	UDINT	Get
02h	Interface flags	DWORD	Get
03h	Physical address	ARRAY of 6 USINTs	Get
Instance Services			
0Eh	Get_Attribute_Single		

Objects Present in an AC/DC Drive

Assembly Object, Class 0x04

Identity	Description	Data Type	Access Rule
Class Attributes			
Not supported			
Class Services			
Not supported			
Instance attributes			
03h	Data	ARRAY of BYTE	Get/set
Instance Services			
0Eh	Get_Attribute_Single		
10h	Set_Attribute_Single		

Note: If Class 1 connection (cyclic data) has been established, then explicit messages cannot be used to control output data. However, this restriction doesn't apply for IO data reading.

Motor Data Object, Class 0x28

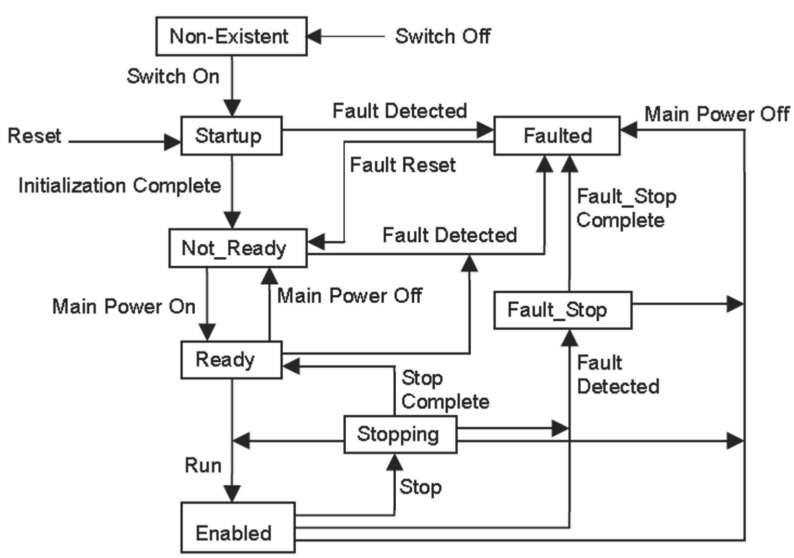
Identity	Description	Data Type	Access Rule
Class Attributes			
Not supported			
Class Services			
Not supported			
Instance Attributes			
03h	Motor type	USINT	Get
06h	Rated current	UINT	Get/set
07h	Rated voltage	UINT	Get/set
09h	Rated frequency	UINT	Get/set
0Ch	Pole count	UINT	Get
0Fh	Base speed	UINT	Get/set
Instance Services			
0Eh	Get_Attribute_Single		
10h	Set_Attribute_Single		

Control Supervisor Object, Class 0x29

Identity	Description	Data Type	Access Rule
Class Attributes			
Not supported			
Class Services			
Not supported			
Instance Attributes			
03h	Run1	BOOL	Get/set
04h	Run2	BOOL	Get/set
05h	NetCtrl	BOOL	Get/set
06h	State	USINT	Get
07h	Running1	BOOL	Get
08h	Running2	BOOL	Get
09h	Ready	BOOL	Get
0Ah	Faulted	BOOL	Get
0Bh	Warning	BOOL	Get
0Ch	FaultRst	BOOL	Get/set
0Fh	CtrlFromNet	BOOL	Get
Instance Services			
0Eh	Get_Attribute_Single		
10h	Set_Attribute_Single		
05h	Reset		

Note: When both Run (Run1 and Run2) attributes set, then no action.

Switch Diagram



AC/DC Drive Object, Class 0x2A

Identity	Description	Data Type	Access Rule
Class Attributes			
Not supported			
Class Services			
Not supported			
Instance Attributes			
03h	AtReference	BOOL	Get
04h	NetRef	BOOL	Get/set
05h	NetProc	BOOL	Get/set
06h	DriveMode	USINT	Get/set
07h	SpeedActual	INT	Get
08h	SpeedRef	INT	Get/set
0Bh	TorqueActual	INT	Get
0Ch	TorqueRef	INT	Get/set
0Dh	ProcessActual	INT	Get
0Eh	ProcessRef	INT	Get/set
1Dh	RefFromNet	BOOL	Get
Instance Services			
0Eh	Get_Attribute_Single		
10h	Set_Attribute_Single		

Vendor Specific Objects

Vendor Parameters Object, Class 0xA0

Vendor parameter object is used in order to get access to drive parameters. Because drive parameters are identified by the 16-bit length ID number, it is impossible to use only attribute ID, which is 8-bit in length. To overcome this issue, we are using the following method to calculate requested drive parameter ID:

Drive parameter ID = instance ID (higher byte) + attribute ID (lower byte)

Vendor Parameters Object

Identity	Description	Access Rule
Class Attributes		
Not supported		
Class Services		
Not supported		
Instance Attributes		
Lower byte of the parameter ID		
Instance Services		
0Eh	Get_Attribute_Single	
10h	Set_Attribute_Single	

Assembly Instance Selector Object, Class 0xBE

Identity	Description	Data Type	Access Rule
Class Attributes			
Not supported			
Class Services			
Not supported			
Instance Attributes			
03h	OutputInstance	USINT	Get/set
04h	InputInstance	USINT	Get/set
Instance Services			
0Eh	Get_Attribute_Single		
10h	Set_Attribute_Single		

Assembly Instances Implemented by the OPTCQ Option Board

Output Instances

Assemblies 20–25 ODVA AC/DC profile; assemblies 71–75 ODVA AC/DC profile; assemblies >100 → Eaton profile.

Assembly Instance 20

Instance 20 (Output)/Length = 4 Bytes

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	—	—	—	—	—	FaultReset	—	RunFwd
1	—	—	—	—	—	—	—	—
2	Speed reference (low byte), RPM							
3	Speed reference (high byte), RPM							

Assembly Instance 21 (Default)

Instance 21 (Output)/Length = 4 Bytes

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	—	NetRef	NetCtrl	—	—	FaultReset	RunRev	RunFwd
1	—	—	—	—	—	—	—	—
2	Speed reference (low byte), RPM							
3	Speed reference (high byte), RPM							

Assembly Instance 23

Instance 23 (Output)/Length = 6 Bytes

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	—	NetRef	NetCtrl	—	—	FaultReset	RunRev	RunFwd
1	—	—	—	—	—	—	—	—
2	Speed reference (low byte), RPM							
3	Speed reference (high byte), RPM							
4	Torque reference (low byte), Nm							
5	Torque reference (high byte), Nm							

Note: Torque reference is not sent to the drive if Motor Control Mode (parameter ID 600) is set to values other than:

- 2—torque control
- 4—closed loop torque control

Torque reference is sent to the drive as a Process Data 1.

Note: Torque reference is not functional in NXL.

