POSITAL

EtherNet/IP™
conformance tested

ABSOLUTE IXARC ROTARY ENCODER WITH ETHERNET/IP INTERFACE
USER MANUAL

CONTENTS

Imprint
POSITAL GmbH
Carlswerkstrasse 13c
D-51063 Köln
Telefon +49 (0) 221 96213-0
Telefax +49 (0) 221 96213-20
Internet http://www.posital.com
E-mail info@posital.eu

Alteration of Specifications reserved
Technical specifications, which are described in this manual, are subject to change due to our permanent strive to improve our products.

Document information
File name: UME-OCD-EE_V1R10.doc
Date: Version 25.7.2019
Version number: 1.13
Author: RBA

Phone Service
For technical support, questions and suggestions for improving our products and documentations call our telephone line +49 (0) 221-96213-0.

POSITAL

ABSOLUTE IXARC ROTARY ENCODER WITH ETHERNET/IP INTERFACE
USER MANUAL

CONTENTS

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 16,384 revolutions (14 Bit). Therefore the largest resulting resolution is 30 Bit = 1,073,741,824 steps. The standard Single-Turn version is 13-bit, the standard Multi-Turn version 25-bit.

The integrated Ethernet interface of the absolute rotary encoder supports all necessary EtherNet/IP functions.
The protocol supports the programming of the following additional functions in several ways:

1. Code sequence (Complement)
2. Resolution per revolution
3. Total resolution
4. Preset value
5. IP-Address

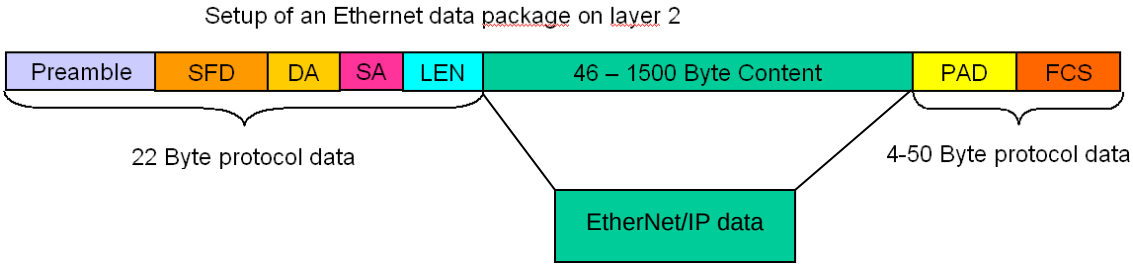
The general use of absolute rotary encoders with EtherNet/IP interface is guaranteed. The data will transmit in a standard Ethernet frame in the data section, see at the bottom of this side the pink field with the blue frame.

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

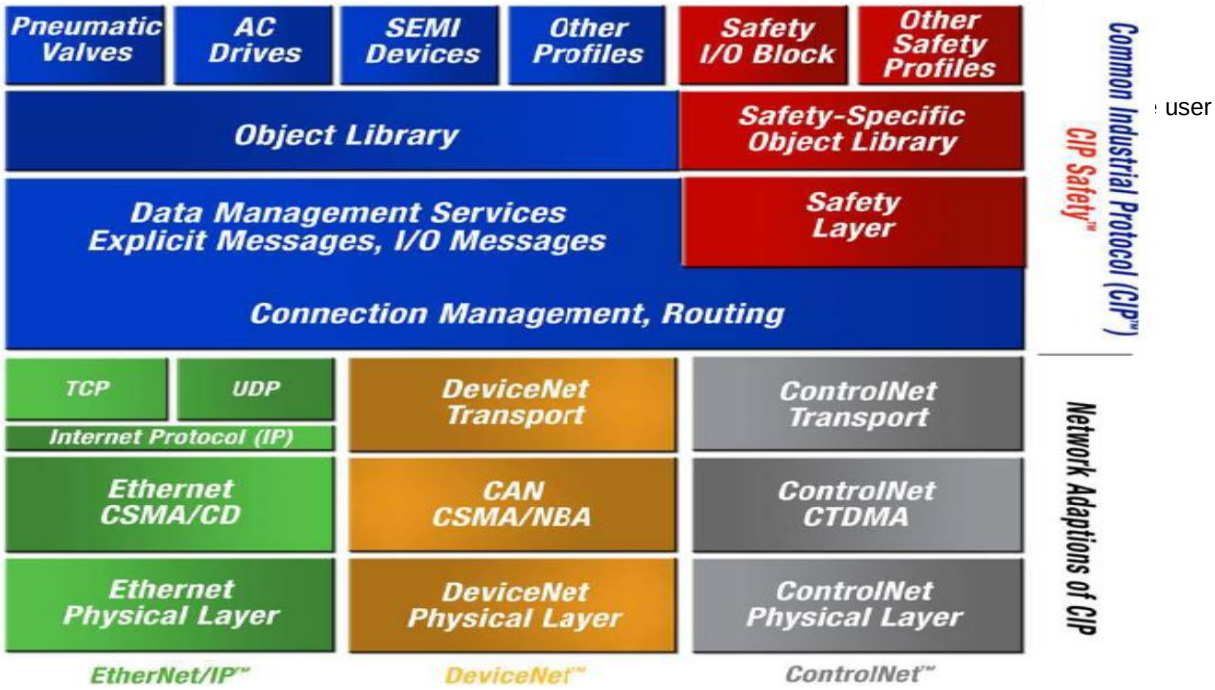
The IP address can be programmed with DHCP or BOOTP via the configuration tools of the PLC.

The physical interface support Autonegotiation and Autocrossing.

General information's about EtherNet/IP are available:
www.ethernetip.de (German)
www.odva.org/default.aspx?tabid=67 (English)



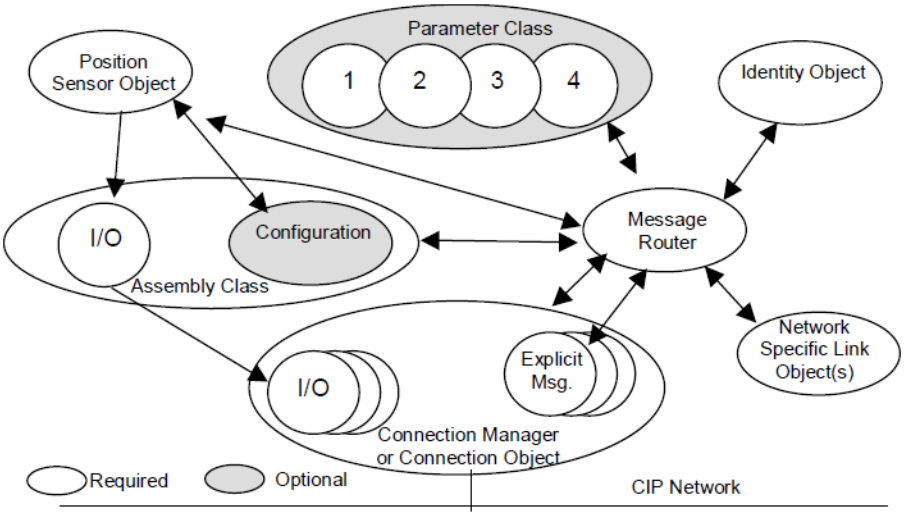
1.1 Control and Information Protocol (CIP)



user

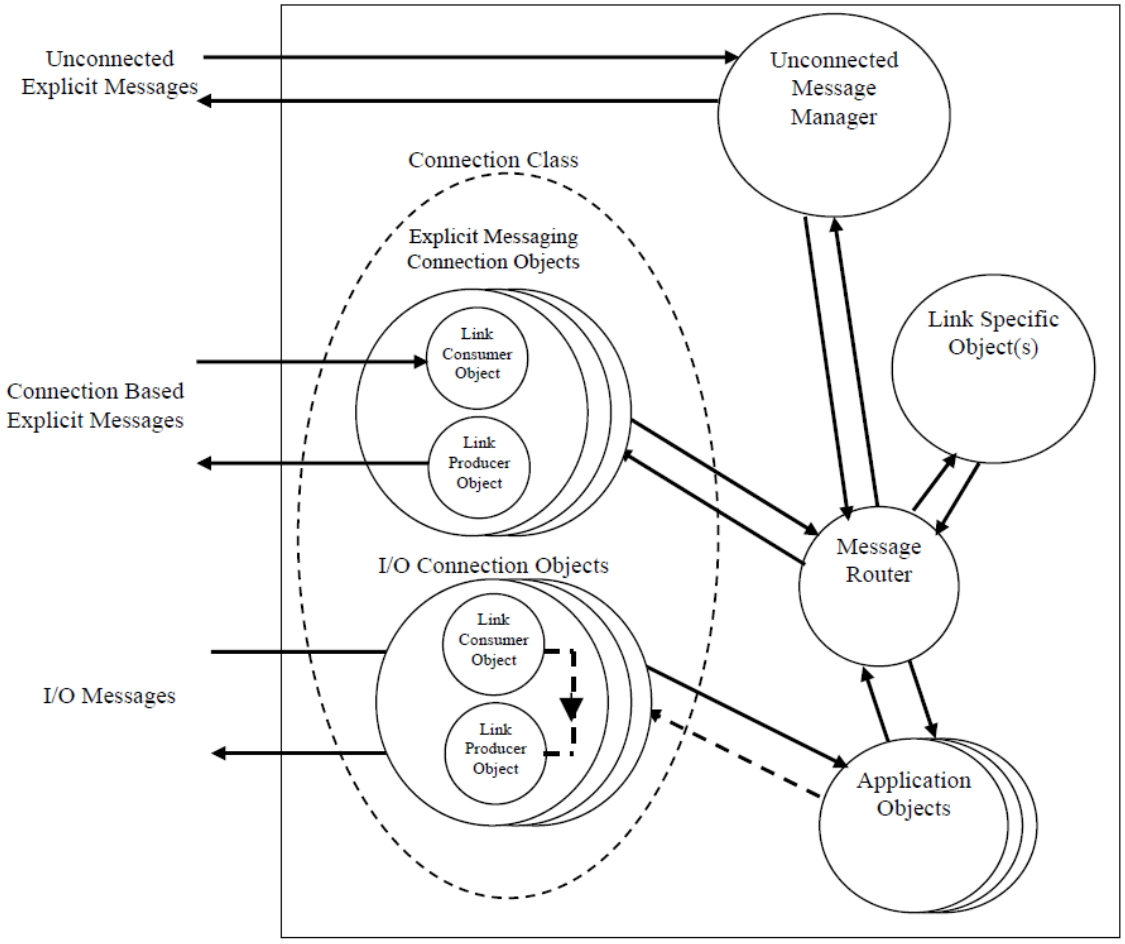
1.2 Object model

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. An 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 its defined behaviors. The absolute rotary encoder supports 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.



