

Instruction Manual

Absolute Rotary Encoder with EtherNet/IP

EVS58N-*-IZ**
EVM58N-*-IZ**
ESS58N-*-IZ**
ESM58N-*-IZ**



1	1. Introduction	3
1.1	Control and Information Protocol (CIP)	4
1.2	Object model	4
2	Data Transmission	5
2.1	Implicit Messaging I/O Connection	5
2.2	Explicit Messaging	7
2.3	Ethernet Link Object	9
2.4	Setting parameters with scanners	11
3	Diagnostic	14
4	Programmable Parameters	15
4.1	Encoder parameters for Position Sensor Object Class 23hex.	15
5	Installation	18
5.1	Electrical connection	18
5.2	Ethernet cables	18
6	Power On	19
7	Installation	19
7.1	Rockwell configuration tools	19
7.2	Schneider configuration tools	32
8	FAQ	35
8.1	Problem:	35
8.2	Problem:	35
9	Glossar	35

Used symbols



Warning

This symbol warns the user of potential danger. Nonobservance may lead to personal injury or death and/or damage to property.



Attention

This symbol warns the user of potential device failure. Nonobservance may lead to the complete failure of the device or other devices connected.



Note

This symbol calls attention to important notes.

Security advice



Warning

***This product must not be used in applications, where safety of persons depend on the correct device function.
This product is not a safety device according to EC machinery directive.***

Notes

These operating instructions refer to proper and intended use of this product. They must be read and observed by all persons making use of this product. This product is only able to fulfill the tasks for which it is designed if it is used in accordance with specifications of Pepperl+Fuchs.

The warrantee offered by Pepperl+Fuchs for this product is null and void if the product is not used in accordance with the specifications of Pepperl+Fuchs.

Changes to the devices or components and the use of defective or incomplete devices or components are not permitted. Repairs to devices or components may only be performed by Pepperl+Fuchs or authorized work shops. These work shops are responsible for acquiring the latest technical information about Pepperl+Fuchs devices and components. Repair tasks made on the product that are not performed by Pepperl+Fuchs are not subject to influence on the part of Pepperl+Fuchs. Our liability is thus limited to repair tasks that are performed by Pepperl+Fuchs.

The preceding information does not change information regarding warrantee and liability in the terms and conditions of sale and delivery of Pepperl+Fuchs.

This device contains sub-assemblies that are electrostatically sensitive. Only qualified specialists may open the device to perform maintenance and repair tasks. Touching the components without protection involves the risk of dangerous electrostatic discharge, and must be avoided. Destruction of basic components caused by an electrostatic discharge voids the warrantee!

Subject to technical modifications.

Pepperl+Fuchs GmbH in D-68301 Mannheim maintains a quality assurance system certified according to ISO 9001.



1 1. Introduction

Absolute rotary encoders provide a definite value for every possible position. All these values are reflected on one or more code discs. The beams of infrared LEDs are sent through code discs and detected by Opto-Arrays. The output signals are electronically amplified and the resulting value is transferred to the interface.

The absolute rotary encoder has a maximum resolution of 65536 steps per revolution (16 Bit). The Multi-Turn version can detect up to 16384 revolutions (14 Bit). Therefore the largest resulting resolution is 30 Bit = 1.073.741.824 steps. The standard Single-Turn version has 13 Bit, the standard Multi-Turn version 25 Bit.

The integrated Ethernet interface of the absolute rotary encoder supports all necessary the EtherNet/IP functions.

The protocol supports the programming of the following additional functions in several ways:

Code sequence (Complement)

Resolution per revolution

Total resolution

Preset value

IP-Address

The general use of absolute rotary encoders with EtherNet/IP interface is guaranteed. The data will transmit in a standard Ethernet frame in the data section, see at the bottom of this side the pink field with the blue frame.

The MAC Address for each encoder is available on the type label.

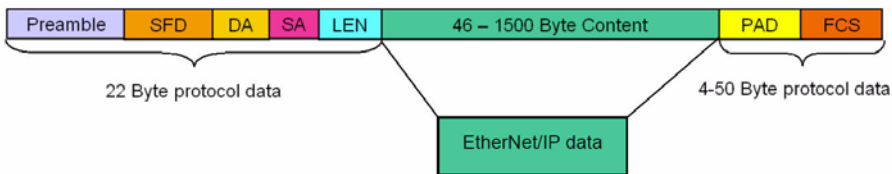
The IP address can be programmed with DHCP or BOOTP by configuration tools of the PLC.

General information's about EtherNet/IP are available:

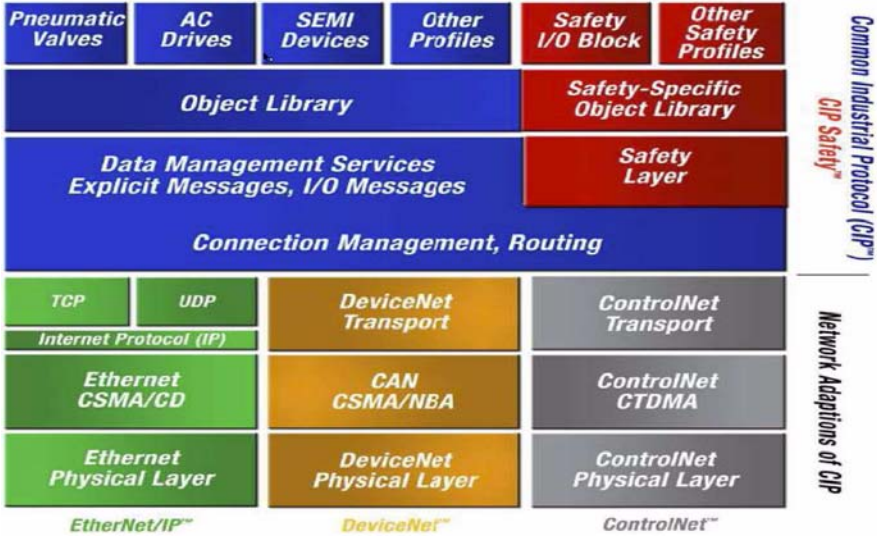
www.ethernetip.de (German)

www.odva.org/default.aspx?tabid=67 (English)

Setup of an Ethernet data package on layer 2

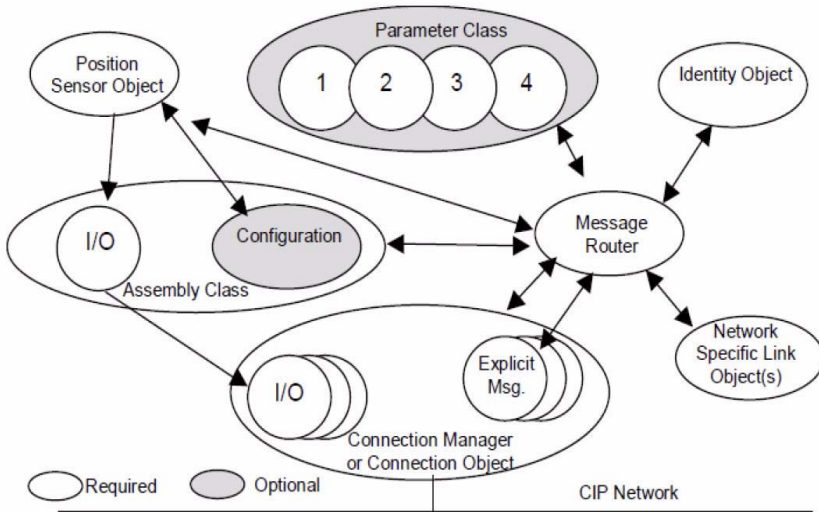


1.1 Control and Information Protocol (CIP)



1.2 Object model

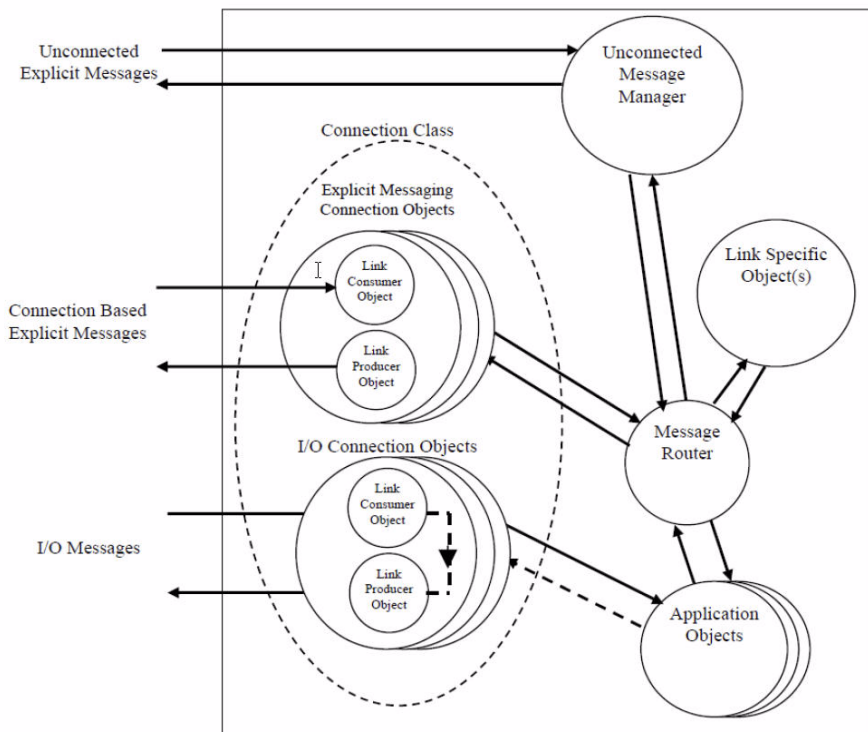
EtherNet/IP describes all data and functions of a device considering an object model. By means of that object oriented description, a device can be defined complete with single objects. A object is defined across the centralization by associated attributes (e.g. process data), its functions (read- or write access of a single attribute) as well as by the defined behavior. The absolute rotary encoder support the Encoder Device Type: 22_{hex} or Generic Device Type: 0_{hex}. This is programmable, see chapter 4.1.6. All parameters will be used with Big Endian notation.