Assembly Instance 25

Instance 25 (Output)/Length = 6 Bytes

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	NetProc	NetRef	NetCtrl	—	—	FaultReset	RunRev	RunFwd
1	Drive mode							
2	Speed reference (low byte), RPM							
3	Speed reference (high byte), RPM							
4	Process reference (low byte)							
5	Process reference (high byte)							

We are supporting the following drive modes:

- 0 (Vendor specific)—process reference is sent to the drive as Process Data 1
- 4 (Process control)—process reference is sent to the drive as Process Data 2 (see **Page 27**)

Other drive modes are not supported. If they are used, then Process Reference is not handled.

Assembly Instance 101

Instance 101 (Output)/Length = 8 Bytes

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	—	NetRef	NetCtrl	—	—	FaultReset	RunRev	RunFwd
1	—							
2	FBSpeed reference (low byte), RPM							
3	FBSpeed reference (high byte), RPM							
4	FBProcessDataIn1(low byte)							
5	FBProcessDataIn1(high byte)							
6	FBProcessDataIn2(low byte)							
7	FBProcessDataIn2(high byte)							

Process data is sent to the drive independently from the NetRef and NetCtrl bits settings.

Assembly Instance 111

Instance 111 (Output)/Length = 20 Bytes

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	FBFixedControl Word (low byte)							
1	FBFixedControl Word (high byte)							
2	FBSpeedReference (low byte) ①							
3	FBSpeedReference (high byte) ①							
4	ProcessDataIn1 (low byte)							
5	ProcessDataIn1 (high byte)							
6	ProcessDataIn2 (low byte)							
7	ProcessDataIn2 (high byte)							
8	ProcessDataIn3 (low byte)							
9	ProcessDataIn3 (high byte)							
10	ProcessDataIn4 (low byte)							
11	ProcessDataIn4 (high byte)							
12	ProcessDataIn5 (low byte)							
13	ProcessDataIn5 (high byte)							
14	ProcessDataIn6 (low byte)							
15	ProcessDataIn6 (high byte)							
16	ProcessDataIn7 (low byte)							
17	ProcessDataIn7 (high byte)							
18	ProcessDataIn8 (low byte)							
19	ProcessDataIn8 (high byte)							

Note

① This is the reference 1 to the frequency converter and is used normally as speed reference. The allowed scaling is 0–10,000. In the application, the value is scaled in percentage of the frequency area between set minimum and maximum frequency. (0 = 0.00% —10,000 = 100.00%).

Control Word

Bit	Description Value = 0	Value = 1
0	STOP	RUN
1	Clockwise	Counterclockwise
2	Rising edge of this bit will reset active fault	Rising edge of this bit will reset active fault
3–15	Not in use	Not in use

Input Instances

Assembly Instance 70

Instance 70 (Input)/Length = 4 Bytes

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	—	—	—	—	—	Running1	—	Faulted
1	—							
2	Speed actual (low byte), RPM							
3	Speed actual (high byte), RPM							

Assembly Instance 71 (Default)

Instance 71 (Input)/Length = 4 Bytes

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	AtReference	RefFromNet	CtrlFromNet	Ready	Running2	Running1	Warning	Faulted
1	Drive state, see **Page 24**							
2	Speed actual (low byte), RPM							
3	Speed actual (high byte), RPM							

Assembly Instance 73

Instance 73 (Input)/Length = 6 Bytes

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	AtReference	RefFromNet	CtrlFromNet	Ready	Running2	Running1	Warning	Faulted
1	Drive state, see **Page 24**							
2	Speed actual (low byte), RPM							
3	Speed actual (high byte), RPM							
4	Torque actual (low byte), Nm							
5	Torque actual (high byte), Nm							

Note: Torque reference is not functional in NXL.

Assembly Instance 75

Instance 75 (Input)/Length = 6 Bytes

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	AtReference	RefFromNet	CtrlFromNet	Ready	Running2	Running1	Warning	Faulted
1	Drive state, see **Page 24**							
2	Speed actual (low byte), rpm							
3	Speed actual (high byte), rpm							
4	Process actual (low byte)							
5	Process actual (high byte)							

Assembly Instance 107

Instance 107 (Input)/Length = 8 Bytes

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	AtReference	RefFromNet	CtrlFromNet	Ready	Running2	Running1	Warning	Faulted
1	Drive state, see below							
2	% speed actual (low byte) ①							
3	% speed actual (high byte) ①							
4	Process DataOut1 (low byte)							
5	Process DataOut1 (high byte)							
6	Process DataOut2 (low byte)							
7	Process DataOut2 (high byte)							

Note

① This is the actual value from the frequency converter. The value is between 0 and 10,000. In the application, the value is scaled in percentage of frequency area between set minimum and maximum frequency. (0 = 0.00%—10,000 = 100.00%).

Drive State

	Reference
0x00	DN_NON_EXISTANT
0x01	DN_STARTUP
0x02	DN_NOT_READY
0x03	DN_READY
0x04	DN_ENABLED
0x05	DN_STOPPING
0x06	DN_FAULT_STOP
0x07	DN_FAULTED

Assembly Instance 117

Instance 117 (Input): EIP Drive Status/Length = 34 Bytes

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	FBStatusWord (low byte)							
1	FBStatusWord (high byte)							
2	% speed actual (low byte) ①							
3	% speed actual (high byte) ①							
4	RPM speed actual (low byte) ②							
5	RPM speed actual (high byte) ②							
6	RPM with slip speed actual (low byte) ③							
7	RPM with slip speed actual (high byte) ③							
8	Reserved							
9	Reserved							
10	Reserved							
11	Reserved							
12	Reserved							
13	Reserved							
14	Reserved							
15	Reserved							
16	Reserved							
17	Reserved							
18	ProcessDataOut1 (low byte)							
19	ProcessDataOut1 (high byte)							
20	ProcessDataOut2 (low byte)							
21	ProcessDataOut2 (high byte)							
22	ProcessDataOut3 (low byte)							
23	ProcessDataOut3 (high byte)							
24	ProcessDataOut4 (low byte)							
25	ProcessDataOut4 (high byte)							
26	ProcessDataOut5 (low byte)							
27	ProcessDataOut5 (high byte)							
28	ProcessDataOut6 (low byte)							
29	ProcessDataOut6 (high byte)							
30	ProcessDataOut7 (low byte)							
31	ProcessDataOut7 (high byte)							
32	ProcessDataOut8 (low byte)							
33	ProcessDataOut8 (high byte)							

Notes

① This is the actual value from the frequency converter. The value is between 0 and 10,000. In the application, the value is scaled in percentage of frequency area between set minimum and maximum frequency. (0 = 0.00%—10,000 = 100.00%).

② The RPM speed actual is the actual speed of the motor. The unit is RPM.

③ The RPM with slip speed actual is the actual speed of the motor with slip speed. The unit is RPM.