2. Data Transmission

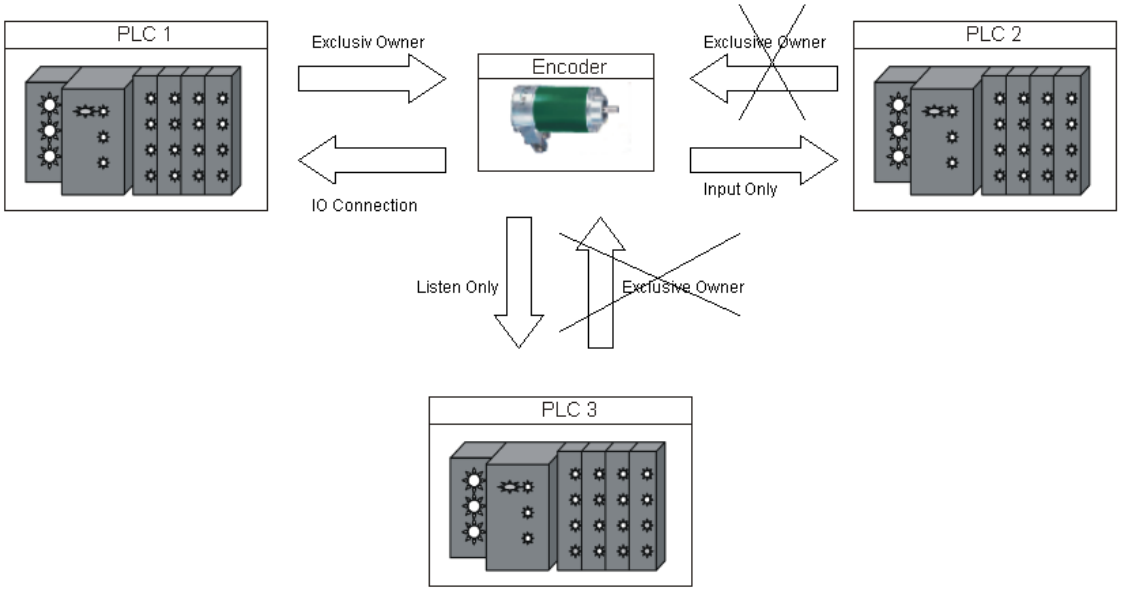
The data transmission within the EtherNet/IP network is realized by implicit or explicit messaging. Explicit messages are split in unconnected and connection-based versions. Unconnected messages will be used by EtherNet/IP scanners.



Exclusive-Owner, Input Only Listen Only

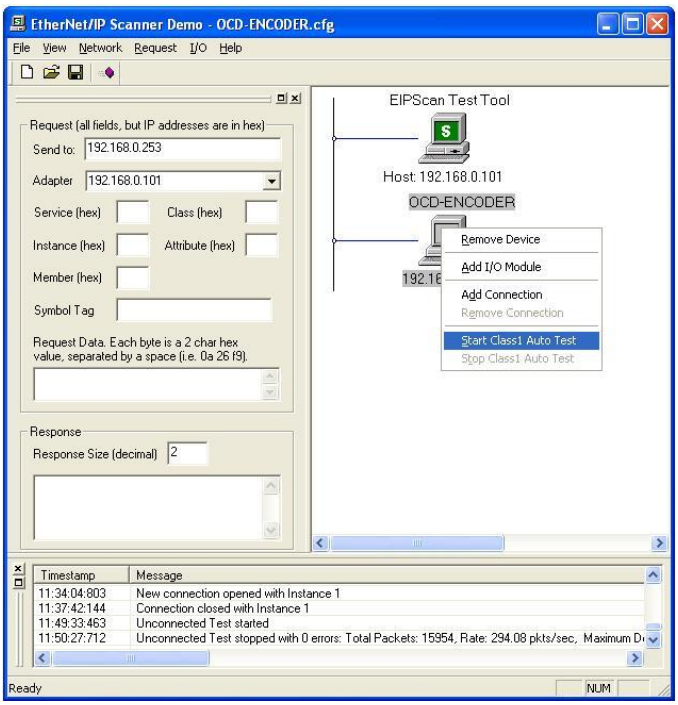
It is possible to open 256 connections with the encoder. One could be an Exclusive Owner connection, 255 additional connections can be realized both as Input Only or Listen Only.
With an Exclusive Owner, connection parameters can be transmitted (cycle time, configuration and Assembly Instances) to the encoder.

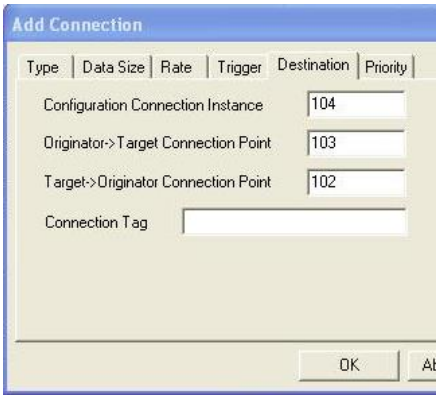
Input Only connections can only work if all of the parameters are set in the PLC accordingly to the encoder parameters.
Listen Only need a connection such as Excusive Owner or Input Only.

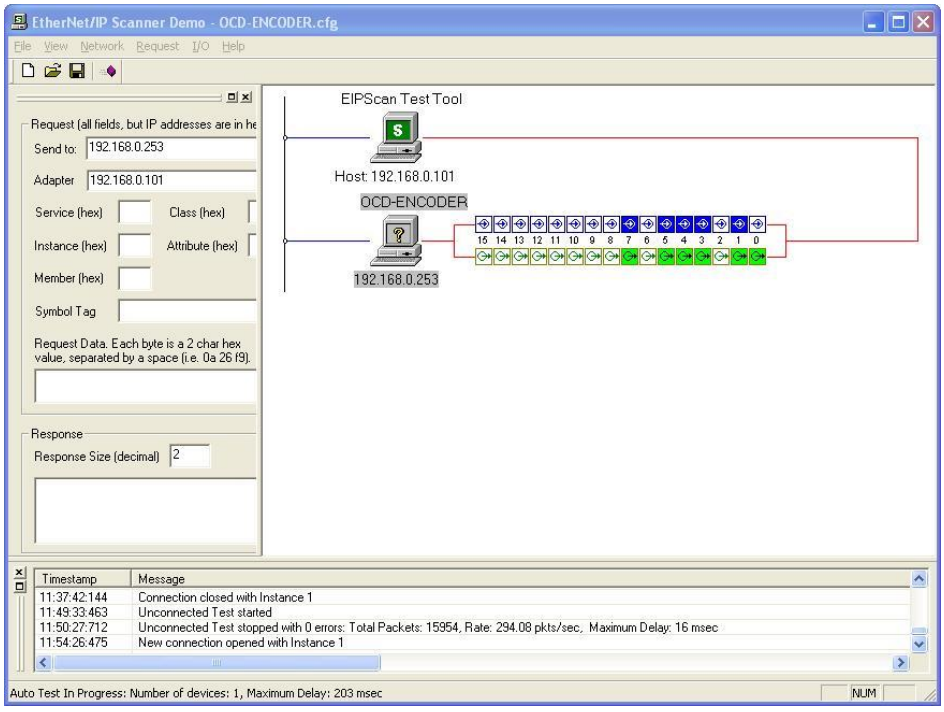


Assembly	Config	Output Instance	Input Instance
Connection Manager	Config	Connection Point 1	Connection Point 2
Exclusiv-Owner	$0x6A_{hex}$ (106)	$0x69_{hex}$ (105)	0x01 Position value
			0x03 Position value + velocity
Input Only	$0x6A_{hex}$ (106)	$0x64_{hex}$ (100)	0x01 Position value
			0x03 Position value + velocity
Listen Only	-	$0x65_{hex}$ (101)	0x01 Position value
			0x03 Position value + velocity
Demo-Scanner	$0x68_{hex}$ (104)	$0x67_{hex}$ (103)	$0x66_{hex}$ (102)

Communication check







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

2.1.2 Data Mapping

Data Component Name	Class		Instance Number	Attribute	
	Name	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 Number	Attribute	
	Name	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 be sent during power up to the encoder.

For some tools it is necessary to use the connection path as parameter:

[20] [04] [24 6A] [2C 69] [2C 01] [80 06 00 01 00100000 00200000 041F]

Segment Groups	Segment	Description
Application Path	20 04	Assembly object class
	24 6A	Instance segment type with Assembly Instance $0x6A_{hex}$ (105) (Configuration)
	2C 69	Assembly Instance $0x69_{hex}$ (106) (Output controller to encoder)
	2C 01	I/O Assembly Instance 1 (Position value)
	80 06	Data segment with lenght 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 for Position sensor object (Class 0x23$_{hex}$)

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. Use instance 0 of position sensor class to restore all configuration parameters at once. To restore a single parameter, use instance 1 of position sensor class with attribute number as argument (see next table).
16$_{hex}$	Save	Save the parameters from chapter 2.1.3 in the nonvolatile memory. Use instance 0 of position sensor class to save all configuration parameters at once.

2.2.2 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 for example 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) indicates that the device is capable of obtaining its network configuration via BOOTP.
1	DNS Client	Not supported
2	DHCP Client	1 (TRUE) indicates that the device is capable of obtaining its network configuration via DHCP.
3	DHCP-DNS Update	Not supported
4	Configuration Settable	1 (TRUE) indicates that 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 communication interface (e.g., an 802.3 interface). There are two components to the attribute: a Path Size (in UINTs) and a Path. The Path contains a Logical Segment type Class, and a Logical Segment type Instance that identifies the physical link object. The maximum Path Size is 6 (in the case of 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 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 IP address of the device. Value of 0 indicates that 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 network mask of the device. 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 is 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 is 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

2.4.0 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-XX-XX-XX-XX"
6	Set	Interface Control	STRUCT of:	Configuration for physical interface	
		Control Bits	WORD	Interface Control Bits	See table below
		Forced Interface Speed	UINT	Speed at which the interface shall be forced to operate	Speed in Mbps (10 or 100)
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"

Control Bits

Bit(s)	Called	Definition
0	Auto-negotiate	802.3 link Auto-negotiation: 0 = disabled, 1 = enabled (standard) If Auto-negotiation is disabled then the device shall use the settings indicated by the Forced Duplex Mode and Forced Interface Speed bits.
1	Forced Duplex Mode	If Auto-negotiation bit = 0 the Forced Duplex Mode bit indicates whether the interface shall operate in full or half duplex mode. 0 = Half Duplex, 1 = Full Duplex
2-15	Reserved	Shall be set to zero

Example

Use on Transmit data size double (4 bytes) 00000064
for Auto-negotiation = disable on 100 MBaud