Part No. xxxxxx
 Document No. DOCT-2115A
 Date of Issue: 2nd January, 2012

2 Data Transmission

The data transmission in the EtherNet/IP network is realized by implicit or explicit messaging. Explicit messages are split in unconnected and connection based versions. Unconnected messages will be use i.e. by EtherNet/IP scanners.



2.1 Implicit Messaging I/O Connection

Provide dedicated, special-purpose communication paths between a producing application and one or more consuming applications for the purpose of moving application-specific data. This is often referred to as implicit messaging. Class 0 and 1 are supported.

2.1.1 I/O Assembly Instances

Instance	Type	Name
1	Input	Position Value
3	Input	Position Value and Velocity

2.1.1.1 Data Attribute Format

Instance	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	0	Position Value (low Byte)							
	1								
	2								
	3								
3	0	Position Value (high Byte)							
	1								
	2								
	3								
	4	Velocity (low Byte)							
	5								
	6								
	7								

2.1.2 Data Mapping

Data Component Name	Class		Instance		Attribute	
	Name	Number	Number	Name	Number	
Position Value	Position Sensor	23 _{hex}	1	Position Value	0A _{hex}	
Velocity	Position Sensor	23 _{hex}	1	Velocity	18 _{hex}	

2.1.3 Data Mapping (Parameter)

On every Forward Open Request, the following parameters, will be sent from the controller to the encoder.

Assembly Instance Configuration: 7, size 12 Bytes

Configuration Parameter Name	Class		Instance		Attribute	
	Name	Number	Number	Name	Number	
Direct Counting Toggle	Position Sensor	23 _{hex}	1	Direct Counting Toggle	0C _{hex}	
Scaling Function Control	Position Sensor	23 _{hex}	1	Scaling Function Control	0E _{hex}	
Measuring units per Revolution	Position Sensor	23 _{hex}	1	Measuring Units per Span	10 _{hex}	
Total Measuring Range in measuring units	Position Sensor	23 _{hex}	1	Total Measuring Range in measuring units	11 _{hex}	
Velocity Format	Position Sensor	23 _{hex}	1	Velocity Format	19 _{hex}	

2.1.3.1 Data Offset

Byte Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Direction Counting Toggle							
1	Scaling Function Control							
2	Measuring units per Revolution (low byte)							
3								
4								
5	Measuring units per Revolution (high byte)							
6	Total Measuring Range in measuring units (low byte)							
7								
8								
9	Total Measuring Range in measuring units (high byte)							
10	Velocity Format (low byte)							
11	Velocity (high byte)							

2.1.4 Connection Path

Is made up of a byte stream that defines the application object to which a connection instance applies. This path will be created from the configuration tools and are available in the EDS file too. This path will sent during power up to the encoder. For some tools it is necessary to use the connection path as parameter:

[20] [04] [24 07]	[2C 06] [2C 01]	[80 06 00 01 00100000 00200000 041F]
Segment Groups	Segment	Description
Application Path	20 04	Assembly object class
	24 07	Instance segment type with Assembly Instance 7 (Configuration)
	2C 06	Assembly Instance 6 (Output controller to encoder)
	2C 01	I/O Assembly Instance 1 (Position value)
	80 06	Data segment with length of 6 Bytes
	00 01 00100000 00200000 041F	Configuration Data, see chapter 2.1.3.1 for details

2.2 Explicit Messaging

Provide generic, multi-purpose communication paths between two devices. These connections often are referred to as just Messaging Connections. Explicit Messages provide the typical request/response-oriented network communications. Class 2 and 3 are supported.

2.2.1 CIP Common Services

Supported Service Code	Service Name	Comment
05 _{hex}	Reset	Boot up of the encoder, the programmed parameter from the customer will use again
0E _{hex}	Get_Attribute_Single	Read out attribute from the encoder
10 _{hex}	Set_Attribute_Single	Write attribute to the encoder
15 _{hex}	Restore	Restore the saved parameters
16 _{hex}	Save	Save the parameters from chapter 2.1.3 to the nonvolatile memory

Position Sensor Objects

Instance Attributes (Get: read, Set: write + read)

Class Code: 23_{hex}

Attrib. ID	Access	Name	Data Type	Description
01 _{hex}	Get	Number of Attributes	USINT	Number of supported Attributes
02 _{hex}	Get	Attribute List	Array of USINT	List of supported Attribute
0A _{hex}	Get	Position Value Signed	DINT	Current position signed
0B _{hex}	Get	Position Sensor Type	UINT	Specifies the device type
0C _{hex}	Set	Direction Counting Toggle	Boolean	Controls the code sequence clockwise or counterclockwise
0E _{hex}	Set	Scaling Function Control	Boolean	Scaling function on/off
10 _{hex}	Set	Measuring units per Span	UDINT	Resolution for one revolution
11 _{hex}	Set	Total Measuring Range in Measuring Units	UDINT	Total resolution
13 _{hex}	Set	Preset Value	DINT	Setting a defined position value
18 _{hex}	Get	Velocity Value	DINT	Current speed in format of attribute 19 _{hex} and 2A _{hex}
19 _{hex}	Set	Velocity Format	ENGUINT	Format of the velocity attributes
29 _{hex}	Get	Operating Status	BYTE	Encoder diagnostic operating status
2A _{hex}	Get	Physical Resolution Span	UDINT	Resolution for one revolution
2B _{hex}	Get	Number of Spans	UINT	Number of revolutions
33 _{hex}	Get	Offset Value	DINT	Shift position value with the calculated value
64 _{hex}	Set	Device Type	DINT	Encoder device = 22 _{hex} , Generic device = 0 (default)
65 _{hex}	Set	Endless Shaft	DINT	Off = 0, On = 1, Auto = 2
66 _{hex}	Set	Velocity Filter	DINT	Fine = 0, Middle = 1, Raw = 2

2.3 TCP/IP Interface Object

The TCP/IP Interface Object provides the mechanism to configure a device's TCP/IP network interface. With this parameter it is possible i.e. to read or write the device's IP Address and Network Mask.

Class Code: F5_{hex}

Attribute ID	Access	Name	Data Type	Description
01 _{hex}	Get	Status	DWORD	Interface status, details in chapter 2.3.1
02 _{hex}	Get	Configuration Capability	DWORD	Interface capability flags, details in chapter 2.3.2
03 _{hex}	Set	Configuration Control	DWORD	Interface control flags, details in chapter 2.3.3
04 _{hex}	Get	Physical Link Object	STRUCT of:	Path to physical link object
		Path size	UINT	Size of path
		Path	Padded EPATH	Logical segments identifying the physical link object
05 _{hex}	Set	Interface Configuration	STRUCT of:	TCP/IP network interface configuration
		IP Address	UDINT	The device's IP address
		Network Mask	UDINT	The device's network mask
06 _{hex}	Set	Host Name	STRING	

2.3.1 Status Instance Attribute (01_{hex})

Bit(s)	Called	Definition
0 - 3	Interface Configuration Status	Indicates the status of the Interface Configuration attribute. 0 = The Interface Configuration attribute has not been configured. 1 = The Interface Configuration attribute contains valid configuration obtained from BOOTP, DHCP or nonvolatile storage. 2 = The Interface Configuration attribute contains valid configuration, obtained from hardware settings (e.g.: pushwheel, thumbwheel, etc.) 3-15 = Reserved for future use.
4	Mcast Pending	Indicates a pending configuration change in the TTL Value and/or Mcast Config attributes. This bit shall be set when either the TTL Value or Mcast Config attribute is set, and shall be cleared the next time the device starts.
5 - 31	Reserved	Reserved for future use and shall be set to zero.

2.3.2 Configuration Instance Attribute (02_{hex})

Bit(s)	Called	Definition
0	BOOTP Client	1 (TRUE) shall indicate the device is capable of obtaining its network configuration via BOOTP.
1	DNS Client	Not supported
2	DHCP Client	1 (TRUE) shall indicate the device is capable of obtaining its network configuration via DHCP.
3	DHCP-DNS Update	Not supported
4	Configuration Settable	1 (TRUE) shall indicate the Interface Configuration attribute is settable. Some devices, for example a PC or workstation, may not allow the Interface Configuration to be set via the TCP/IP Interface Object.
5 - 31	Reserved	Reserved for future use and shall be set to zero.

2.3.3 Configuration Control Inst. Attribute (04_{hex})

Bit(s)	Called	Definition
0 - 3	Startup Configuration	Determines how the device shall obtain its initial configuration at start up. 0 = The device shall use the interface configuration values previously stored (for example, in non-volatile memory or via hardware switches, etc). 1 = The device shall obtain its interface configuration values via BOOTP. 2 = The device shall obtain its interface configuration values via DHCP upon start-up. 3-15 = Reserved for future use.

2.3.4 Physical Link Object (05_{hex})

This attribute identifies the object associated with the underlying physical communications interface (e.g., an 802.3 interface). There are two components to the attribute: a Path Size (in UINTs) and a Path. The Path shall contain a Logical Segment, type Class, and a Logical Segment, type Instance that identifies the physical link object. The maximum Path Size is 6 (assuming a 32 bit logical segment for each of the class and instance).

The physical link object itself typically maintains link-specific counters as well as any link specific configuration attributes. If the CIP port associated with the TCP/IP Interface Object has an Ethernet physical layer, this attribute shall point to an instance of the Ethernet Link Object (class code = F6_{hex}). When there are multiple physical interfaces that correspond to the TCP/IP interface, this attribute shall either contain a Path Size of 0, or shall contain a path to the object representing an internal communications interface (often used in the case of an embedded switch).

For example, the path could be as follows:

Path	Meaning
0-3	[20] = 8 bit class segment type; [F6] = Ethernet Link Object class; [24] = 8 bit instance segment type; [01] = instance 1.

2.3.5 Interface Configuration (06_{hex})

Name	Meaning
IP Address	The device's IP address. Value of 0 indicates no IP address has been configured. Otherwise, the IP address shall be set to a valid Class A, B, or C address and shall not be set to the loopback address (127.0.0.1).
Network mask	The device's network mask. The network mask is used when the IP network has been partitioned into subnets. The network mask is used to determine whether an IP address is located on another subnet. Value of 0 indicates no network mask address has been configured.