Status Word

	Description	
Bit	**Value = 0**	**Value = 1**
0	Not ready	Ready
1	Stop	Run
2	Clockwise	Counterclockwise
3	No fault	Faulted
4	No alarm	Alarm
5	Reference frequency not reached	Reference frequency reached
6	Motor not running at zero speed	Motor running at zero speed
7	Flux ready	Flux not ready
8–15	Not in use	Not in use

Assembly Instance 127

Instance 127: Length = 20 Bytes

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Status Word (low byte)							
1	Status Word (high byte)							
2	Speed Actual (low byte) in % of maximum speed							
3	Speed Actual (high byte) in % of maximum speed							
4	Process Data Out 1 (low byte)							
5	Process Data Out 1 (high byte)							
6	Process Data Out 2 (low byte)							
7	Process Data Out 2 (high byte)							
8	Process Data Out 3 (low byte)							
9	Process Data Out 3 (high byte)							
10	Process Data Out 4 (low byte)							
11	Process Data Out 4 (high byte)							
12	Process Data Out 5 (low byte)							
13	Process Data Out 5 (high byte)							
14	Process Data Out 6 (low byte)							
15	Process Data Out 6 (high byte)							
16	Process Data Out 7 (low byte)							
17	Process Data Out 7 (high byte)							
18	Process Data Out 8 (low byte)							
19	Process Data Out 8 (high byte)							

Assemblies 117 and 127 Semantics

Assembly 127 is a shortened version of assembly 117. Speed Actual and Process Data Out 1–8 are the same for both assemblies. But the Data Select Sync Word is only present in assembly 117, and the status word is defined differently for assemblies 117 and 127.

If FB Status Type of the Selectors object (0 x BE) is 0, for assembly 117 the status word is defined the same as bytes 0 and 1 of assembly 75; and for assembly 127, it is the fixed status word. See **Page 25** and table above.

Appendix A—Process Data Variables For All-In-One Application

This appendix lists how process data variables are defined for the all-in-one application. Other applications may define the process data variables differently.

Process Data Out (Slave to Master)

The fieldbus master can read the frequency converter's actual values using process data variables. All software applications use process data as follows:

Process Data Out Variables

ID	Data	Value	Unit	Scale
2104	Process data OUT 1	Output frequency	Hz	0.01 Hz
2105	Process data OUT 2	Motor speed	rpm	1 rpm
2106	Process data OUT 3	Motor current	A	0.1A
2107	Process data OUT 4	Motor torque	%	0.1%
2108	Process data OUT 5	Motor power	%	0.1%
2109	Process data OUT 6	Motor voltage	V	0.1V
2110	Process data OUT 7	DC link voltage	V	1V
2111	Process data OUT 8	Active fault code	—	—

The multipurpose control application has a selector parameter for every process data. The monitoring values and drive parameters can be selected using the ID number. Default selections are as in the table above.

Process Data In (Master to Slave)

ControlWord, reference and process data are used with all-in-one applications as follows.

Basic, Standard, Local/Remote Control and Multistep Speed Control Applications

ID	Data	Value	Unit	Scale
2003	Reference	Speed reference	%	0.01%
2001	ControlWord	Start/stop command fault reset command	—	—
2004–2011	PD1–PD8	Not used	—	—

Multipurpose Control Application

ID	Data	Value	Unit	Scale
2003	Reference	Speed reference	%	0.01%
2001	ControlWord	Start/stop command fault reset command	—	—
2004	Process Data In 1	Torque reference	%	0.1%
2005	Process Data In 2	Free analogia INPUT	%	0.01%
2006–2011	PD3–PD8	Not used	—	—

PID Control and Pump and Fan Control Applications

ID	Data	Value	Unit	Scale
2003	Reference	Speed reference	%	0.01%
2001	ControlWord	Start/stop command fault reset command	—	—
2004	Process Data In 1	Reference for PID controller	%	0.01%
2005	Process Data In 2	Actual value 1 to PID controller	%	0.01%
2006	Process Data In 3	Actual value 2 to PID controller	%	0.01%
2007–2011	PD4–PD8	Not used	—	—

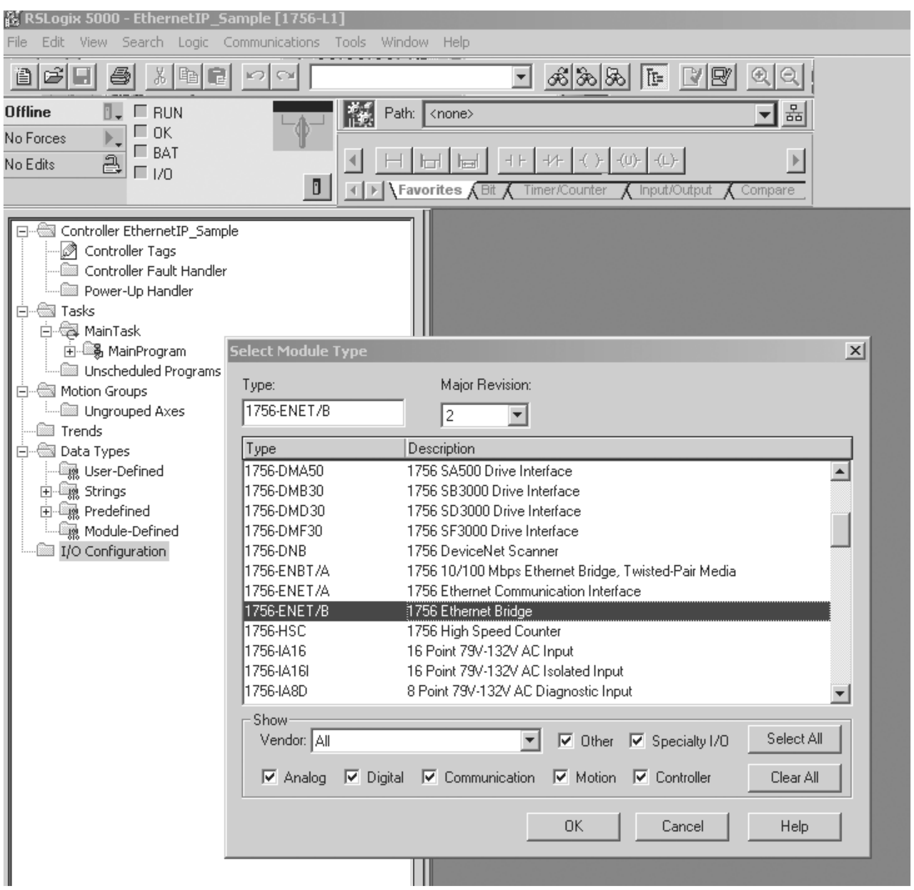
PLC Programming

ControlLogix 5000

When using a ControlLogix PLC as an OPTCQ master, you must first configure a compatible EtherNet/IP scanner, and then map ladder logic variables to the scanner. The following example is for a ControlLogix5550 with an ENET/B Ethernet bridge module. The ENET/B supports polled messaging. Some PLCs do not support polled messaging for EtherNet/IP. For example, the SLC500 only supports explicit messaging.