2.4.1 Interface Flags

Bit(s)	Called	Definition
0	Link Status	Indicates whether 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 specific to the implementation. In some case, 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 if it receives incoming packets.
1	Half/Full Duplex	Indicates whether the duplex mode is 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 not determinate.
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 that the interface can perform changes to link parameters (auto-negotiate, duplex mode, interface speed) automatically. 1 indicates that the device requires a Reset service to be issued to its Identity Object for the changes to be effective.
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. For example, an AUI/MII interface can detect that no transceiver is plugged, or a radio modem can detect that no antennae plugged. In contrast to the soft, possible self-correcting nature of the Link Status being inactive, this concerns major issues requiring the intervention of the user.
7	Reserved	Shall be set to zero

2.4.2 Common Services

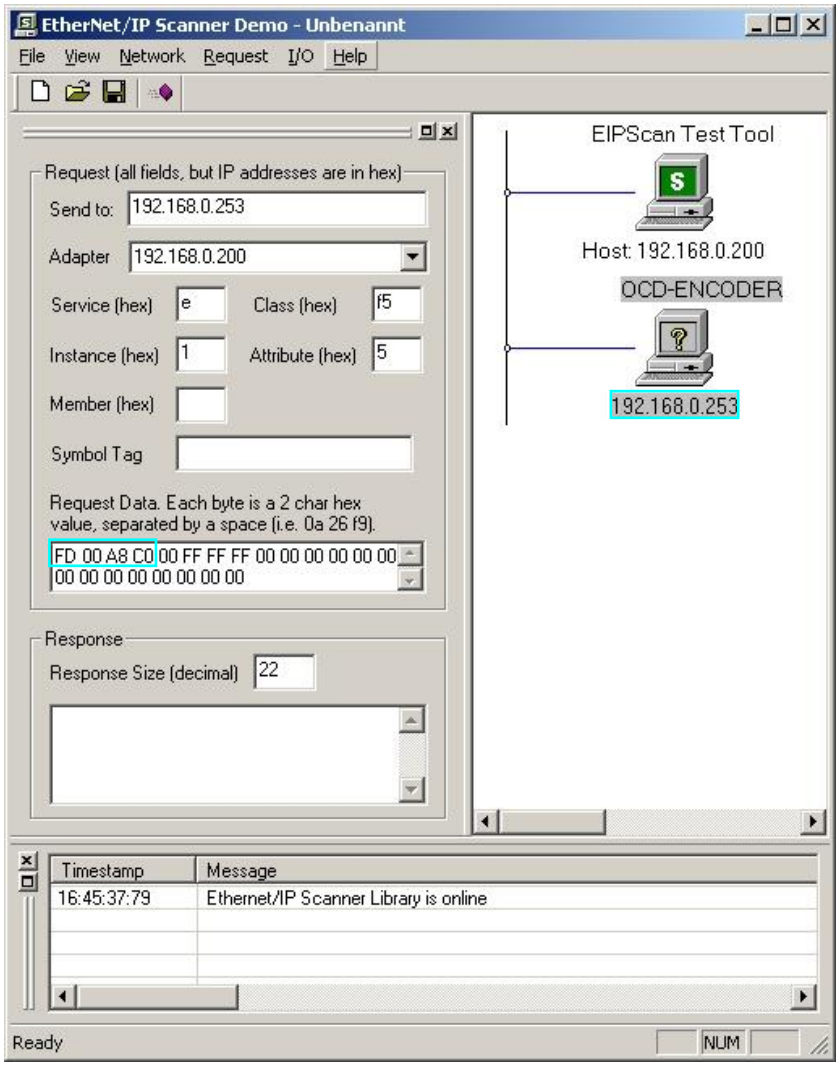
Service Code	Class	Instance*	Service Name	Description of Service
$0E_{hex}$	Conditional	Required	Get_Attribute_Single	Returns the contents of the specified attribute
10_{hex}	n/a	Conditional	Set_Attribute_Single	Modifies a single attribute

2.4.3 Link Object Instances

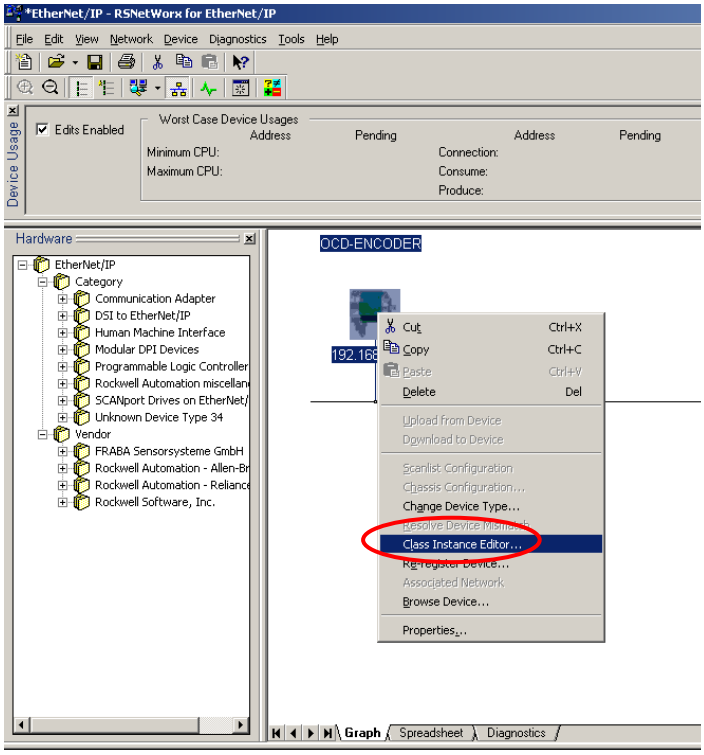
Instance	Description
1	Internal interface
2	Intern switch Port 1
3	Intern switch Port 2

2.5 Setting parameters with scanners

There are several external scanners for EtherNet/IP available. RS-NetWorks™ has one such scanner. In the figure you can see an example where the IP-Address (FD 00 A8 C0 complies 192.168.0.253), the Subnet (00 FF FF FF complains 255.255.255.0), Gateway (00 00 00 00), DNS1 (00 00 00 00), DNS2 (00 00 00 00) and Domain Name = "" (ASCII Character max length = 48 bytes) are read out of the encoder.

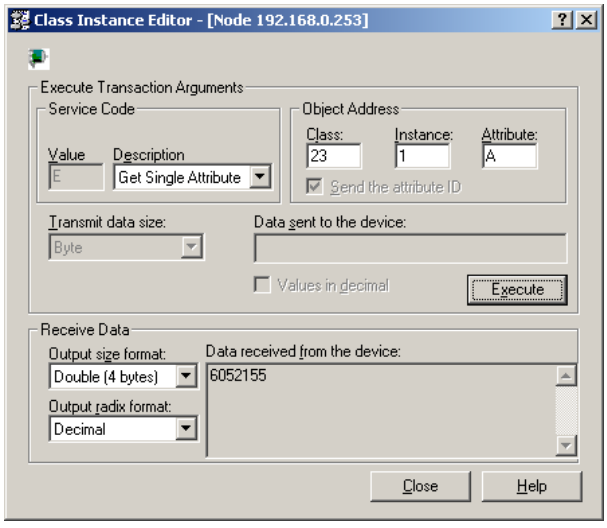


In RSNetWorx, a scanner is available too. In the next section you can see an example of how to set the Preset value.



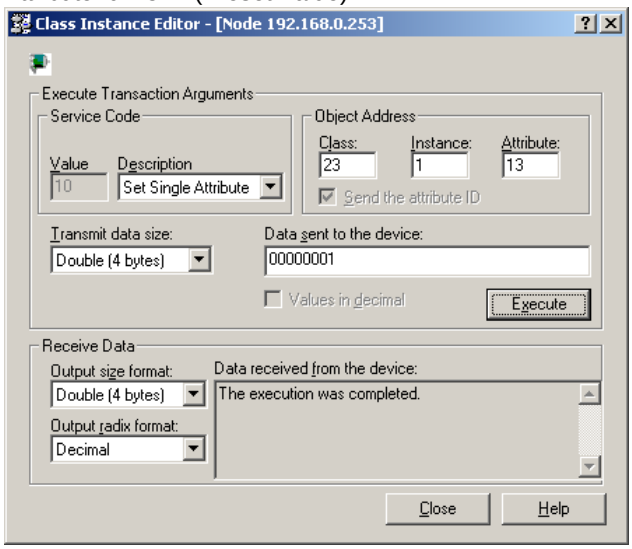
2.5.1 Read out position value

Get Single Attribute Position sensor value:
Class: 0x23 (Position sensor object)
Instance: 0x01
Attribute: 0x0A (Position Value)



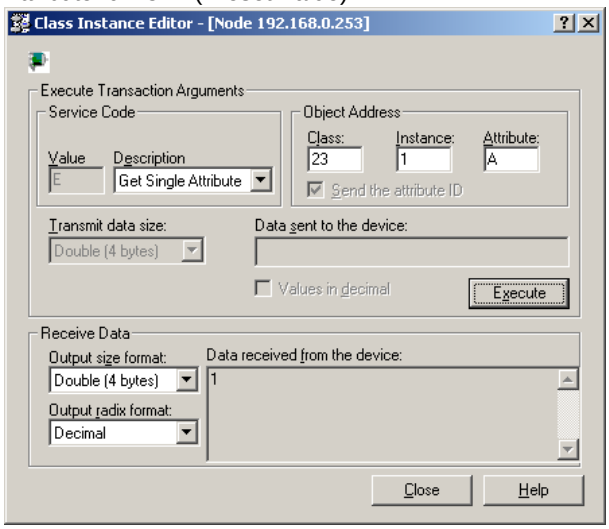
2.5.2 Set preset value

Set Single Attribute Position Preset Value to 1
Class: 0x23 (Position sensor object)
Instance: 0x01
Attribute: 0x13 (Preset Value)



2.5.3 Get preset value

Get Single Attribute Position Value
Class: 0x23 (Position sensor object)
Instance: 0x01
Attribute: 0x13 (Preset Value)



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		

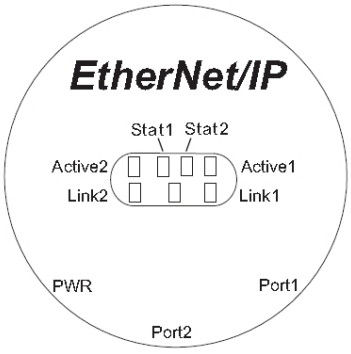
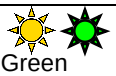


Table 1: Module Status Indicator Stat1/Stat2

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 (e.g. 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 e.g. an incorrect or inconsistent configuration, the module will be flashing red
Steady On Red	Major fault	Happens when 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

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 the target of one or more of the connections in which this device is 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 the 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 23hex

4.1.1 Direction counting

This operating parameter can be used to select the code sequence. The parameter can be set via 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 from the shaft (clockwise or counterclockwise). The counting direction is defined in the attribute 0Chex:

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

Bit 0	Scaling function on/off
0	on
1	off

4.1.2 Scaling function control

This parameter enables or disables the scaling function i.e. the possibility to modify the resolution.