2.3.6 Host Name

Name	Meaning
Host Name	ASCII characters. Maximum length is 64 characters. Shall be padded to an even number of characters (pad not included in length). A length of 0 shall indicate no Host Name is configured.

2. 4 Ethernet Link Object

Class Code: F6_{hex}

Attribute ID	Access	Name	Data Type	Description	Semantics of Values
01 _{hex}	Get	Revision	UINT	Revision of this object	The minimum value shall be 1. Shall be 2 or greater if instance attribute 6 is implemented. Shall be 3 if any instance attributes 7-10 are implemented. The maximum value shall be 3.
02 _{hex}	Get	Max Instance	UINT	Maximum instance number of an object currently created in this class level of the device	The largest instance number of a created object at this class hierarchy level
03 _{hex}	Get	Number of Instances	UINT	Number of object instances currently created at this class level of the device	The number of object instances at this class hierarchy level

2.4.1 Instance Attributes

ID	Access	Name	Data Type	Description of Attribute	Semantics of Values
1	Get	Interface Speed	UINT	Interface speed currently in use	Speed in Mbps (e.g., 10, 100)
2	Get	Interface Flags	DWORD	Interface status flags	See chapter 2.4.1
3	Get	Physical Address	ARRAY of 6 USINTs	MAC layer address	Displayed format "XX-XX-XXXX-XX-XX"
7	Get	Interface Type	USINT	Type of interface	1 = The interface is internal to the device, i.e. in the case of an embedded switch 2 = Twisted-pair (e.g. 100Base-TX)
8	Get	Interface State	USINT	Current state of the interface	0 = No link 1 = The interface is enabled and is ready to send and receive data
10	Get	Interface Label	SHORT_STRING	Human readable identification	„Internal switch“ or „External Port 1“ or „External Port 2“

2.4.2 Instance Flags

Bit(s)	Called	Definition
0	Link Status	Indicates whether or not the Ethernet 802.3 communications interface is connected to an active network. 0 indicates an inactive link; 1 indicates an active link. The determination of link status is implementation specific. In some cases devices can tell whether the link is active via hardware/driver support. In other cases, the device may only be able to tell whether the link is active by the presence of incoming packets.
1	Half/Full Duplex	Indicates the duplex mode currently in use. 0 indicates the interface is running half duplex; 1 indicates full duplex. Note that if the Link Status flag is 0, then the value of the Half/Full Duplex flag is indeterminate.
2 - 4	Negotiation Status	Indicates the status of link auto-negotiation 0 = Auto-negotiation in progress. 1 = Auto-negotiation and speed detection failed. Using default values for speed and duplex. Default values are product-dependent; recommended defaults are 10Mbps and half duplex. 2 = Auto negotiation failed but detected speed. Duplex was defaulted. Default value is product-dependent; recommended default is half duplex. 3 = Successfully negotiated speed and duplex. 4 = Auto-negotiation not attempted. Forced speed and duplex.
5	Manual Setting Requires Reset	0 indicates the interface can activate changes to link parameters (autonegotiate, duplex mode, interface speed) automatically. 1 indicates the device requires a Reset service be issued to its Identity Object in order for the changes to take effect.
6	Local Hardware Fault	0 indicates the interface detects no local hardware fault; 1 indicates a local hardware fault is detected. The meaning of this is product-specific. Examples are an AUI/MMI interface detects no transceiver attached or a radio modem detects no antennae attached. In contrast to the soft, possible self-correcting nature of the Link Status being inactive, this is assumed a hardfault requiring user intervention.
7	Reserved	Shall be set to zero

2.4.3 Common Service

Service Code	Class	Instance*	Service Name	Description of Service
0E _{hex}	Conditional	Required	Get_Attribute_Single	Returns the contents of the specified attribute
10 _{hex}	n/a	Conditional	Set_Attribute_Single	Modifies a single attribute

Document No. DOCT-2115A

Part No. xxxxxx

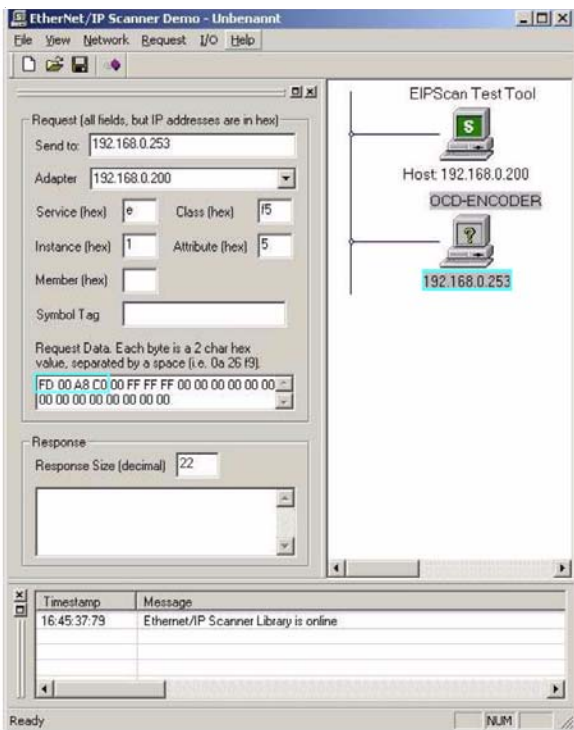
Date of Issue: 2nd January, 2012

2.4.4 Link Object Instances

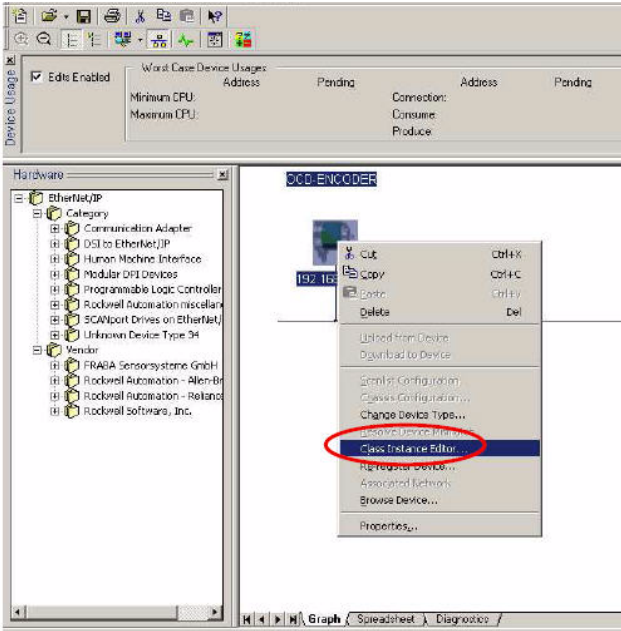
Instance	Description
1	Internal interface
2	Intern switch Port 1
3	Intern switch Port 2

2.5 Setting parameters with scanners

There are several external scanners for EtherNet/IP available. RS-NetWorks™ has one such scanner. In the figure is an example where the IP-Address (FD 00 A8 C0 complies 192.168.0.253) and the Gateway (00 FF FF FF complains 255.255.255.0) was read out of the encoder



In RSNetWorx is a scanner available too. In the next section is a sample to set the Preset value.



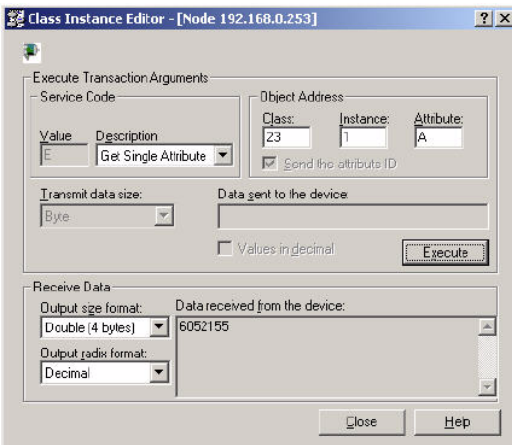
2.5.1 Read out position value

Get Single Attribute Position sensor value:

Class: 0x23 (Position sensor object)

Instance: 0x01

Attribute: 0x0A (Position Value)



2.5.2 Set preset value

Set Single Attribute Position Preset Value to 1

Class: 0x23 (Position sensor object)

Instance: 0x01

Attribute: 0x13 (Preset Value)

The screenshot shows the 'Class Instance Editor' window for Node 192.168.0.253. It is configured for a 'Set Single Attribute' transaction. The 'Service Code' is 10. The 'Object Address' is set to Class: 23, Instance: 1, and Attribute: 13. The 'Transmit data size' is 'Double (4 bytes)' and the 'Data sent to the device' is '00000001'. The 'Receive Data' section shows 'Data received from the device' as 'The execution was completed.' Buttons for 'Execute', 'Close', and 'Help' are visible.

2.5.3 Get preset value

Get Single Attribute Position Value

Class: 0x23 (Position sensor object)

Instance: 0x01

Attribute: 0x13 (Preset Value)

The screenshot shows the 'Class Instance Editor' window for Node 192.168.0.253. It is configured for a 'Get Single Attribute' transaction. The 'Service Code' is 1. The 'Object Address' is set to Class: 23, Instance: 1, and Attribute: A. The 'Transmit data size' is 'Double (4 bytes)' and the 'Data sent to the device' is empty. The 'Receive Data' section shows 'Data received from the device' as '1'. Buttons for 'Execute', 'Close', and 'Help' are visible.