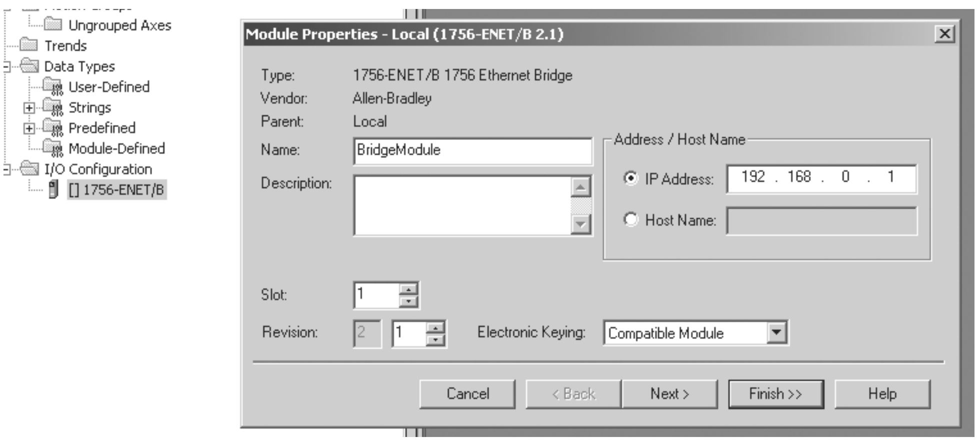
Right-click on I/O configuration and select "New Module." Select the 1756-ENET/B Ethernet Bridge (see figure below).

1756-ENET/B Ethernet Bridge



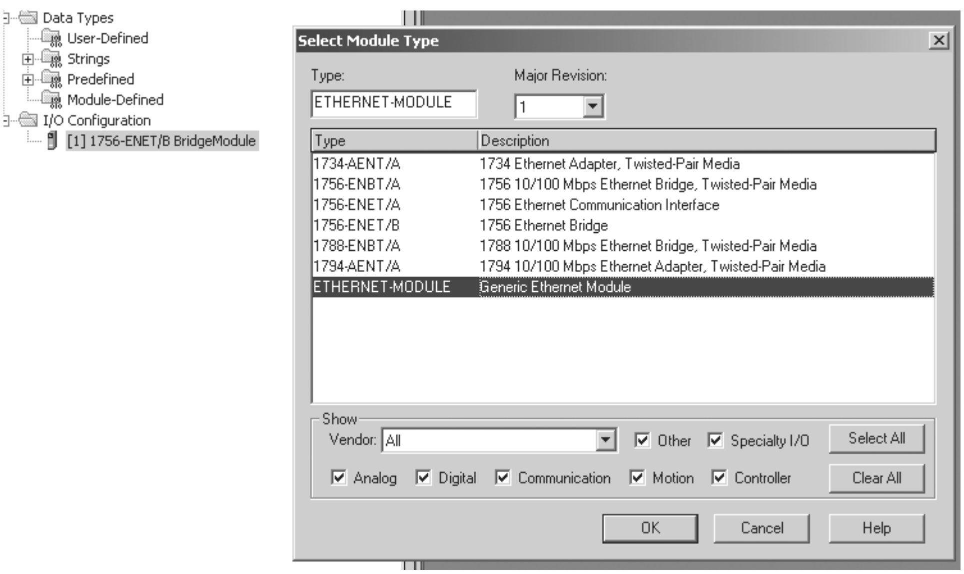
After the bridge module is added, a dialog box will appear requesting the configuration of the bridge module parameters. Enter a name and the IP address used by the bridge module on the first tab (see figure below). Select next and enter a polling interval for the bridge. A polling interval of 200 ms to 1000 ms is recommended.

Module Properties



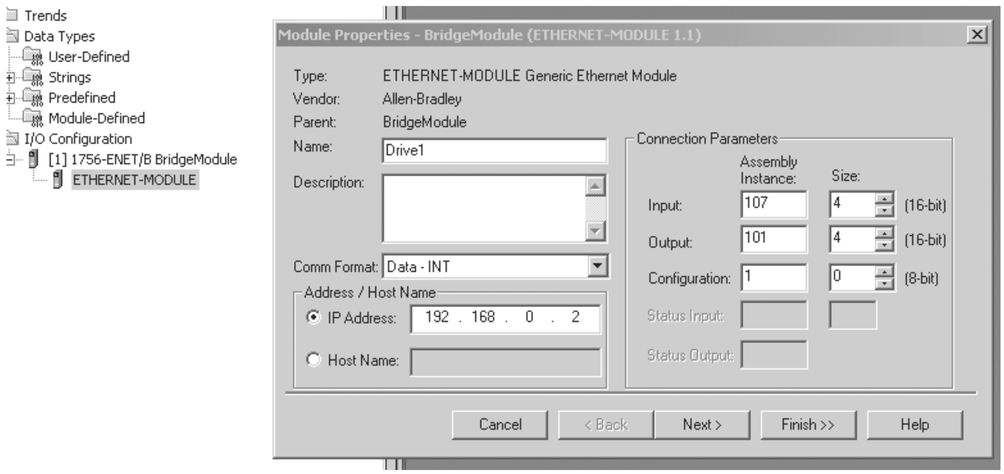
The next step is to add a drive to the bridge module. Right click on the bridge module, and add a new Generic Ethernet Module (see figure below). Fill in the drive specific information. Be sure to select comm. Format INT. Do this before entering the connection parameters. In this example, the input and output assemblies match the default assembly numbers used by the OPTCQ. Use a configuration assembly value of 1 with a length of zero (see **Page 30**).

Select Module Type



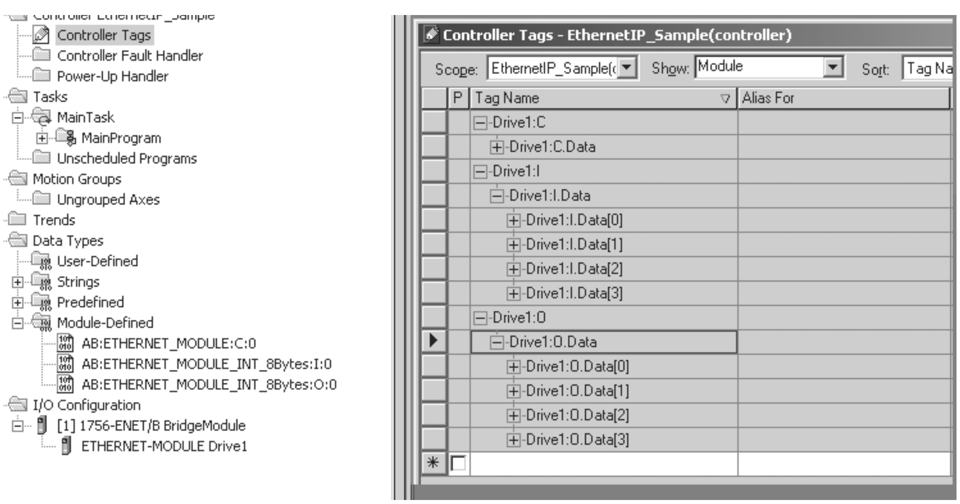
Add additional drive modules as needed, remembering to assign unique names and IP addresses to each module. Variable tags may then be viewed from the controller tags item in the property tree.