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

This parameter can be set via Configuration Assembly and Explicit Messaging

4.1.3 Resolution per revolution

Use case: measure a precise number of steps over a defined number of turns. The ratio between them gives:

Res/Rev = X_{steps} / X_{turns}

Res/Rev = Resolution per Revolution – *also called Measuring steps per revolution.*
X_{steps} = Number of steps
X_{turns} = Number of Turns

Examples:

Use case 1	3600 steps over 10 turns	Res/Rev_1 = 360
Use case 2	1000 steps over 3 turns	Res/Rev_2 = 333.33

Rules:

1. To be able to set the resolution as needed, the **Scaling function control must be activated**! If the Scaling function control is deactivated, then the output value is the physical resolution.
2. Posital's Ethernet/IP encoders only accept integers as resolution per revolution. In the examples above, only the Res/Rev$_1$ of Use case 1 can be set up.
3. Each value between 1 and the maximum (see type label) can be realized.
4. If the resolution is set to a value larger than the physical resolution (e.g. 8,192 for a 13-bit encoder), it will lead to values being skipped while rotating the shaft. Therefore, it is recommended, to keep the resolution per revolution below the physical resolution (8,192 steps in this example).

The parameter can be set via Configuration Assembly and Explicit Messaging.

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

(*) see type label
Example of maximum resolution:
16Bit Encoder: $10{,}000_{hex}$ (65,536)

When the **Resolution per revolution** is changed, the **Total resolution** must be changed accordingly (see next paragraph).

4.1.4 Total resolution

This value represents the number of steps over the total measuring range. This value must not exceed the total resolution of the encoder written on the type label (e.g. for a 25-bit encoder; 33,554,432 steps).

If the **Resolution per revolution** has been changed (see 4.1.3), the **Total resolution** must be adapted accordingly to the equation:

Res_{tot} = $Xtot_{turns}$ x Res/Rev

Res_{tot} = Total resolution
Res/Rev = Resolution per revolution
$Xtot_{turns}$ = Total number of turns

The total number of turns is a fixed number thus cannot be modified. Example: for 13-bit revolution encoder it is 8,192 turns.

Scaling function control **must be switched on**!

The parameter can be set via 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 label
Example of maximum total resolution
30-bit encoder: $40,000,000_{hex}$ (1,073,741,824)

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 value by the preset parameter.

The preset value must not exceed the total measuring range. The parameter can be set via 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 per second.
This parameter can be set via Configuration Assembly or 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
		$1F08_{hex}$	Steps per hour
		$1F09_{hex}$	Steps per day
		$1F0E_{hex}$	RPS (revolutions per second)
		$1F0F_{hex}$	RPM (revolutions per minute)

4.1.7 Velocity Filter

To increase the accuracy of the velocity in the application, it is possible to switch between three different filter types of the exponential moving average. Default: Fine.

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

Ratio between old and current speed:
Fine: 7:3, Normal: 96:4, Rough: 996:4

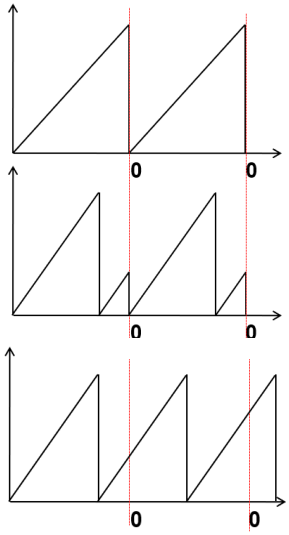
4.1.8 Endless Shaft

It is important that the ratio “Total resolution” / “Measuring units per revolution” remains a multiple of the total number of turns. For example, for a 12-bit revolutions encoder, the ratio must remain a multiple of 4,096: **Res_{tot} / Res/Rev = n x 4,096**

Res_{tot} = Total resolution
Res/Rev = Measuring units per revolution
n is an integer

However, the Endless Shaft functionality of POSITAL’s EtherNet/IP encoders solves the problem directly. It will check whether n is an integer (see equation here above).

If not, it will run calculous to make sure we do not see any drops in the position (see pictures here after).
Endless Shaft Functionality must be enabled.
Default value: Auto.
Can be set only via Explicit Messaging.



Endless Operation

Revolutions = 256

Revolutions = 200
Without

Endless Functionality

Revolutions = 200

With

Endless Functionality

Note: The internal software routine only works if the encoder is in operation. If it is necessary to turn the encoder shaft more than 1,024 revolutions without power supply this can lead to problems (the internal routine will not work without power supply). Therefore, the rule ahead should be observed for 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 communication.

The encoder uses a second D-coded connector and provides integrated switch functionality.

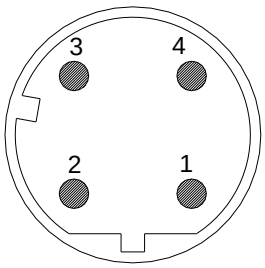
Connector Ethernet
4 pin female, D-coded

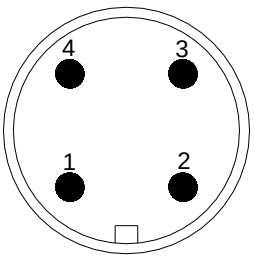
Pin Number	Signal
1	Tx +
2	Rx +
3	Tx -
4	Rx -

Connector power supply
4 pin male, A-coded

Pin Number	Signal
1	US (10 - 30 V DC)
2	N.C.
3	GND (0V)
4	N.C.

Sketch as seen on the encoder





5.2 Ethernet cables

5.2.1 RJ45 – M12 crossed

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

5.2.2 RJ45 – M12 straight

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

5.2.3 M12 – M12 straight

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

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

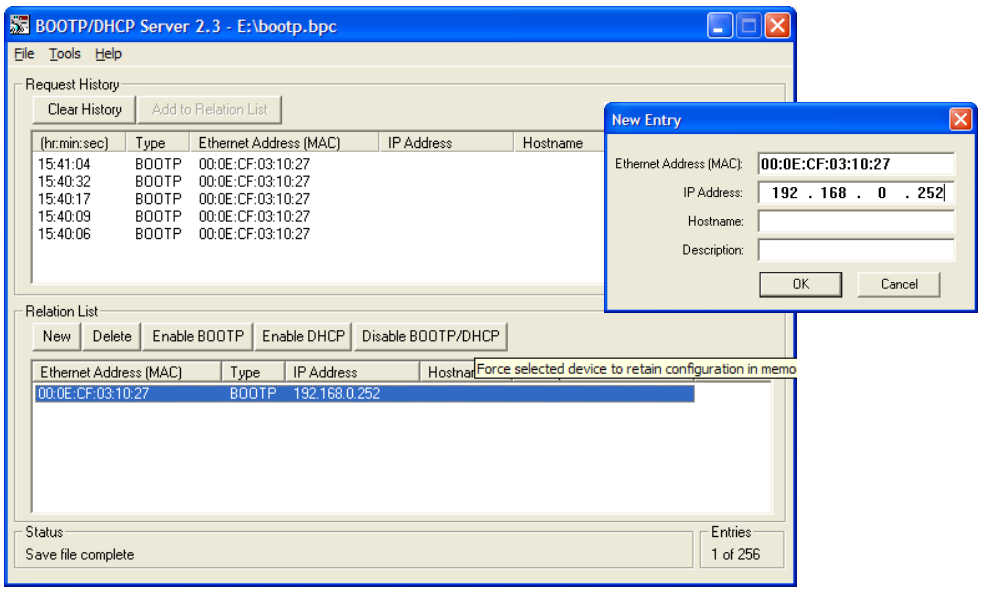
7.1.1 Setting IP-Address (BOOTP/DHCP)

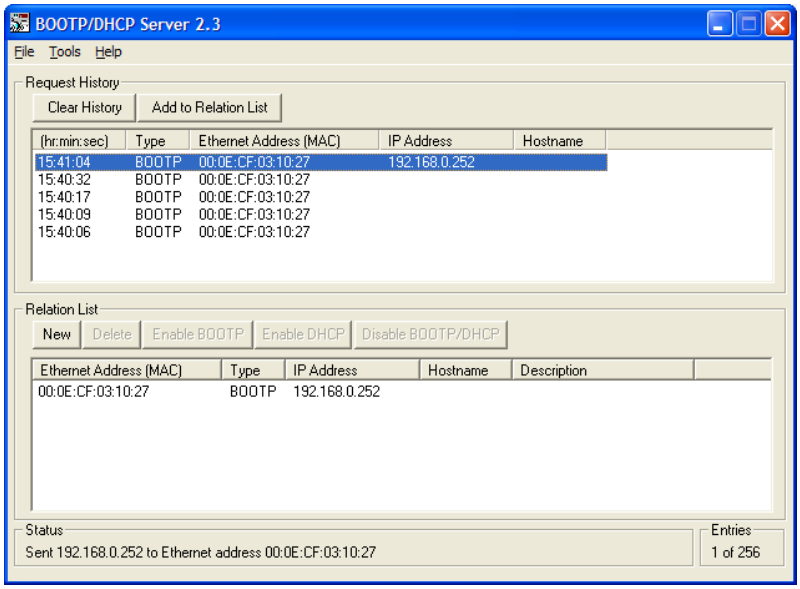
To set the IP Address there are special tools available. I.e. the BOOTP/DHCP Server is installed with the software package from RSNetWorx™. The server scans the network for the MAC Addresses of all products with active BOOTP or DHCP. If one MAC address is selected in the Request History then the IP Address 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 again. But after several times with no answer the frequency of requests decreases. A power up after a long pause could solve the missing requests.

If you cannot find all the encoders within the BOOTP/DHCP Server list then check the following points:

- LED status of the encoder OK?
- Is the Network correctly set?
- Are the BOOTP and/or DHCP enabled?