3 Diagnostic

LED	Color	EtherNet/IP name	Description
Active1	Yellow	Network Status Indicator 1	Details in table 2
Link1	Green		
Active2	Yellow	Network Status Indicator 2	Details in table 2
Link2	Green		
Stat1	Green	Module Status Indicator	Details in table 1
Stat2	Red		

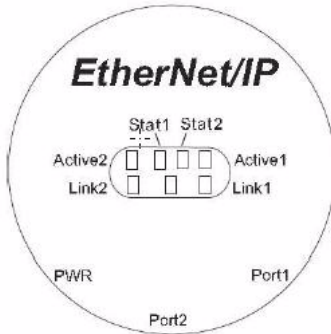


Table 1: Module Status Indicator Stat1/Stat2







LED		Summary	Requirements
Steady Off		No power	
Steady On Green		Device operational	If the device is operating correctly, the module status indicator shall be steady green
Flashing Green 1		Standby	If the device has not been configured including the IP-Address, the module status indicator shall be flashing green with 1 Hz
Flashing Green 2		Missing IP	If the device does not have an IP-Address, the module status indicator shall be flashing green with 4 Hz
Flashing Red		Minor fault	If the device has detected a recoverable minor fault. I.e. an incorrect or inconsistent configuration
Steady On Red		Major fault	If the device has detected a non-recoverable major fault
Flashing Red + Green		Self-test	While the device is performing its power up testing, the Stat1 and Stat2 LED shall be flashing red / green

Table 2: Network Status Indicator Stat2

Document No. DOCT-2115A

Part No. xxxxxx

Date of Issue: 2nd January, 2012

LED		Summary	Requirements
Steady Off		No power, no IP address	If the device does not have an IP address or is powered off
Steady Green		Connected	If the device has at least one established connection (even to the Message router)
Flashing Green		No connection	If the device has no established connections, but has obtained an IP address
Flashing Yellow		Connection timeout	If one or more of the connections in which this device is the target has timed out. This shall be left only if all timed out connections are reestablished or if the device is reset
Steady Yellow		Duplicate IP	If the device has detected that its IP address is already in use
Flashing Yellow / Green		Self-test	While the device is performing its power up testing, the Stat1 and Stat2 LED shall be flashing yellow / green

4 Programmable Parameters

4.1 Encoder parameters for Position Sensor Object Class 23_{hex}

4.1.1 Direction counting

This operating parameter can be used to select the code sequence. The parameter can set with Configuration Assembly and Explicit Messaging

Attribute ID	Default value	Value range	Data Type
0C _{hex}	0 _{hex}	0 _{hex} - 1 _{hex}	Boolean

The parameter code sequence (complement) defines the counting direction of the process value as seen on the shaft (clockwise or counter clockwise).

The counting direction is defined in the attribute 0C_{hex}:

Bit 0	Counting direction	Position values
0	CW	Increase
1	CCW	Decrease

4.1.2 Scaling function control

If the Scaling function control is deactivated then complains the output value the physical resolution.

Attribute ID	Default value	Value range	Data Type
0E _{hex}	1 _{hex}	0 _{hex} - 1 _{hex}	Boolean

If the Scaling function control is deactivated then complains the output value the physical resolution.l

Bit 0	Scaling function on/off
0	on
1	off

This parameter can be set with Configuration Assembly and Explicit Messaging

4.1.3 Resolution per revolution

The parameter resolution per revolution is used to program the encoder to set a desired number of steps per revolution. Each value between 1 and the maximum (see type label) can be realized.

The parameter can set with Configuration Assembly and Explicit Messaging.

Attribute ID	Default value	Value range	Data Type
10 _{hex}	(*)	0 _{hex} - 10000 _{hex}	Double Integer32

(*) see type label, Maximum resolution:

16 Bit Encoder: 10,000_{hex} (65,536)

When the value is set larger than 8192 for a 13Bit encoder, the process value of the encoder will not be single stepped and values will be skipped while rotating the shaft. So, it is recommended, to keep the measuring steps per revolution below 8192 measuring steps.

4.1.4 Total resolution

This value is used to program the desired number of measuring steps over the total measuring range. This value must not exceed the total resolution of the encoder with 25 bit = 33,554,432 steps. Please note the value written on the type shield. The parameter can set with Configuration Assembly and Explicit Messaging

Attribute ID	Default value	Value range	Data Type
11 _{hex}	(*)	0 _{hex} - 40,000,000 _{hex}	Unsigned Integer 32

(*) see type shield

Maximum total resolution

30 Bit Encoder: 40,000,000_{hex} (1,073,741,824)

Attention:

The following formula letters will be used:

- PGA Physical total resolution of the encoder (see type label)
- PAU Physical resolution per revolution (see type label)
- GA Total resolution (customer parameter)
- AU Resolution per revolution (customer parameter)

If the desired resolution per revolution is less than the physical resolution per revolution of the encoder, then the total resolution must be entered as follows:

Total resolution

$$GA = PGA * AU / PAU, \text{ if } AU < PAU$$

Example: Customer requirement: AU = 2048,

Encoder type shield: PGA=25 bit, PAU=13 bit

$$GA = 16777216 * 2048 / 8192$$

$$GA = 8388608$$

If the total resolution of the encoder is less than the physical total resolution, the parameter total resolution must be a multiple of the physical total resolution:

- k = PGA / GA
- k = integer

4.1.5 Preset value

The preset value is the desired position value, which should be reached at a certain physical position of the axis. The position value of the encoder is set to the desired process value by the parameter preset. The preset value must not exceed the parameter total measuring units. The parameter can set with Explicit Messaging.

Set the preset value only in standstill!

Attribute ID	Default value	Value range	Data Type
13 _{hex}	0 _{hex}	0 _{hex} - total measuring range	Unsigned Integer 32

4.1.6 Velocity Format

Default value for Velocity Format is steps persecond. This parameter can be set with Configuration Assembly and Explicit Messaging.

Attribute ID	Default value	Value range	Data length
19 _{hex}	1F04 _{hex}	1F04 _{hex}	Steps per second
		1F05 _{hex}	Steps per millisecond
		1F06 _{hex}	Steps per microsecond
		1F07 _{hex}	Steps per minute
		1F0F _{hex}	RPM

4.1.7 Velocity Filter

To manage the noise of the velocity it is possible to switch between three classes. See the diagrams to see the differences.

Attribute ID	Default value	Value range	Description	Data Type
66 _{hex}	0 _{hex}	0 _{hex} /1 _{hex} /2 _{hex}	0 = Fine, 1 = Middle, 2 = Raw	Double Integer

Diagram 1

Diagram 2

Diagram 3

4.1.8 Device Type

The EtherNet/IP interface supports the Encoder Device with device type 22 hex functionality or the Generic Device type 0hex according to the CIP specification. As all controllers do not support the Encoder Device the encoder changed to the Device Type. This parameter can set only with Explicit Messaging.

Attribute ID	Default value	Value range	Data Type
64 _{hex}	0 _{hex}	0 _{hex} /22 _{hex}	Double Integer

4.1.9 Endless Shaft

Normally the period, i.e. "Total resolution"/"measuring units" per revolution must be an integer and it must fit an integer number of times (integer multiple) into 4096 for an encoder with 12 Bit for the revolutions. So the following equation must apply:

(4096 x measuring units per revolution) / Total resolution = integer

But with this EtherNet/IP encoder it is possible to solve this problem. If the Endless Shaft is activated then this problem will be solved by the encoder. The default value is Auto. In this case the encoder checks if the parameters need the endless shaft. The parameter can be set only with Explicit Messaging.

Note: The internal software routine only works if the encoder is in operation. If it is necessary to turn the encoder shaft more than 1024 revolutions without power supply this can lead to problems (the internal routine will not work without power supply). In this case the rule ahead should be observed even with new devices.

Attribute ID	Default value	Value range	Description	Data Type
65 _{hex}	2 _{hex}	0 _{hex} /1 _{hex} /2 _{hex}	0 = off, 1 = on, 2 = Auto	Double Integer

5 Installation

5.1 Electrical connection

The rotary encoder is connected by a 4 pin M12 connector for the power supply and two 4 pin, D-coded M12 connector for Ethernet. The Encoder uses a second D-coded connector and provides an integrated switch functionality.

On or in the packaging of the connector is the mounting description.

Connector Ethernet

4 pin female, D-coded

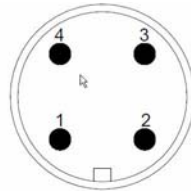
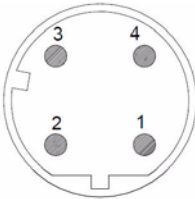
Pin Number	Signal
1	Tx+
2	Rx+
3	Tx-
4	Rx-

Connector power supply

4 pin male, A-coded

Pin Number	Signal
1	US (15 - 30 VDC)
2	n.c.
3	GND (0 V)
4	n.c.

Sketch as seen on the encoder



5.2 Ethernet cables

5.2.1 RJ45 – M12 crossed

Signal	RJ45 Pin	M12 Pin	Signal
Tx+	1	2	Rx+
Tx-	2	4	Rx-
Rx+	3	1	Tx+
Rx-	6	3	Tx-

5.2.2 RJ45 – M12 straight

Signal	RJ45 Pin	M12 Pin	Signal
Tx+	3	1	Tx+
Tx-	6	3	Tx-
Rx+	1	2	Rx+
Rx-	2	4	Rx-

5.2.3 RJ45 – M12 - M12 crossed

Signal	M12 Pin	M12 Pin	Signal
Tx+	1	2	Rx+
Tx-	3	4	Rx-
Rx+	2	1	Tx+
Rx-	4	3	Tx-

6 Power On

After power on the LED's on the absolute rotary encoder will flash between green and red or yellow.

7 Installation

7.1 Rockwell configuration tools

7.1.1 Setting IP-Address (BOOTP/DHCP)

To set the IP Address there are special tools available. I.e. the BOOTP/DHCP Server is installed with the software package from RSNetWorx™. The server scan the network for the MAC Addresses of all products with active BOOTP or DHCP. If one MAC address is selected in the Request History then the IP Address can be set by the “Add to Relation List” button. The MAC Address of each EtherNet/IP encoder is available on the type label.

Note:

After a power up the encoder send the BOOTP or DHCP request often. But after several time comes no answer the frequency of requests decrease. A power up after a longer pause could solve the missing requests.