Module Properties—Bridge Module

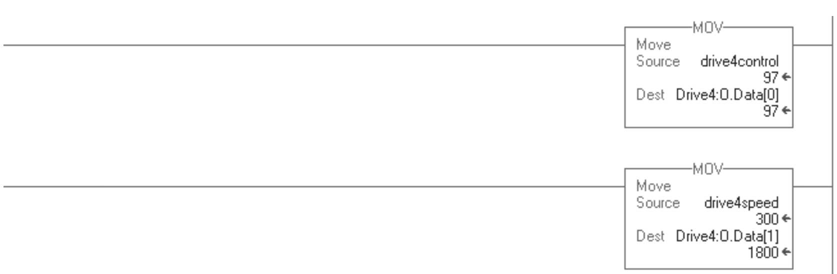


Tags from each drive may now be accessed using standard ladder instructions. For example, in the bottom figure, move instructions are used to move the speed and start commands for drive4. Notice that the use of INT data types in the scan list allow for simplified tag access. For example, the speed reference can be changed without having to use math operators to adjust the upper and lower bytes.

Controller Tags—EtherNet/IP_Sample (Controller)



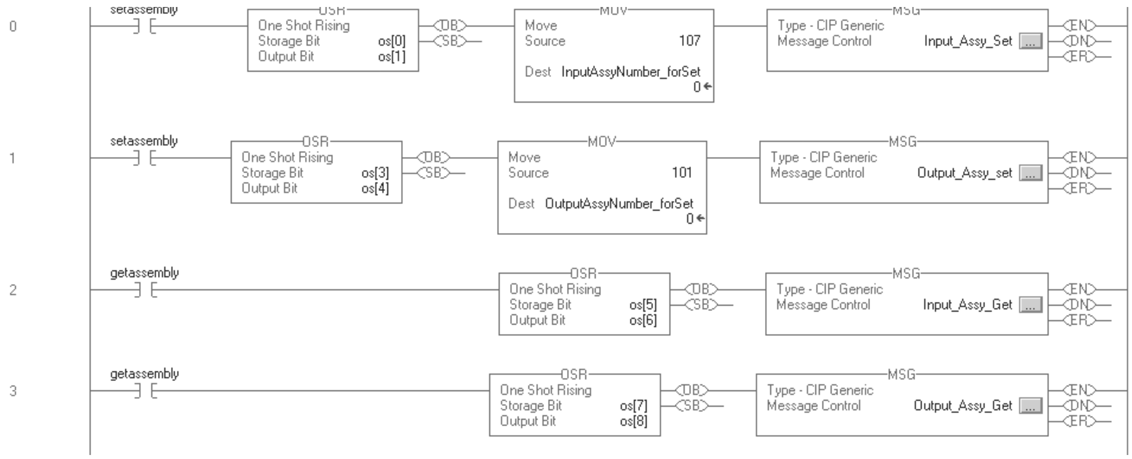
Move Instructions



Explicit Messages

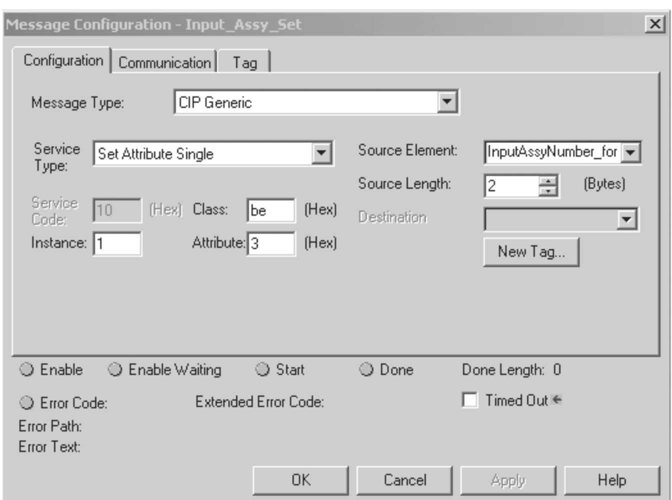
The ladder logic in the figure below creates and sends an explicit message that changes the input and output assembly instance numbers used by the drive. It does this by using a message block, configured to send a Set Attribute Single CIP message. The configuration of the drive's input and output assemblies is done by changing attributes 3, and 4 of the selector class (0BE hex), instance 1. These items are used in the class, instance, and attribute argument fields of the configuration dialog in the figure at bottom.

Ladder Logic Message Blocks in RSLogix5000



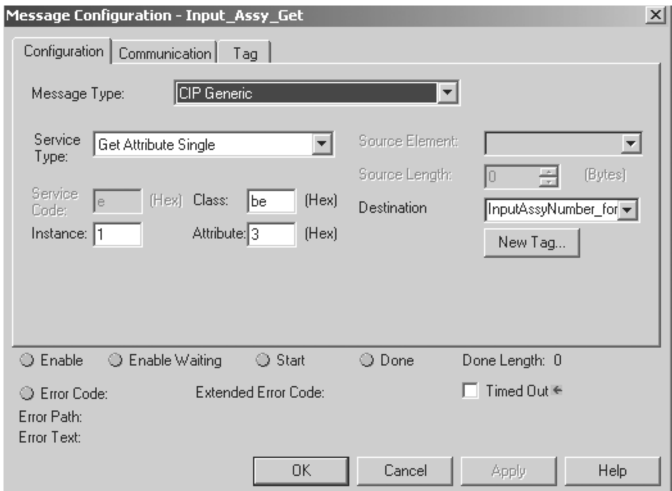
Closing the SetAssembly contact fires a one shot, which in turn sets the variable InputAssyNumberForSet to a value of 107. This variable is used as the source element in the message configuration dialog (see figure below). You must also set the device path on the communication tab to the name of the drive you wish to send the message to, in this case Drive1. This device path determines which drive receives the explicit message.

Message Configuration for RSLogix5000



Forcing the **GetAssembly** contact fires a one-shot that triggers another message block that sends a **Get Attribute Single** message. The result of the get attribute single message is then placed in the destination element, **InputAssyNumberForGet**. This message response verifies that the drive has correctly received and responded to the previous setAttributeSingle message.