If the encoder already has an IP-Address, the BOOTP and DHCP must be disabled. Otherwise the encoder will ask for a new IP-Address again. After setting the IP-Address the Status LED is flashing with 1 Hz. 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 Address and IP-Address are available and BOOTP or DHCP can be deactivated 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





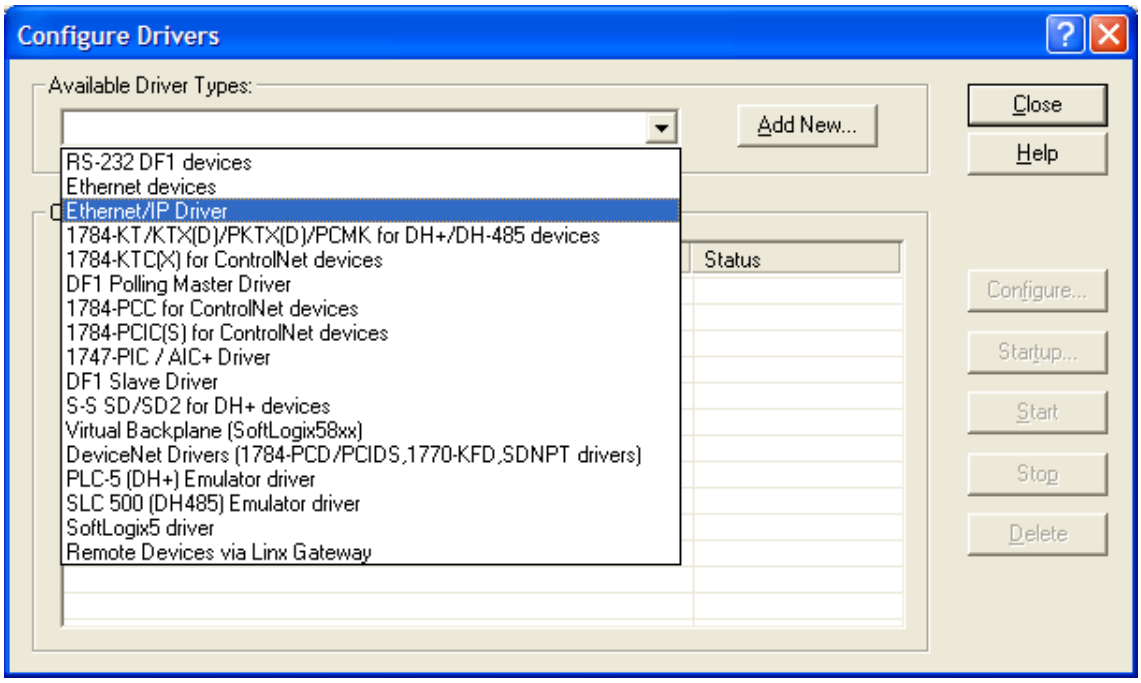
After setting the IP-Address with this tool the IP-Address will be available only after the next BOOTP request.

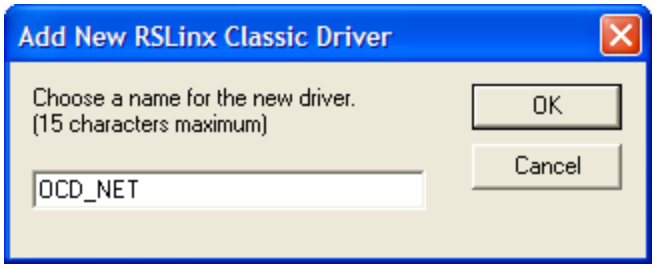
If the IP-Address is not known and BOOTP and DHCP are deactivated it is possible with a special tool to find the IP-Address or to activate BOOTP or DHCP. See details in chapter 7.3.

7.1.2 Configuration RSLinx Classic™

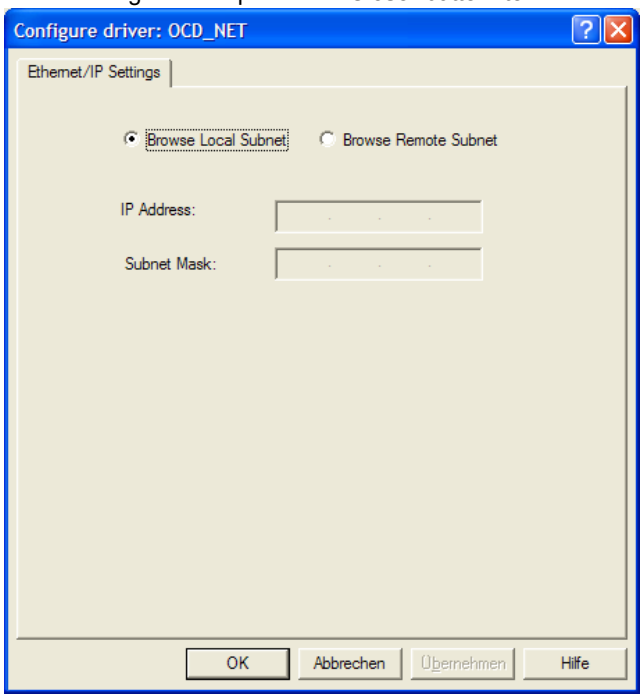
RSLinx™ is a complete communication server providing plant-floor device connectivity for a wide variety of Rockwell Software applications such as RSLogix™, RSNetWorx™,…

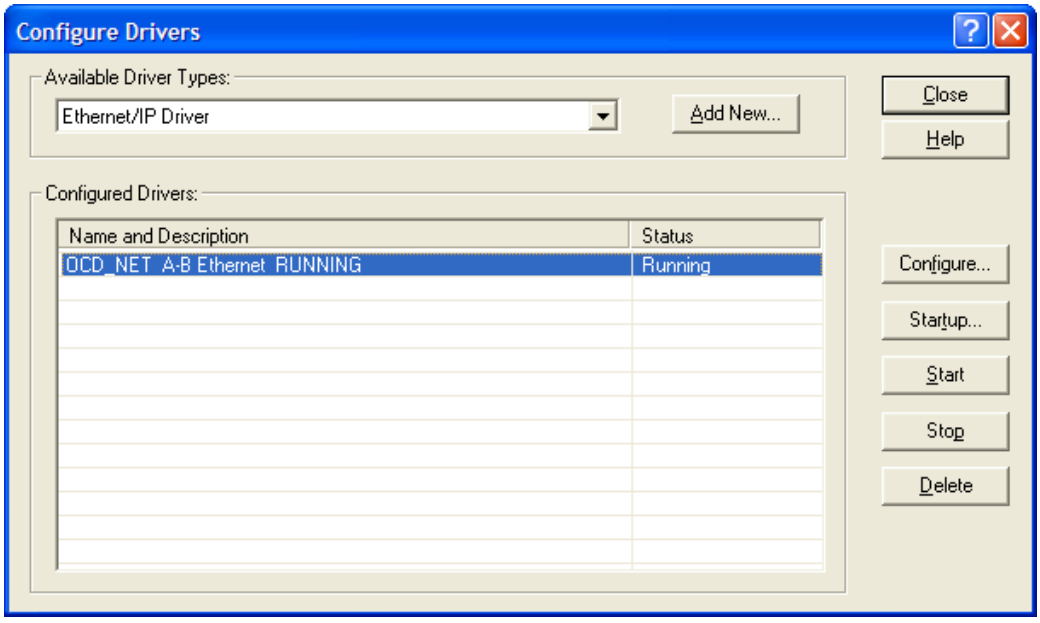
To start a new project add first a new RSLinx Classic™ Driver for EtherNet/IP under *Communications Configuration Drivers* and input the name.





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.





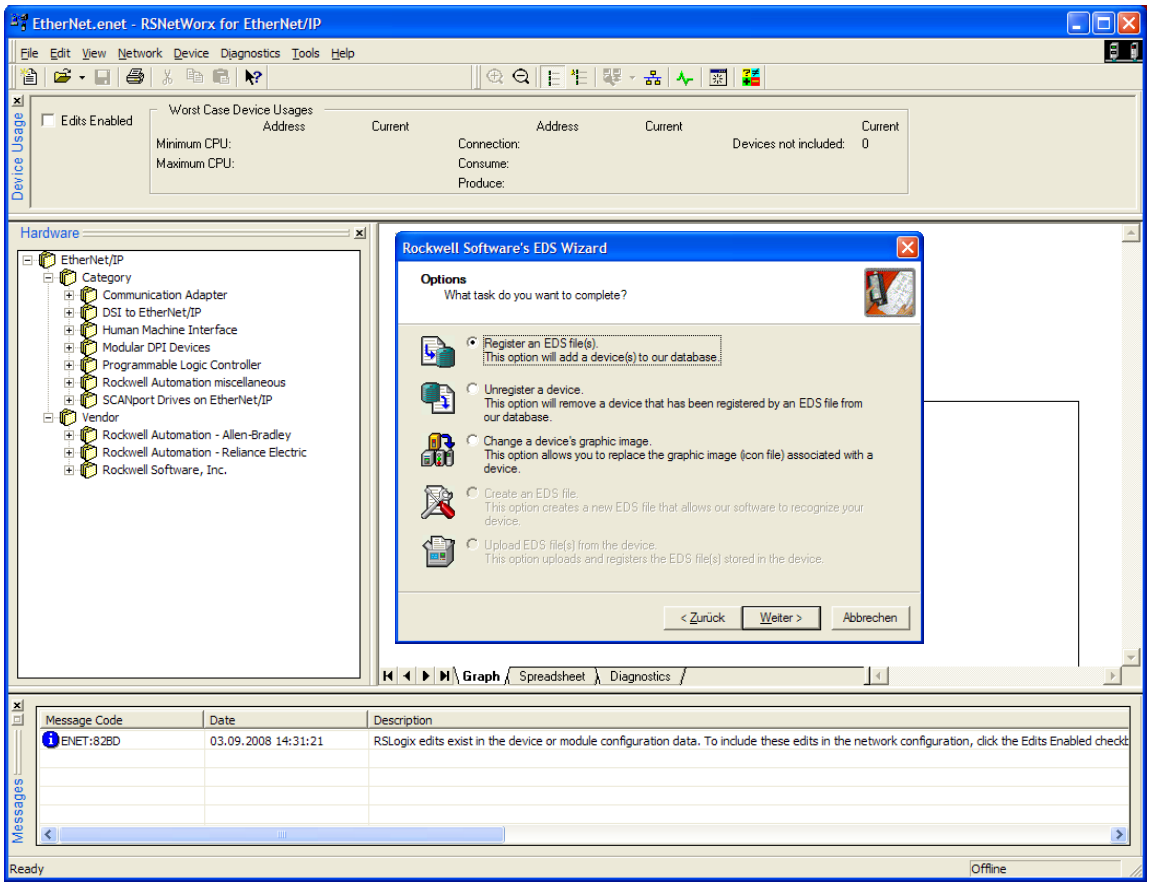
7.1.3 RSNetWorx™

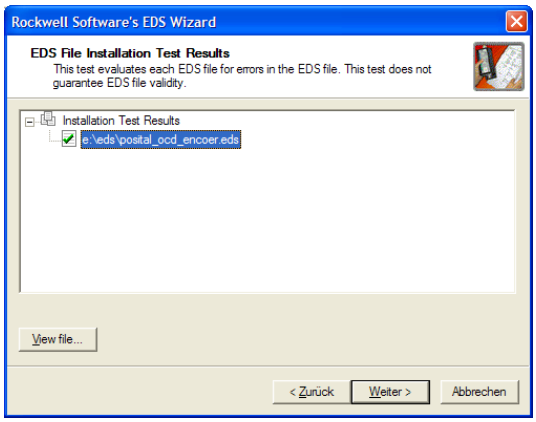
RSNetWorx™ products provide design and configuration management services for EtherNet/IP. The program defines and configures the devices on the network quickly through a simple software interface. This definition can take place offline using drag and drop operations or online by using RSLinx® to browse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