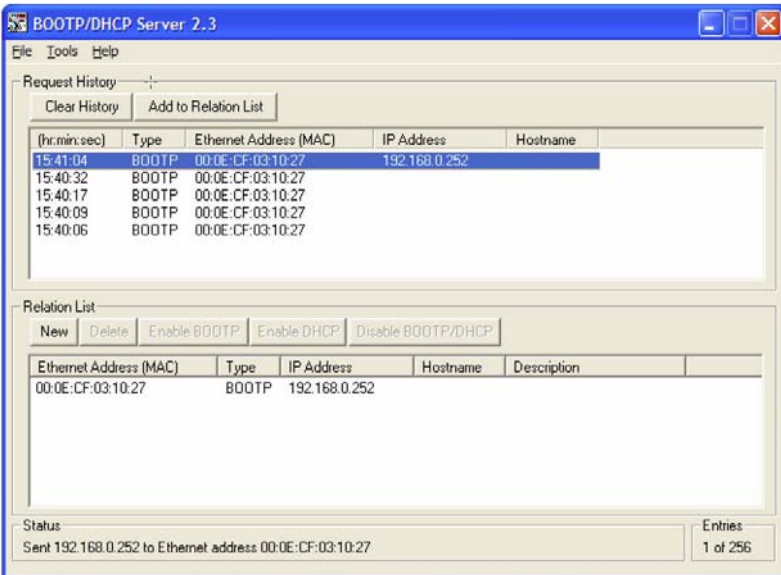
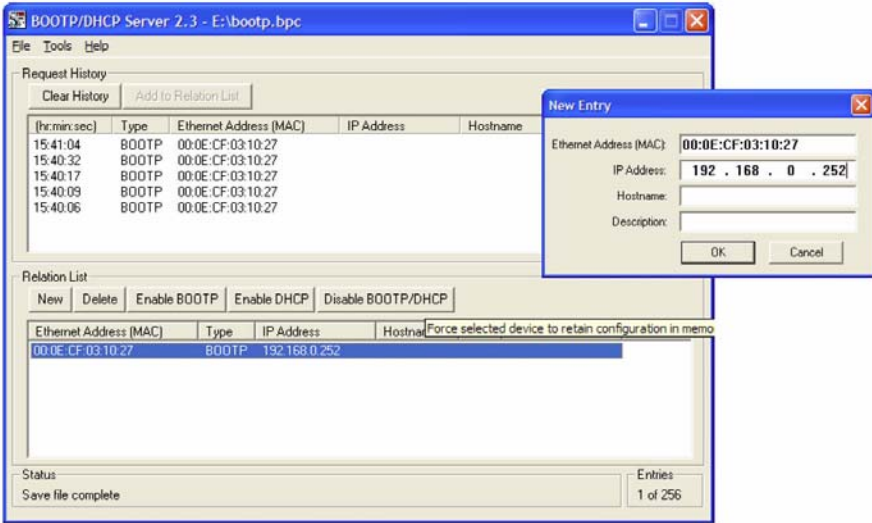
If not all encoders are listed in the BOOTP/DHCP Server then check the following points:

- LED status of the encoder OK?
- Is the Network setting correct?
- Is the BOOTP and/or DHCP enabled?

If the encoder has got his IP-Address, the BOOTP and DHCP must be disabled with the corresponding button. Otherwise the encoder start up to get a new IP-Address again. After setting the IP-Address the Status LED is flashing with 1 Hz. But in this case save the configuration in the File menu, because the products cannot be found by the BOOTP/DHCP Server. After loading this file the MAC Addresses and IP-Addresses are available and BOOTP or DHCP can be activated by the corresponding button. Possible IP-Range:

Class A-C (0.0.0.0 – 223.255.255.255) without Loopback range (127.x.x.x)

Referenced IP-Address range: 192.168.0.x

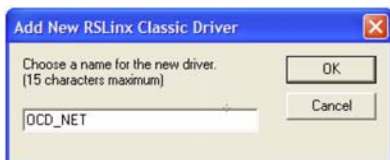
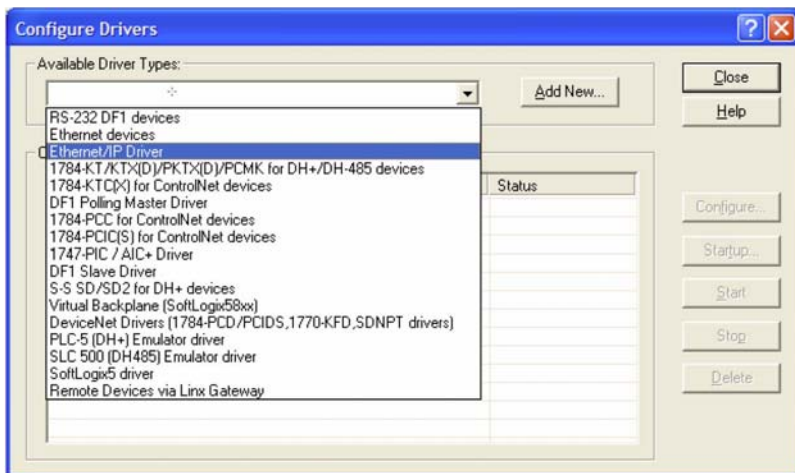


After setting the IP-Address with this tool the IPAddress will be available only after the next BOOTP request.

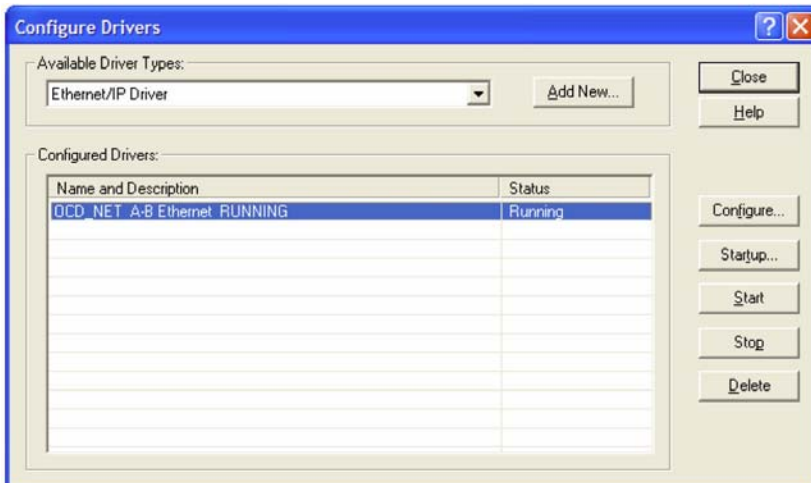
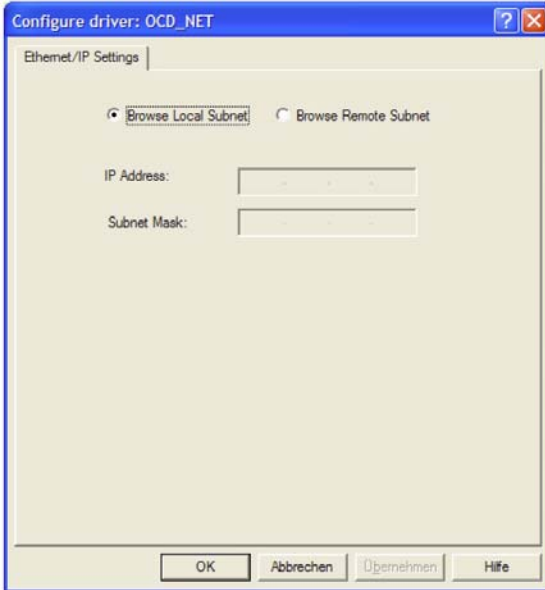
7.1.2 Configuration RSLinx Classic™

RSLinx™ is a complete communication server providing plant-floor device connectivity for a wide variety of Rockwell Software applications such as RSLogix™, RSNetWorx™,...

To start a new project add first a new RSLinx Classic™ Driver for EtherNet/IP under *Communications Configuration Drivers* and input the name.



Use Browse Local Subnet to find the EtherNet/IP components in the network. The status should be “Running”. Then push the Close button to finish this configuration.



7.1.3 RSNetWorx™

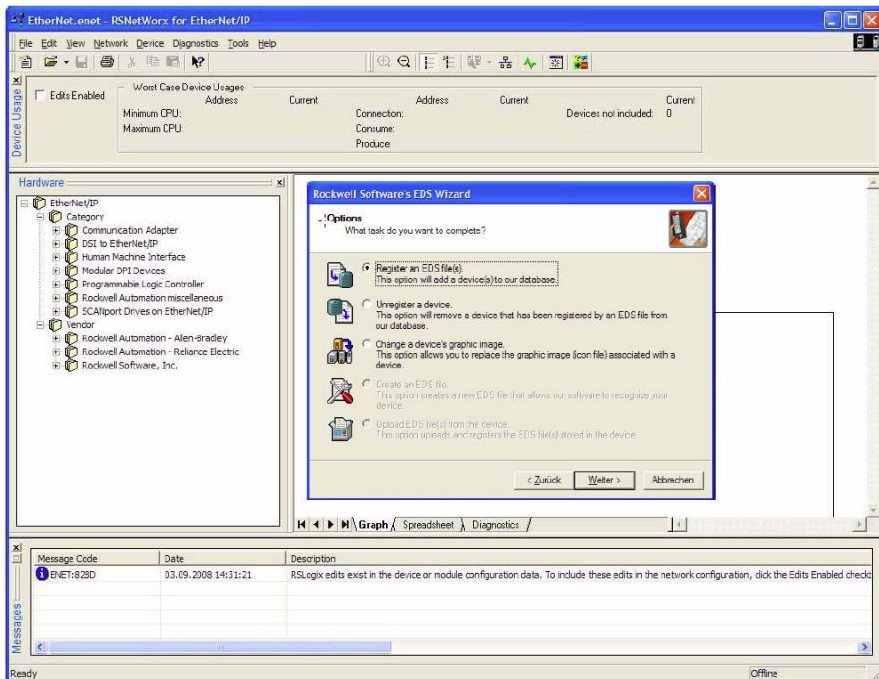
RSNetWorx™ products provide design and configuration management services for EtherNet/IP. The program defines and configures the devices on the network quickly through a simple software interface. This definition can take place offline using drag and drop operations or online by using RSLinx® to browse a EtherNet/IP network.