Message Configuration

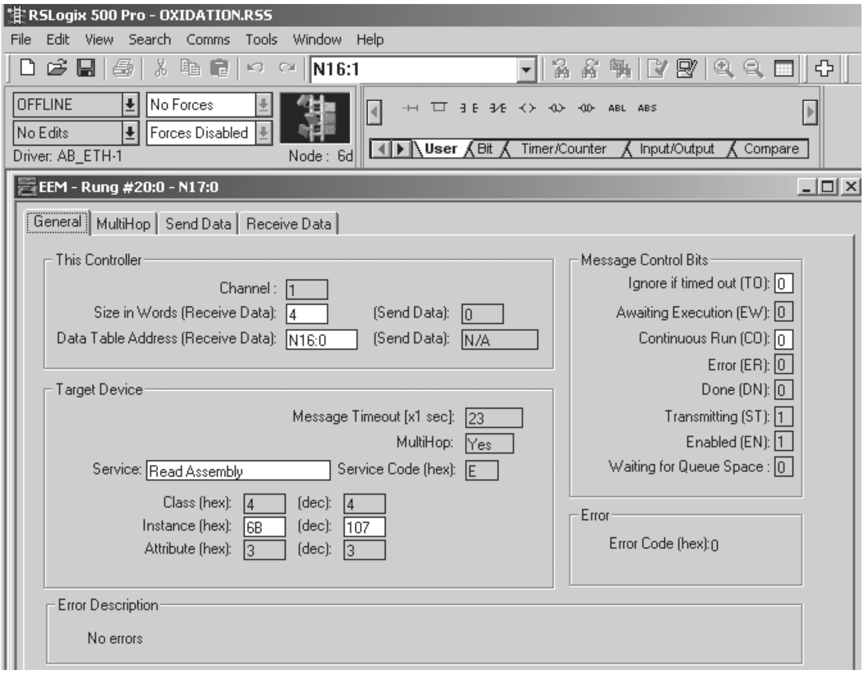


It's important to remember that explicit messages use PLC processor cycles that are best used to scan ladder logic. In the sample logic of figure A, the explicit message that sets the I/O assembly numbers is required to run only one time. Once the drive is configured to use a specific I/O assembly, it retains that information and the logic no longer has to run. This is the reason that a one-shot function block is used; it ensures that only one message is sent to the drive, and then will not execute again until the setAssembly contact opens a closes again.

Using Explicit Messages with I/O Assemblies

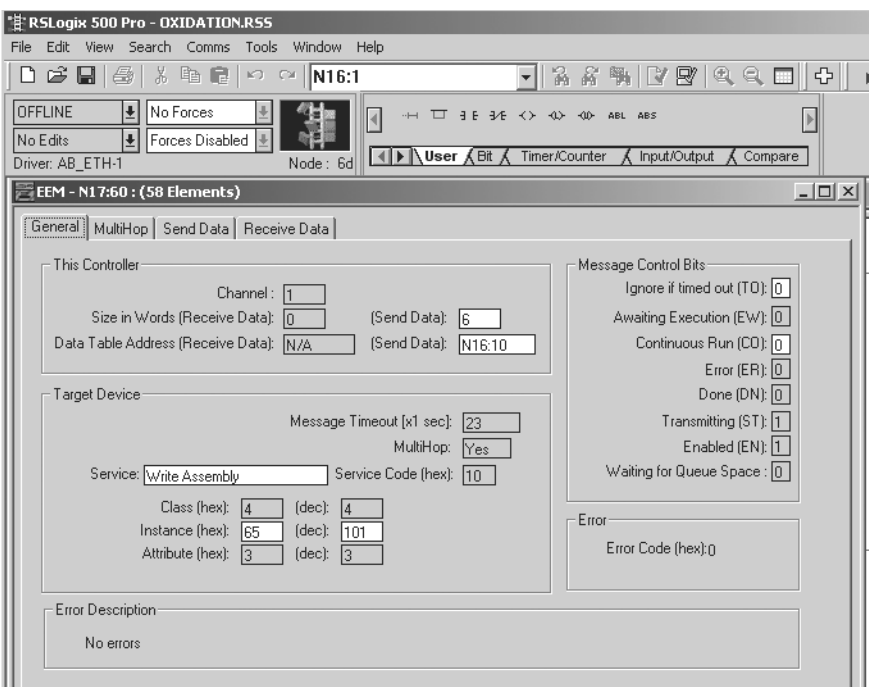
Some PLCs, such as the Rockwell SLC500, do not allow for polled messaging over an EtherNet/IP. It is possible to transfer data using an I/O assembly as a template, but an explicit message must be used in place of the usual polled (implicit) message. The CIP specification provides for explicit access to the I/O assemblies via the "assembly object" class. The use of a "get attribute single" or "set attribute single" service to class 4, instance N, where N is the assembly number, attribute 3 (assembly data) is used. The same ladder logic structure used in figure A may be used, but a mechanism must be employed to periodically trigger the explicit messages. A timer may be used for this purpose. The timer should be set to a reasonable interval for reading information (~100 ms). The set service need only be called when control, speed change, or some other parameter write to the drive is required. A timer is still recommended to throttle messages, as event driven changes (such as a very slight speed change) may result in calling the message block logic too frequently. Excessive calls to message blocks can result in poor ladder logic performance.

RSLogix500 Configuration of Get Attribute Single



Example configuration dialogs for getting and setting RSLogix 500 message blocks are shown in the figures on this page. The figure above shows configuration of the read assembly message block, which is used to get input information from the drives assembly number 107. The figure below shows the equivalent write assembly message block.

RSLogix500 Configuration of Set Attribute Single



Appendix B—IP Tool NCIPConfig

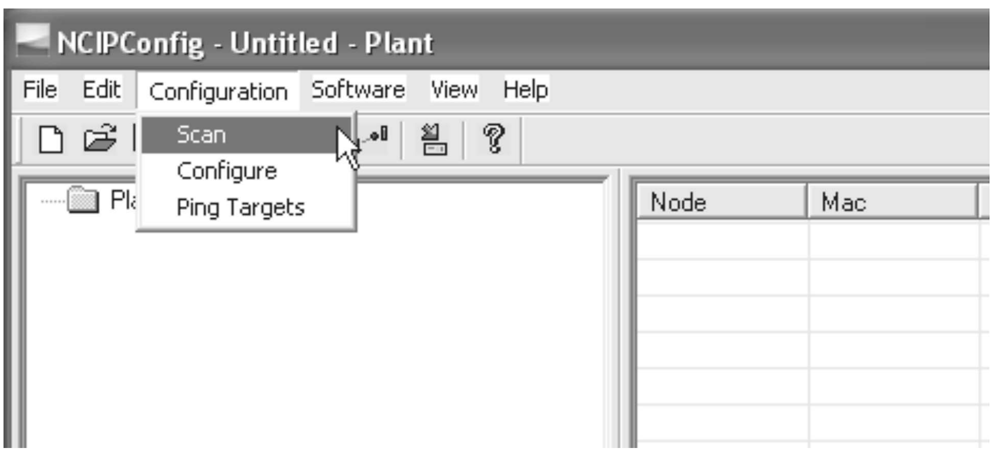
You need a PC with an Ethernet connection and the NCIPConfig tool installed to set the EtherNet/IP board's IP addresses. To install the NCIPConfig tool, start the installation program from the CD or download it from the www.eaton.com Web site. After starting the installation program, follow the on-screen instructions.