EDS Wizard

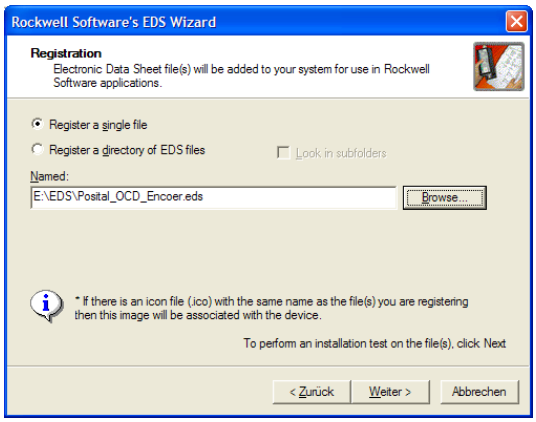
The EDS File contains information about device specific parameters as well as possible operating modes of the encoder. With this file you have a data sheet in an electronic format, which can be used to configure the device in the network, for example with RSNetWorx™ from Rockwell. In this sample the PLC uses address 192.168.0.100 and the encoder 192.100.0.252.

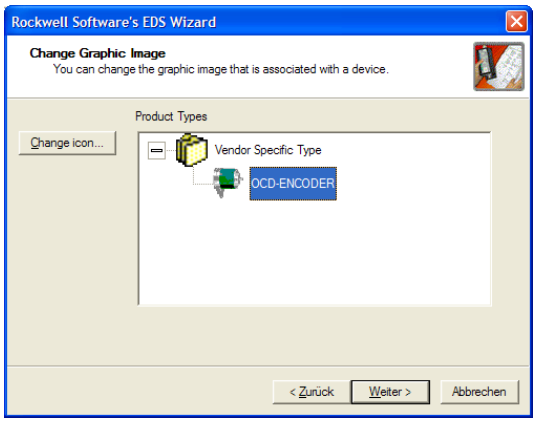
To install the EDS file the EDS Wizard has to be started, that can be done in the menu *Tools/EDS Wizard*. If the EDS Wizard is activated successfully the *Register an EDS File(s)* has to be chosen and after that the button *weiter*. In the next step the *Register a directory of EDS files* has to be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





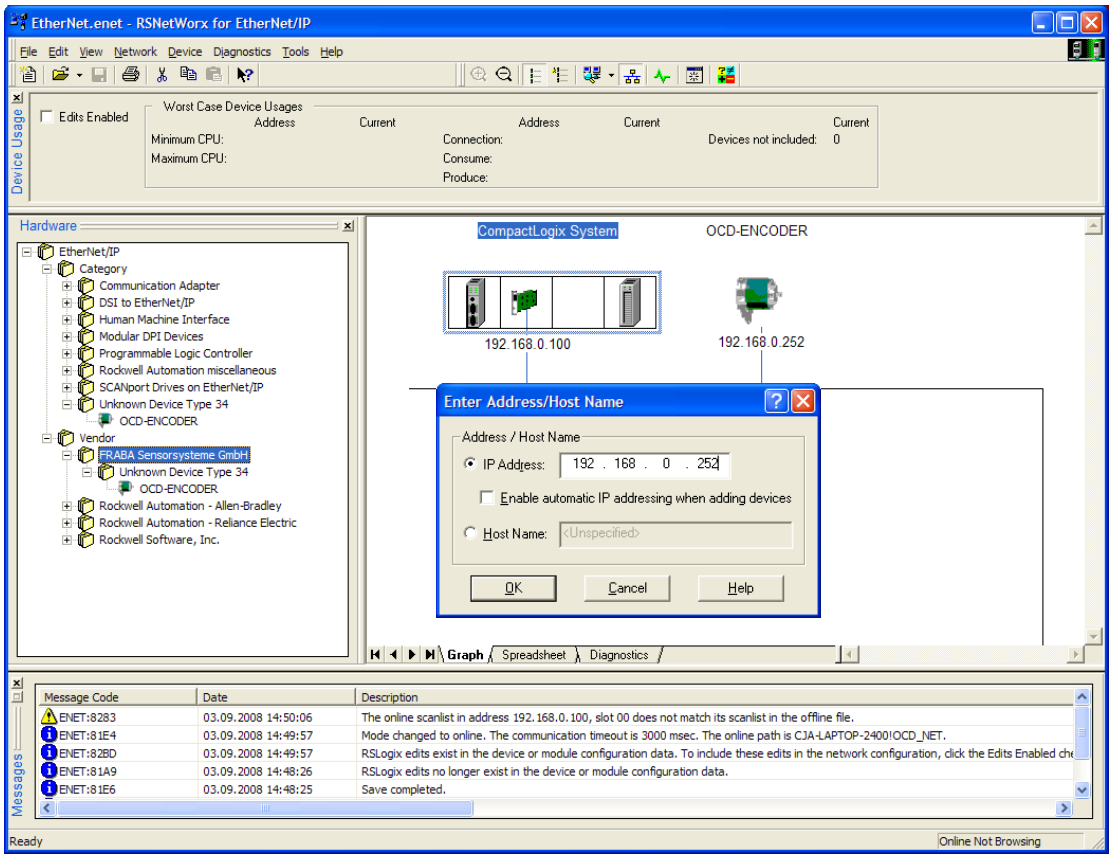
The Wizard finds all EDS files that are discarded



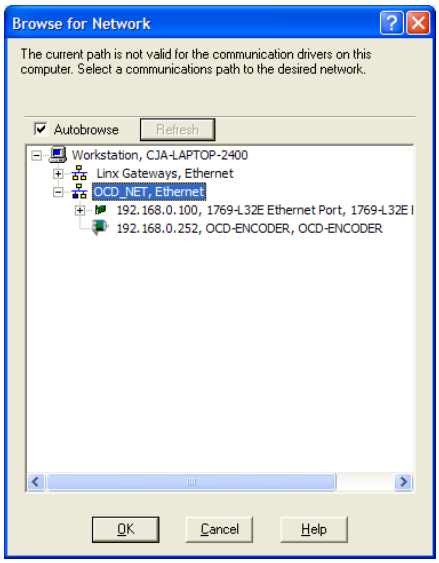


in the choosing path and operates a test to check the EDS files on errors. In the next step pictures can be selected for the using nodes. With the button *weiter* the installation can be continued and finished.

Load a saved *.enet file or start a new project. Add the devices per Drag and Drop to the net work line and set the IP-Address.



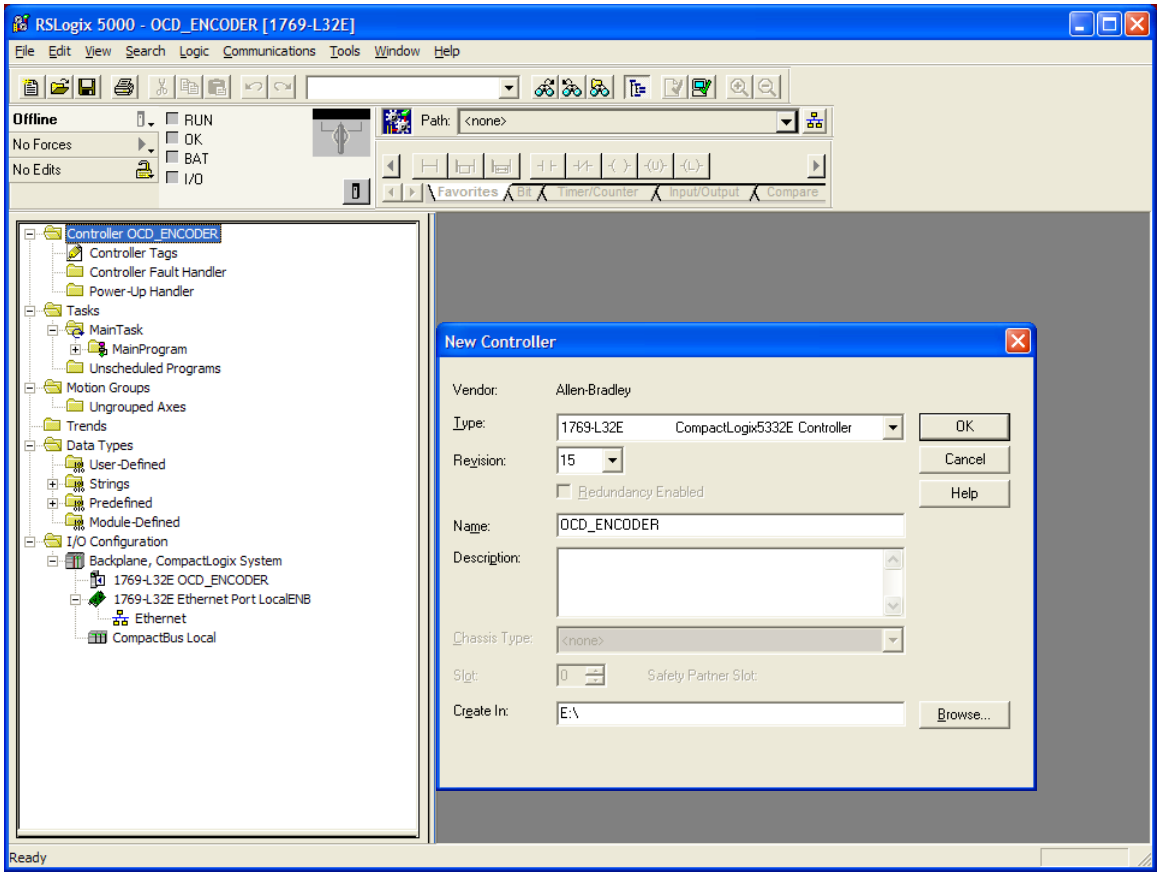
Optionally, browse the network with all devices with Button  or *Upload from Network*. It is not necessary to set the IP-Address manually. To use this configuration in RSLogix save the *.enet file.