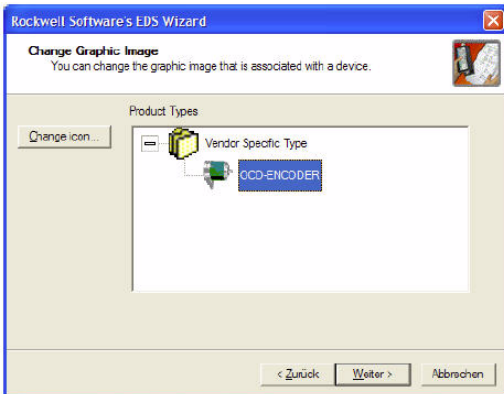
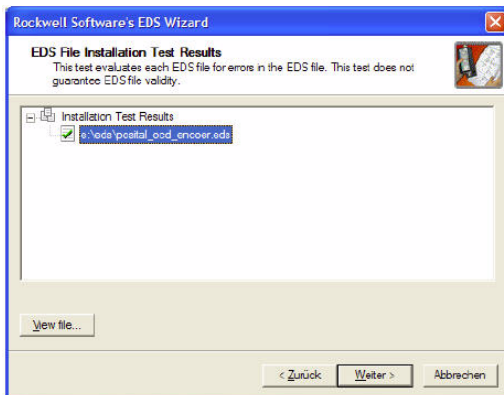
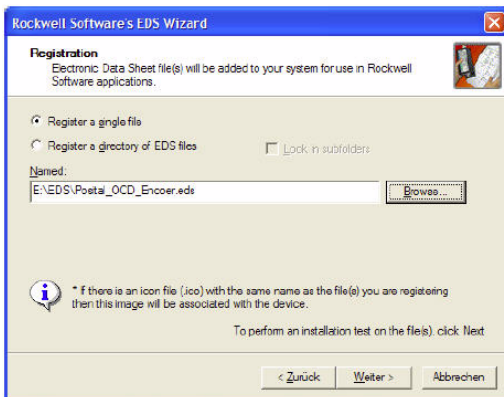
EDS Wizard

The EDS File contains information about device specific parameters as well as possible operating modes of the encoder. With this file you have a data sheet in an electronic format, which can be used to configure the device in the network, for example with RSNetWorx™ from Rockwell.

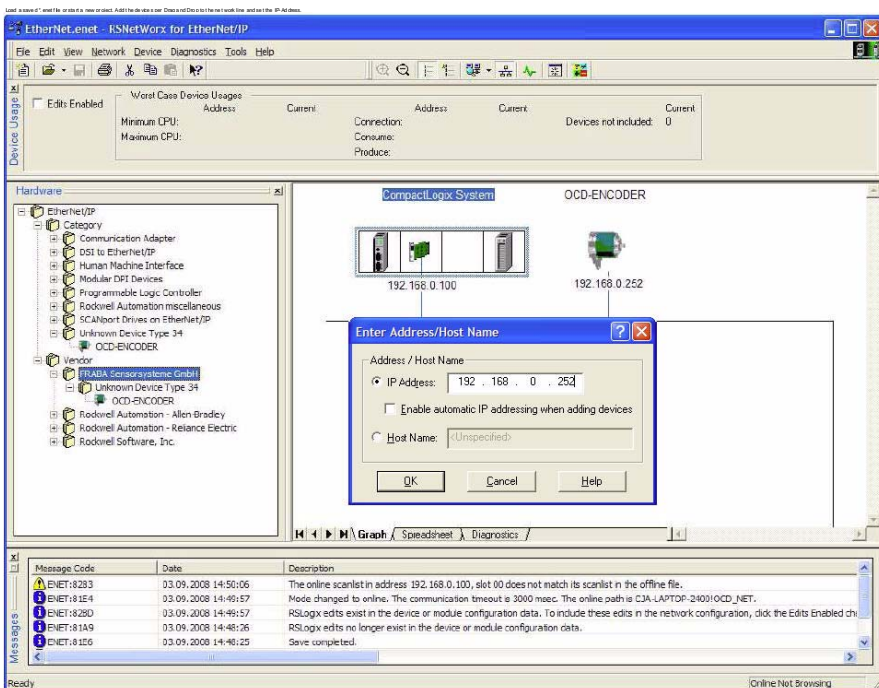
In this sample the PLC uses address 192.168.0.100 and the encoder 192.100.0.252.


To install the EDS file the EDS Wizard has to be started, that can be done in the menu *Tools/EDS Wizard*. If the EDS Wizard is activated successfully the *Register an EDS File(s)* has to be chosen and after that the button *Weiter*. In the next step the *Register a directory of EDS files* has to be chosen and with *Browse* the path of the EDS file(s). That is indicated in the next pictures.

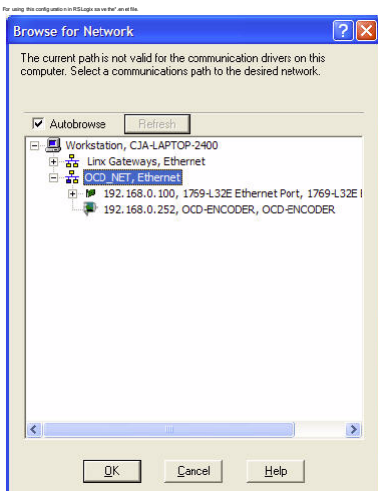




The Wizard finds all EDS files that are discarded in the choosing path and operates a test to check the EDS files on errors. In the next step (see picture 1.3) pictures can be selected for the using nodes. With the button *weiter* the installation can be continued and finished.



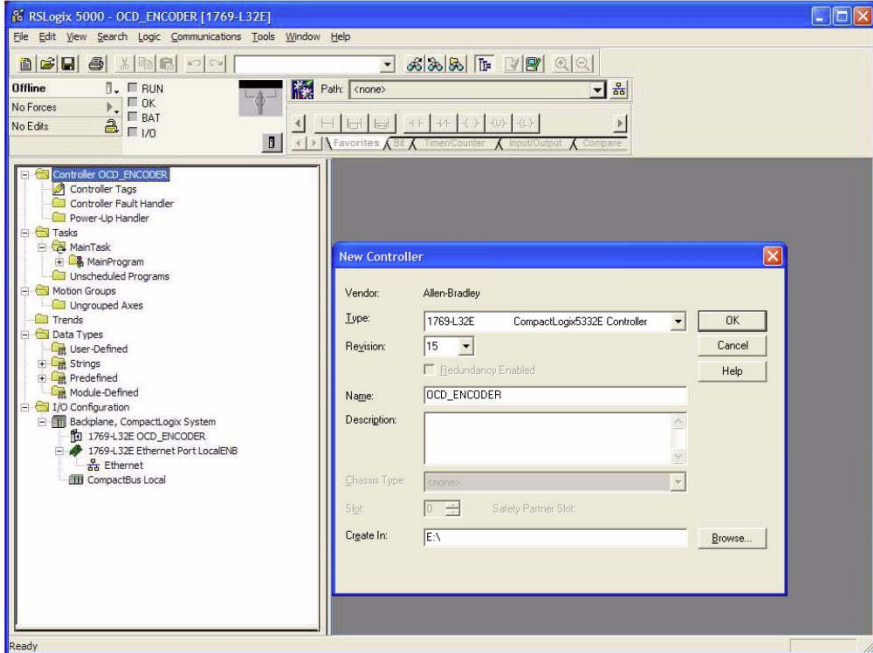
Optional browse the network with all devices with Button  or Upload from Network. So it is not necessary to set the IP-Address manually.



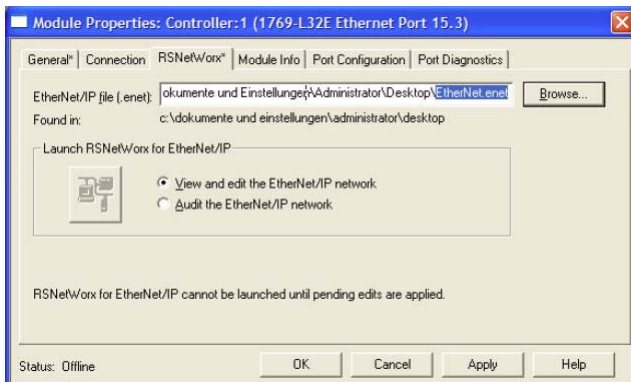
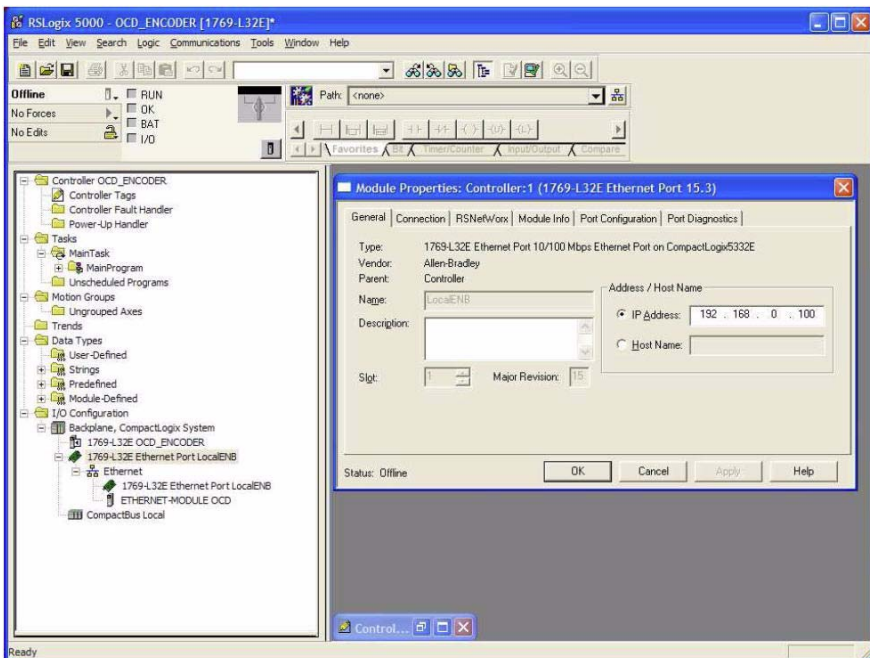
7.1.4 Configuration RSLogix 5000

The RSLogix 5000 Series environment offers an easy-to-use, IEC61131-3 compliant interface, symbolic programming with structures and arrays, and a comprehensive instruction set that serves many types of applications. It supports relay ladder, structured text, function block diagram, and sequential function chart editors for you to develop application programs.