Once the program is installed successfully, you can launch it by selecting it in the Windows Start menu. Follow these instructions to set the IP addresses. Select Help --> Manual if you want more information about the software features.

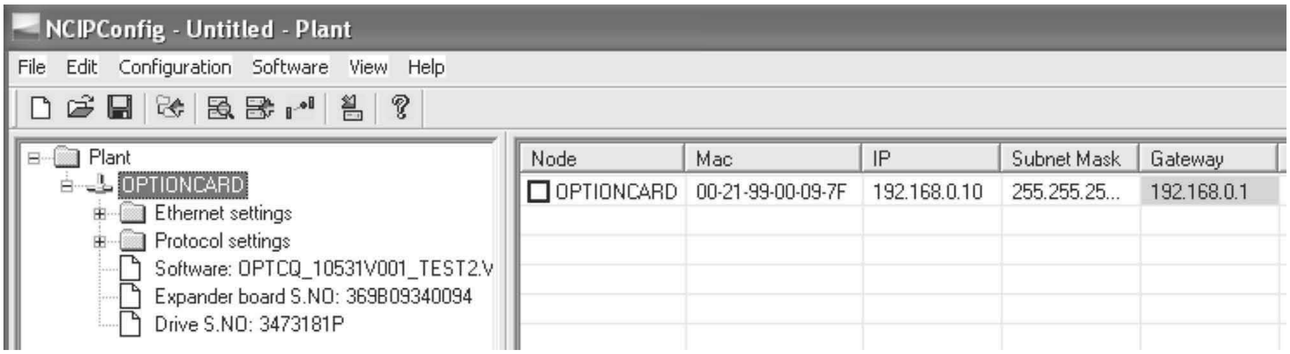
Step 1. Connect your PC to the Ethernet network with an Ethernet cable. You can also connect the PC directly to the OPTCQ. A crossover cable may be needed if your PC does not support automatic crossover function.

Step 2. Scan network nodes. Select Configuration --> Scan and wait until the devices connected to the bus in the tree structure are displayed to the left of the screen.

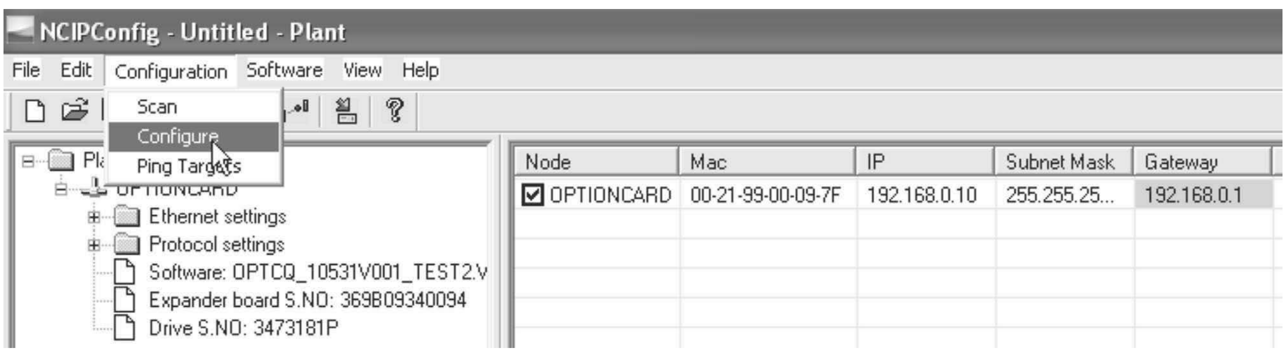
Note: Some switches block broadcast messages. In this case, each network node must be scanned separately. Read the manual under Help menu!



Step 3. Set IP addresses. Change the node's IP settings according to the network IP settings. The program will report conflicts with a red color in a table cell. Read the manual under Help menu.



Step 4. Send configuration to boards. In the table view, check the boxes for boards whose configuration you want to send and select Configuration, then Configure. Your changes are sent to the network and will be valid immediately.

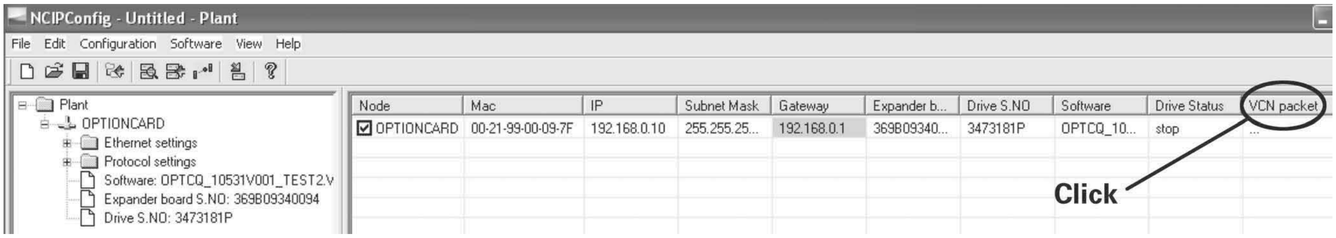


Update OPTCQ Option Board Program with the NCIPConfig Tool

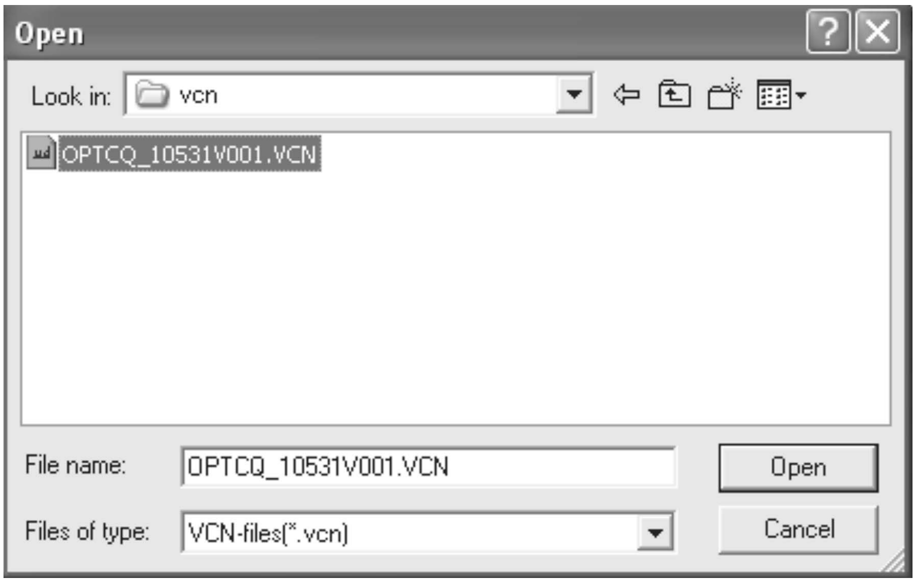
In some cases it may be necessary to update the option board's firmware. Differing from other OPTC option boards, the EtherNet/IP option board's firmware is updated with the NCIPConfig tool.