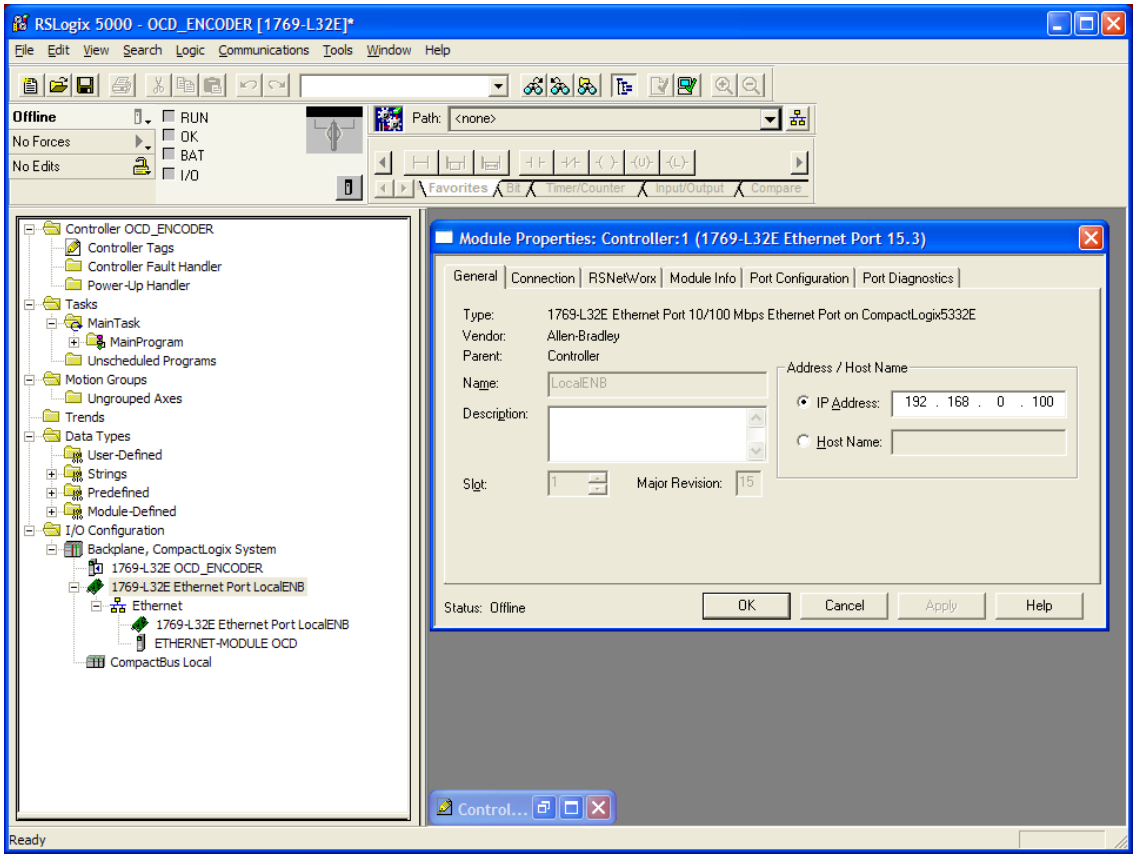
7.1.4 Configuration RSLogix 5000

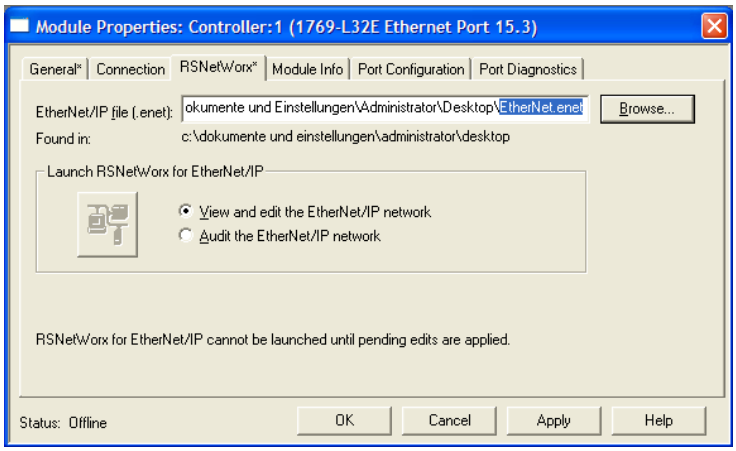
The RSLogix 5000 Series environment offers an easy-to-use, IEC61131-3 compliant interface, symbolic programming with structures and arrays, and a comprehensive instruction set that serves many types of applications. It supports relay ladder, structured text, function block diagram, and sequential function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

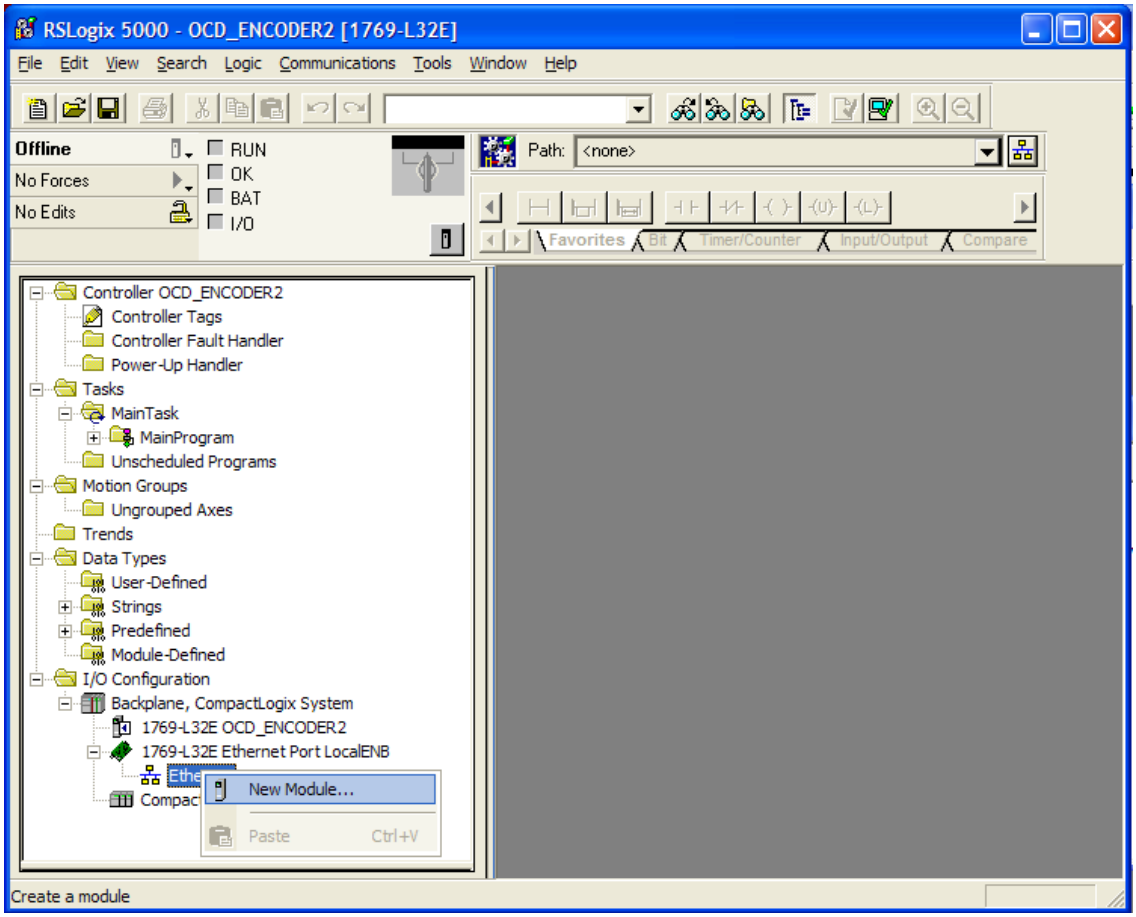


Start the configuration of the controller or load the *.enep file in the module properties of tab RSNetWorx™ that was created with RSNetWorx™.

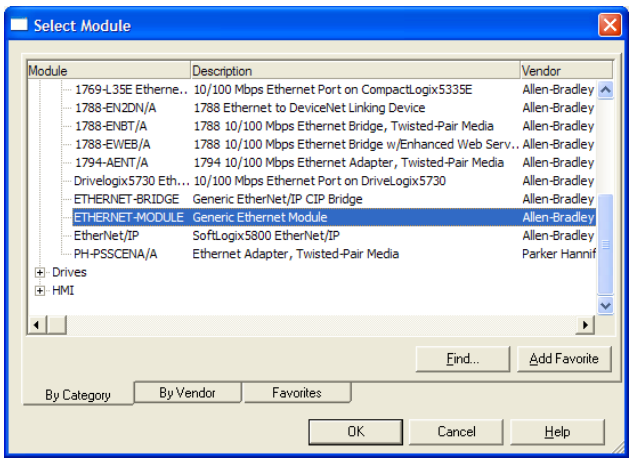




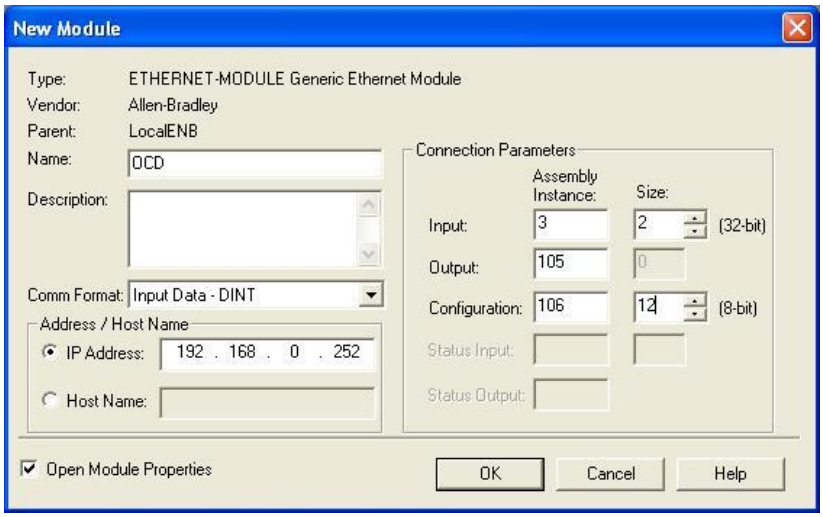
Select the network in the I/O Configuration and add New Module.