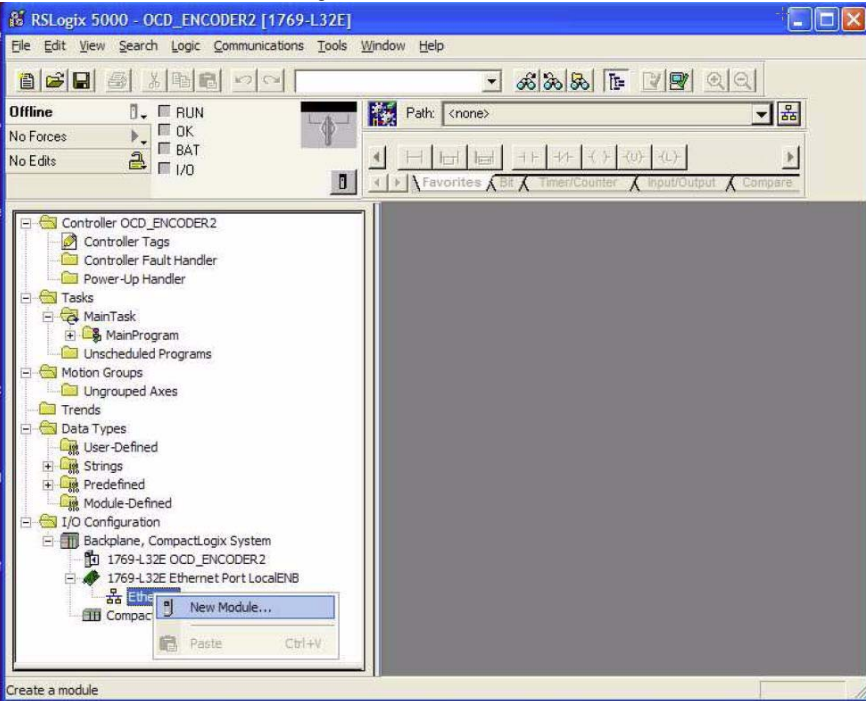
In the first step load a configuration or add a new controller and input a name. In this sample is used the CompactLogix5332E.



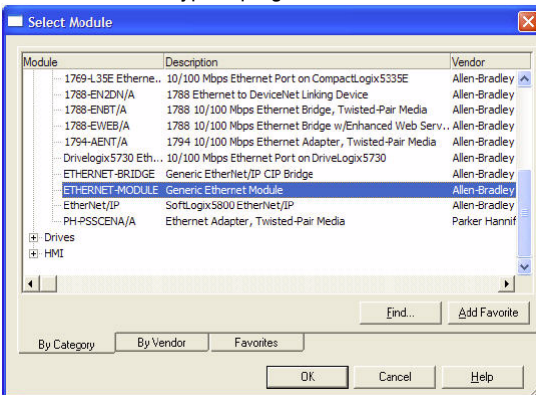
Start the configuration of the controller or load the *.enep file in the module properties of tab RSNetWorx™ that was created with RSNetWorx™.



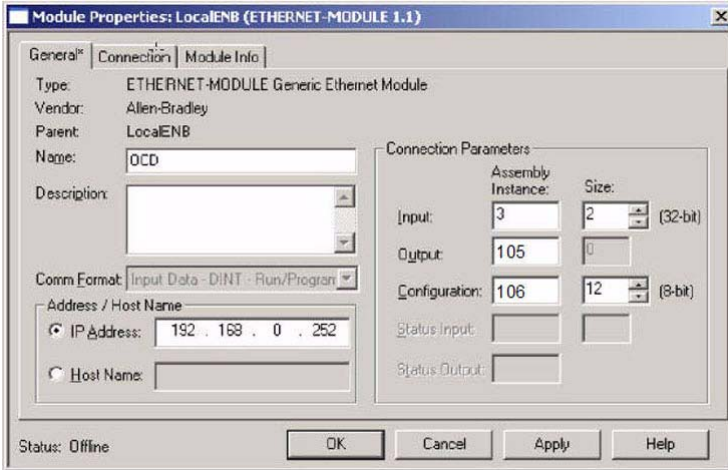
Select the network in the I/O Configuration and add New Module.



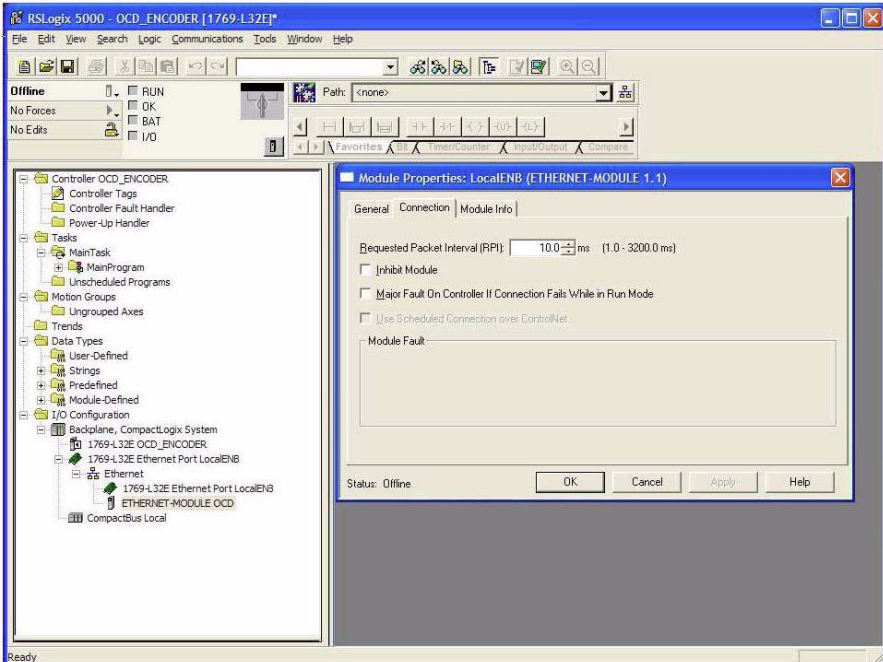
For using a Generic Device select the Generic Ethernet Module. Some PLC's support Encoder Devices too. Please check that the matching EDS file complies to the configuration of the encoder. The device type is programmable.



Set the Connection Parameters according to the following figure.



Set the cycle time.



To read or write data use *Logic - Monitor Tags*

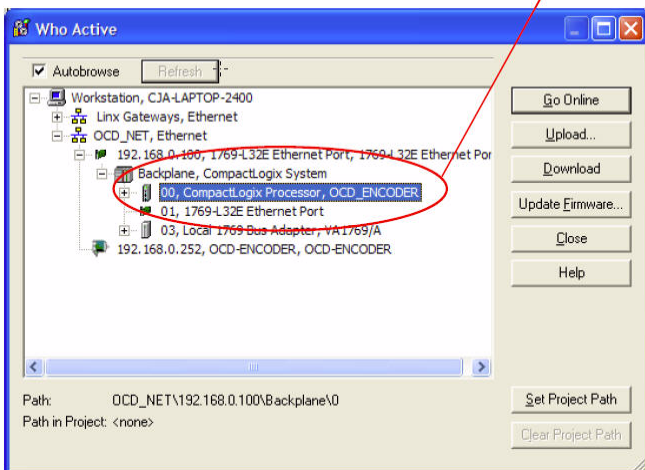
Controller Tags - OGD_ENCODER(controller)

Name	Value	Force Mask	Style	Data Type
- OGD.C	{...}			AB:ETHERNET_...
+ OGD.C.Data	{...}		Hex	SINT(400)
- OGD.I	{...}			AB:ETHERNET_...
- OGD.I.Data	{...}		Decimal	DINT[2]
+ OGD.I.Data[0]	0		Decimal	DINT
+ OGD.I.Data[1]	0		Decimal	DINT

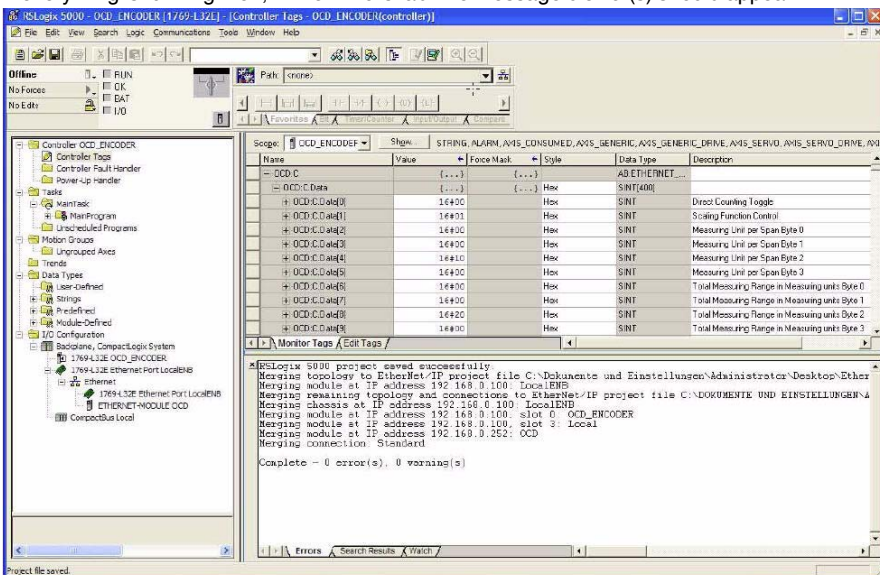
Controller Tags - CompactLogix_132E(controller)

Name	Value	Force Mask	Style	Data Type	Description
- OGD.C.Data	{...}	{...}	Hex	SINT(400)	
+ OGD.C.Data[0]	16#00		Hex	SINT	Direction Counting Toggle
+ OGD.C.Data[1]	16#01		Hex	SINT	Scaling Function Control
+ OGD.C.Data[2]	16#00		Hex	SINT	Measuring Units per Span byte 0 (LSB)
+ OGD.C.Data[3]	16#20		Hex	SINT	Measuring Units per Span byte 1
+ OGD.C.Data[4]	16#00		Hex	SINT	Measuring Units per Span byte 2
+ OGD.C.Data[5]	16#00		Hex	SINT	Measuring Units per Span byte 3 (MSB)
+ OGD.C.Data[6]	16#00		Hex	SINT	Total Measuring byte 0 (LSB)
+ OGD.C.Data[7]	16#20		Hex	SINT	Total Measuring byte 1
+ OGD.C.Data[8]	16#00		Hex	SINT	Total Measuring byte 2
+ OGD.C.Data[9]	16#00		Hex	SINT	Total Measuring byte 3 (MSB)
+ OGD.C.Data[10]	16#04		Hex	SINT	Velocity 0 (LSB)
+ OGD.C.Data[11]	16#1F		Hex	SINT	Velocity 1 (MSB)
+ OGD.C.Data[12]	16#00		Hex	SINT	
+ OGD.C.Data[13]	16#00		Hex	SINT	
+ OGD.C.Data[14]	16#00		Hex	SINT	
+ OGD.C.Data[15]	16#00		Hex	SINT	
+ OGD.C.Data[16]	16#00		Hex	SINT	
+ OGD.C.Data[17]	16#00		Hex	SINT	
+ OGD.C.Data[18]	16#00		Hex	SINT	
+ OGD.C.Data[19]	16#00		Hex	SINT	
+ OGD.C.Data[20]	16#00		Hex	SINT	
+ OGD.C.Data[21]	16#00		Hex	SINT	
+ OGD.C.Data[22]	16#00		Hex	SINT	

- If the value is 00 then the standard configuration will be used
- If the Parameter are out of range the maximum value of the encoder will be used as parameter
- To change parameters open *Communication Who Active*, *Go Offline*, *File Save*, select controller,
- Download, Run
- These parameter can set by a standard EtherNet/IP scanner tool too.