Note: The IP addresses of the PC and the option board must be in the same area when the software is loaded.

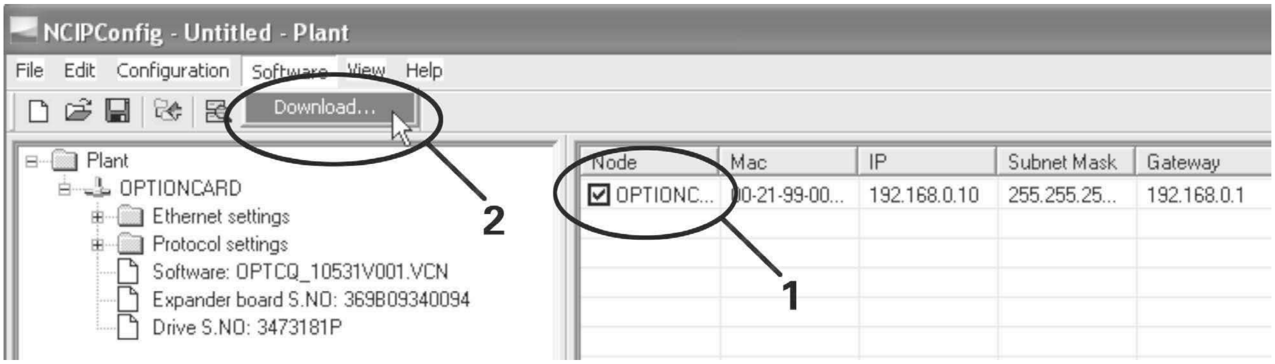
To start the firmware update, scan the nodes in the network according to the instructions in Assembly Instances Implemented by the OPTCQ Option Board section on **Page 20**. Once you can see all nodes in the view, you can update the new firmware by clicking the VCN packet field in NCIPCONFIG's table view on the right.



After clicking the VCN packet field, a file open window where you can choose a new firmware packet is displayed.



Send the new firmware packet to the option board by checking its box in the "VCN Packet" field at the right corner of the table view. After selecting all nodes to be updated by checking the boxes, send the new firmware to the board by selecting "Software" then "Download."

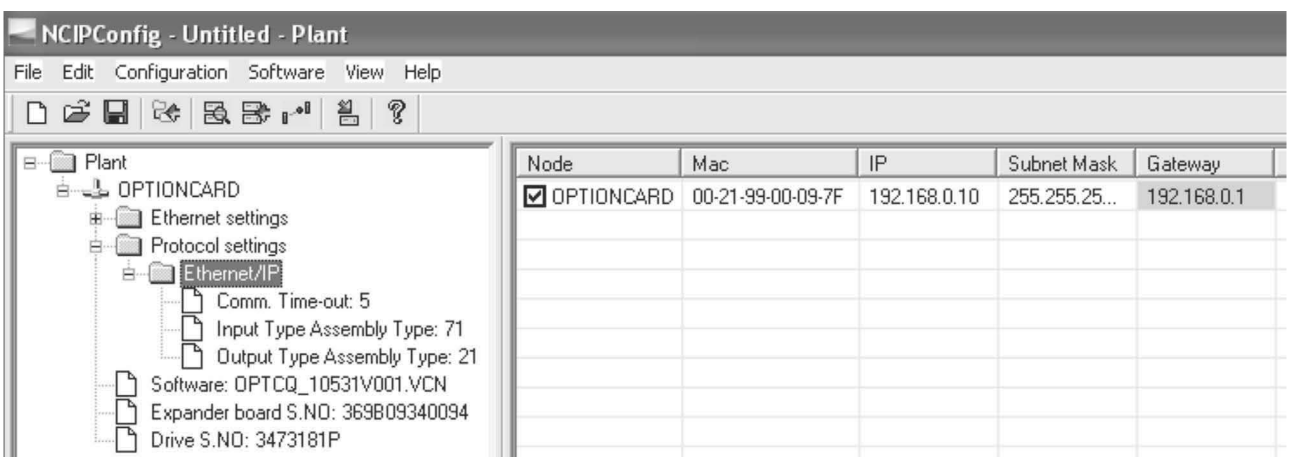


Note: Do not do a power-up cycle within 1 minute after downloading the option board software. This may cause the option board to go to "Safe Mode." This situation can only be solved by re-downloading the software. The Safe Mode triggers a fault code (F54). The Board slot error F54 may also appear due to a faulty board, a temporary malfunction of the board, or disturbance in the environment.

Configure Option Board Parameters

These features are available from NCIPConfig tool version 1.6.

In the tree-view, expand the folders until you reach the board parameters. Slowly double-click the parameter (Comm. Time-out in figure below) and enter the new value. New parameter values are automatically sent to the option board after the modification is complete.



Note: If the fieldbus cable is broken at the Ethernet board end or removed, a fieldbus error is immediately generated.

Additional Information

Handling of the NetCtrl bit (Network Control)

If NetCtrl bit is set, then Output Instance's Control Word is sent to the Drive. Additionally, BusCtrl bit of the FBFixedControlWord is set.

Handling of the NetRef bit (Network Reference)

If NetRef bit is set, then Torque Reference and Speed Reference are sent to the Drive. Additionally, BusRef bit of the FBFixedControlWord is set.

Handling of the NetProc bit in Assembly Instance 25 (Net Process)

If NetProc bit is set, then Process Reference is sent to the Drive.

Handling of RefFromNet and CtrlFromNet bits

RefFromNet and CtrlFromNet bits are set if value of REMOTEIndication (Drive Parameter Index: 552) is more than 0.

License for LWIP

Eaton is dedicated to ensuring that reliable, efficient and safe power is available when it's needed most. With unparalleled knowledge of electrical power management across industries, experts at Eaton deliver customized, integrated solutions to solve our customers' most critical challenges.

Our focus is on delivering the right solution for the application. But, decision makers demand more than just innovative products. They turn to Eaton for an unwavering commitment to personal support that makes customer success a top priority. For more information, **visit www.eaton.com/electrical.**

Eaton Corporation
Electrical Sector
1111 Superior Ave.
Cleveland, OH 44114
United States
Eaton.com

Printed in USA
Publication No. MN04002005E / Z11293
August 2012

Powering Business Worldwide