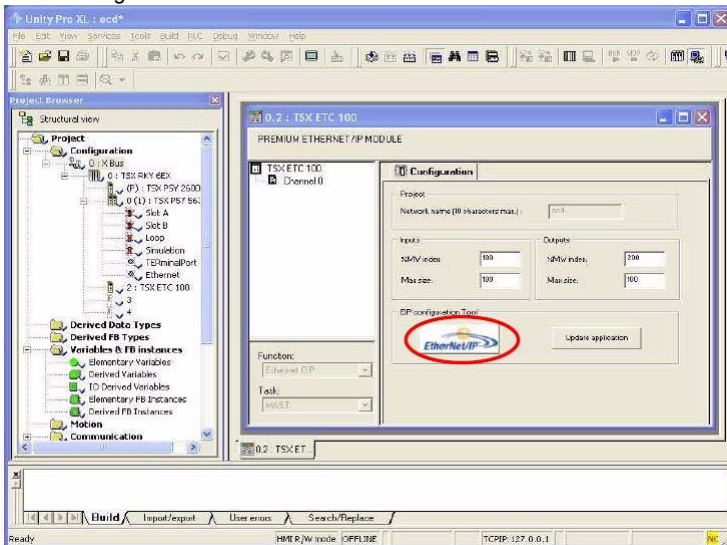


If everything is running then, in the "Errors tab" the message 0 error(s) should appear.

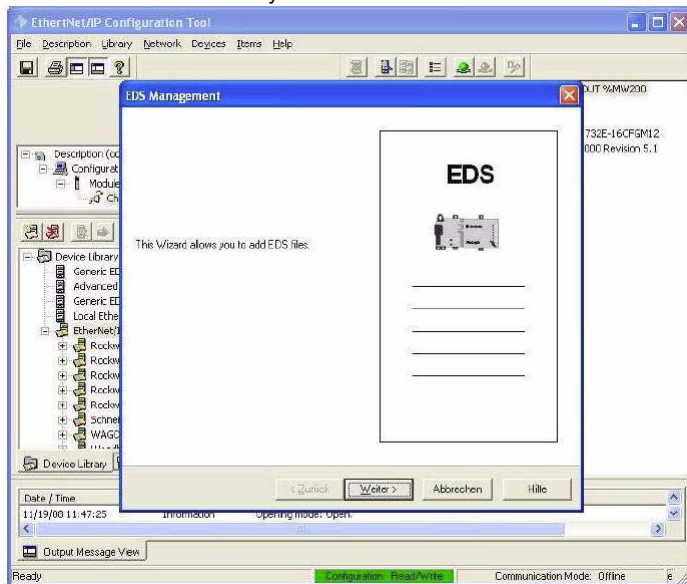


7.2 Schneider configuration tools

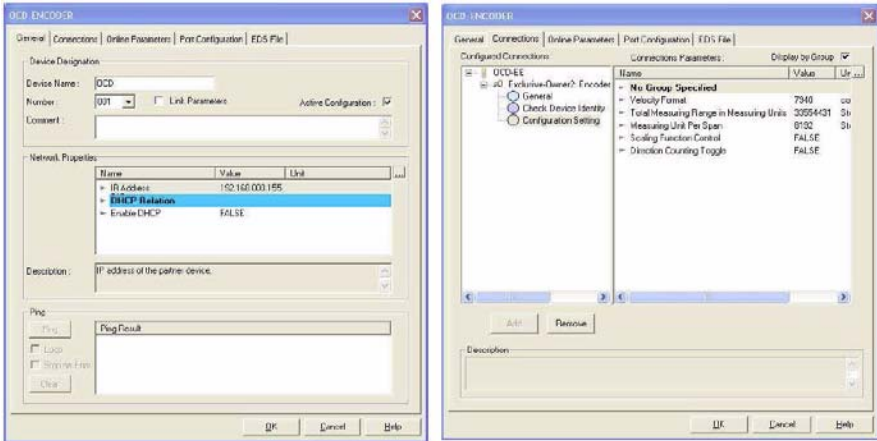
In the software tool Unity it is possible to configure the parameters of the encoders. EDS file help to change the parameters on an easy way. Select the EtherNet/IP module and start the EtherNet/IP configuration tool.



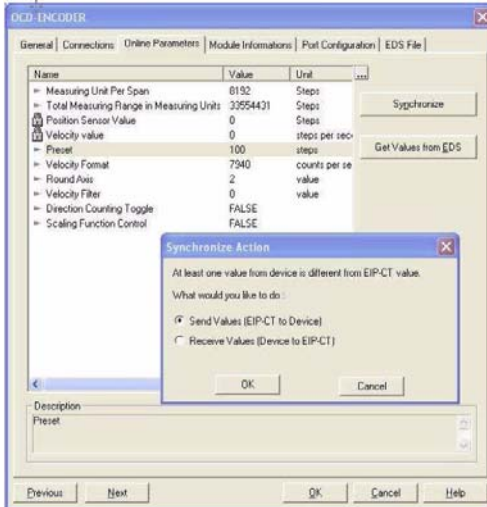
In the first time it is necessary to install the EDSFile with the wizard.



Set the IP-Address of the encoder in the General-Tab. In Tab *Connections under Configuration Setting* is it possible to change the parameters in the offline state. The PLC will send this parameters to the encoder during the start up phase.



In Tab *Online Parameters* can be changed or read out all parameters. There can set i.e. the preset value. Change all parameters, push the Synchronize button and Send Values to the device.



8 FAQ

8.1 Problem:

IP Address unknown and BOOTP/DHCP deactivated

Solution:

Use a Ethernet "sniffer" (i.e. <http://www.wireshark.org>) or

8.2 Problem:

Replace a rotary encoder in the machine and the controller cannot start the application.

Additional the Stat LED is flashing with 2 Hz

Solution:

Start the BOOTP/DHCP server to set the IP-Address. See chapter 7.1

9 Glossar

Term	Explanation
10Base-T	Transmission line with 10 Mbit data transmission rate
100Base-T	Transmission line with 100 Mbit data transmission rate
Baudrate	Transmission rate; it display the transmission bits per second
Big Endian	Variables will use Byte 0 as Low and last Byte as High
Binary	Numeric system with value 0 or 1.
BootP	A UDP network protocol used by a network client to obtain its IP address automatically
CAT5	Terminations for transmission rates up to 100 Mbit
QP	C ontrol and I nformation P rotocol
DHCP	D ynamic H ost C onfiguration P rotocol is a protocol used by networked devices (clients) to obtain the parameters necessary for operation in an Internet Protocol network. This protocol reduces system administration workload, allowing devices to be added to the network with little or no manual configuration.
EIP	E therNet/ I P
EMC	E lectromagnetic C ompatibility, there are rules to verifying devices.
ENIP	E therNet/ I P
Ethernet	Ethernet is a computer network technology based on frames.
Explicit Messages	Communication between i.e. a Ethernet scanner and encoder
Fast Ethernet	Transmission technology with 100 Mbit transmission rate
Flash	Internal memory, saved data will be available after power down
Implicit Messaging	I O Connection: communication between controller and device
IP-Address	Allow a logic addressing from computer in a network
IP-Protocol	The I nternet P rotocol is widespread in computer networks. It is the implementation of the internet layer of the TCP/IP-model
MAC Address	World wide explicit address of a device. The encoder use three MAC Addresses: one for internal interface and two for the ports
Mbit	Transmission rate or baud rate, million bits per second
OCD	Acronym: O PTO C ODE, name of an encoder series manufactured by FRABA POSITAL
OSI-Model	The O pen S ystem I nterconnection reference model is a open layer model for the organization of a communication.
Scanner	Program to send Explicit Messages to the encoder
Switch	A switch is an electronic device to connect computers e.g. network segments in a local network. Unlike a hub, a switch uses stacks to avoid network collisions
TCP	The T ransmission C ontrol P rotocol is a connection orientated transmission protocol, in a network.
UDP	U ser D atagram P rotocol is utilized to send data that does not need to be transferred in a reliable way.



FACTORY AUTOMATION – SENSING YOUR NEEDS



Worldwide Headquarters

Pepperl+Fuchs GmbH
68307 Mannheim · Germany
Tel. +49 621 776-0
E-mail: info@de.pepperl-fuchs.com

USA Headquarters

Pepperl+Fuchs Inc.
Twinsburg, Ohio 44087 · USA
Tel. +1 330 4253555
E-mail: sales@us.pepperl-fuchs.com

Asia Pacific Headquarters

Pepperl+Fuchs Pte Ltd.
Company Registration No. 199003130E
Singapore 139942
Tel. +65 67799091
E-mail: sales@sg.pepperl-fuchs.com

www.pepperl-fuchs.com

 **PEPPERL+FUCHS**
SENSING YOUR NEEDS

Subject to modifications
Copyright PEPPERL+FUCHS • Printed in Germany

TDOCT-2115A_ENG

xxxxxx
01/2012