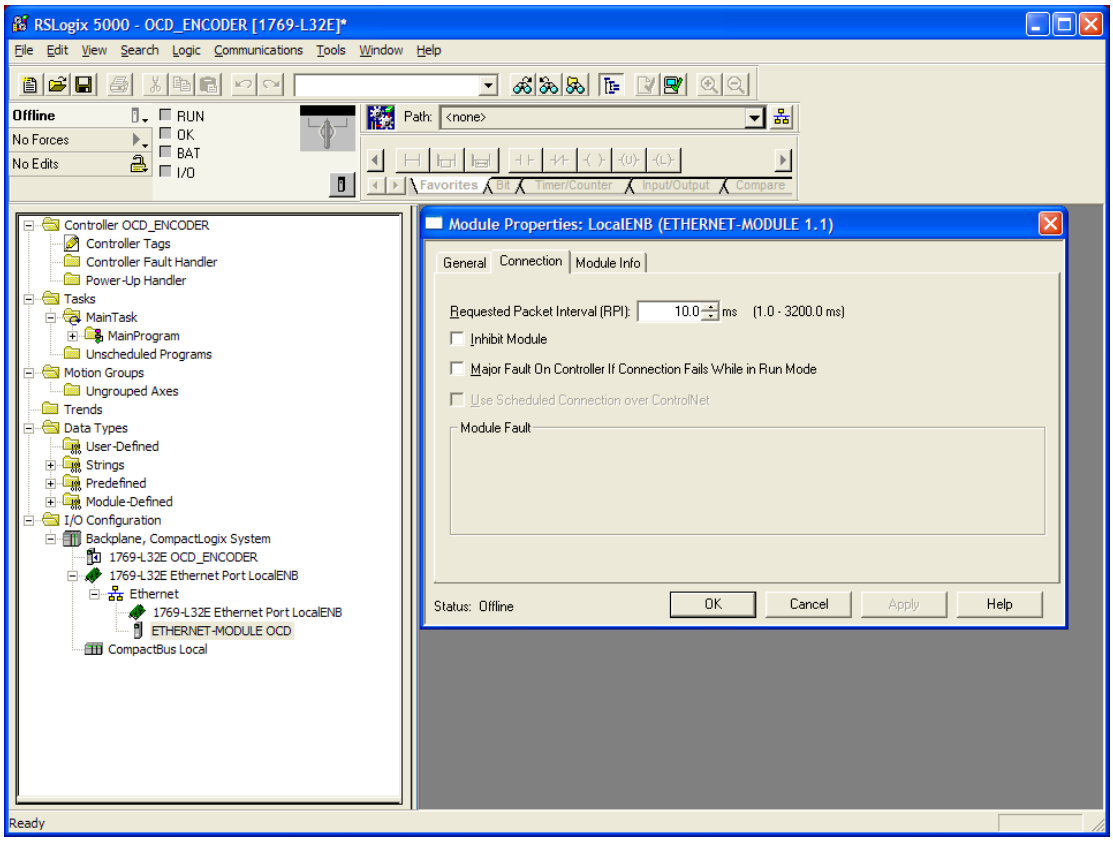
To use a Generic Device select the Generic Ethernet Module. Some PLC's support Encoder Devices too. Please check that the matching EDS file complies with the configuration of the encoder. The device type is programmable.



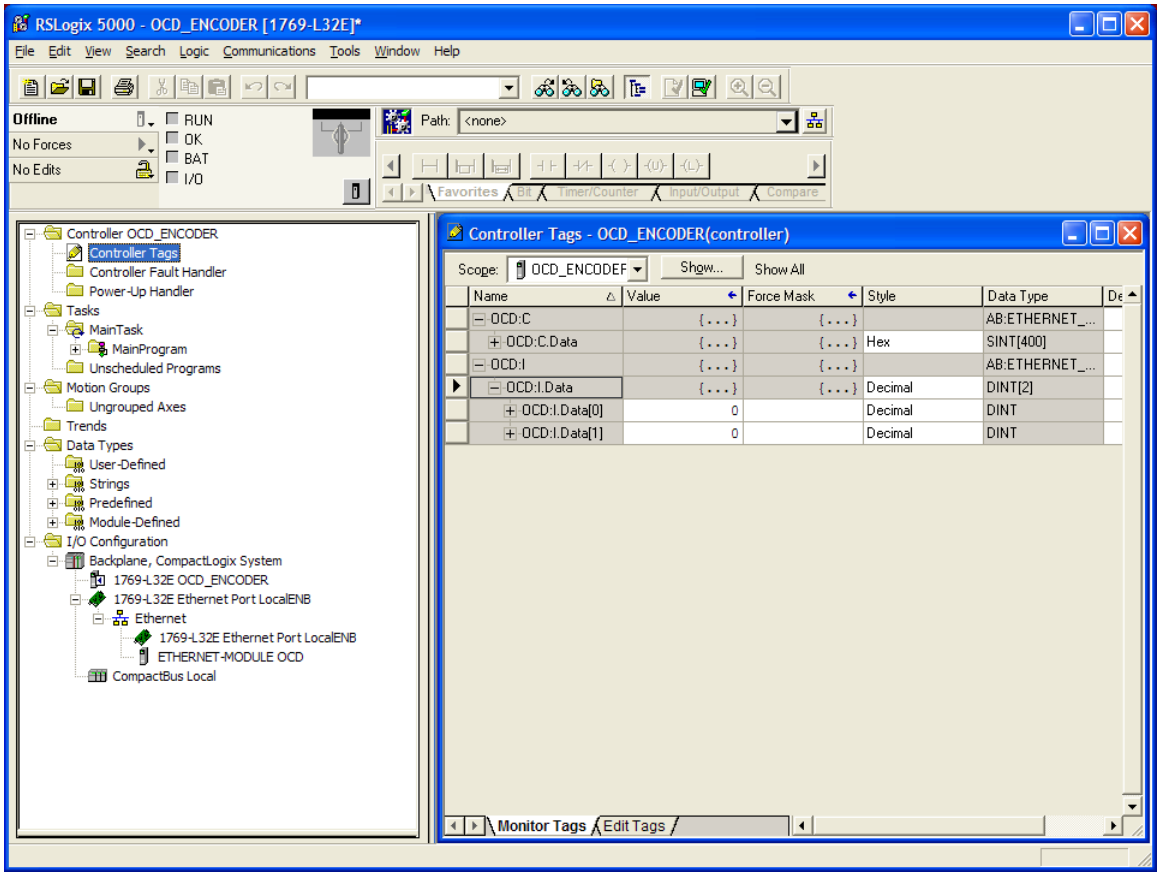
Set the Connection Parameters according the following figure.

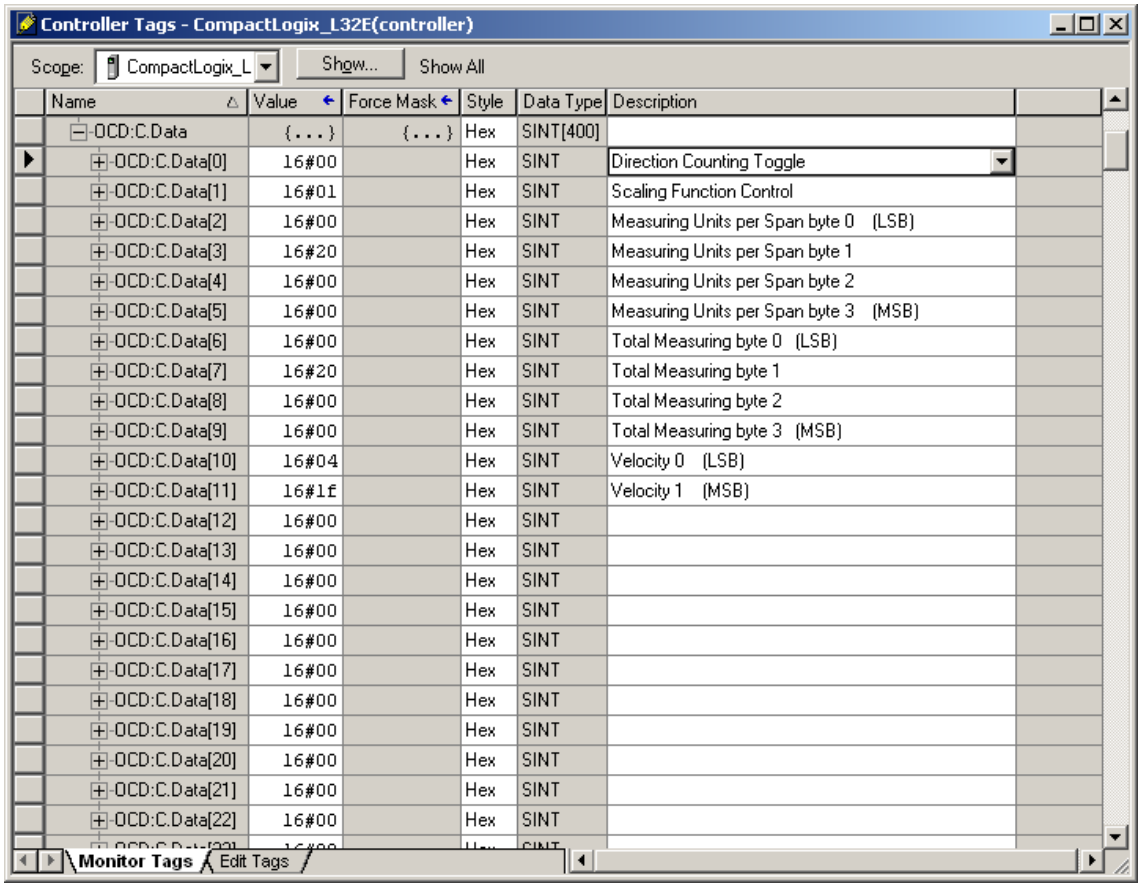


Set the cycle time.



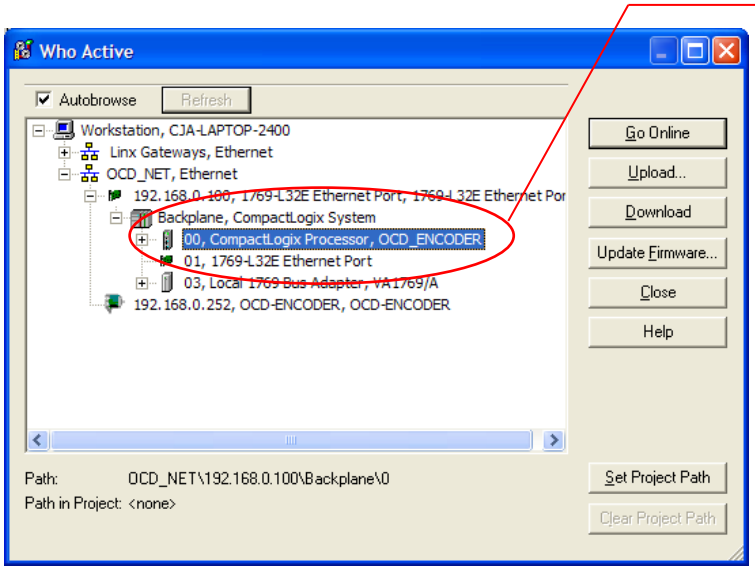
To read or write data use *Logic - Monitor Tags*



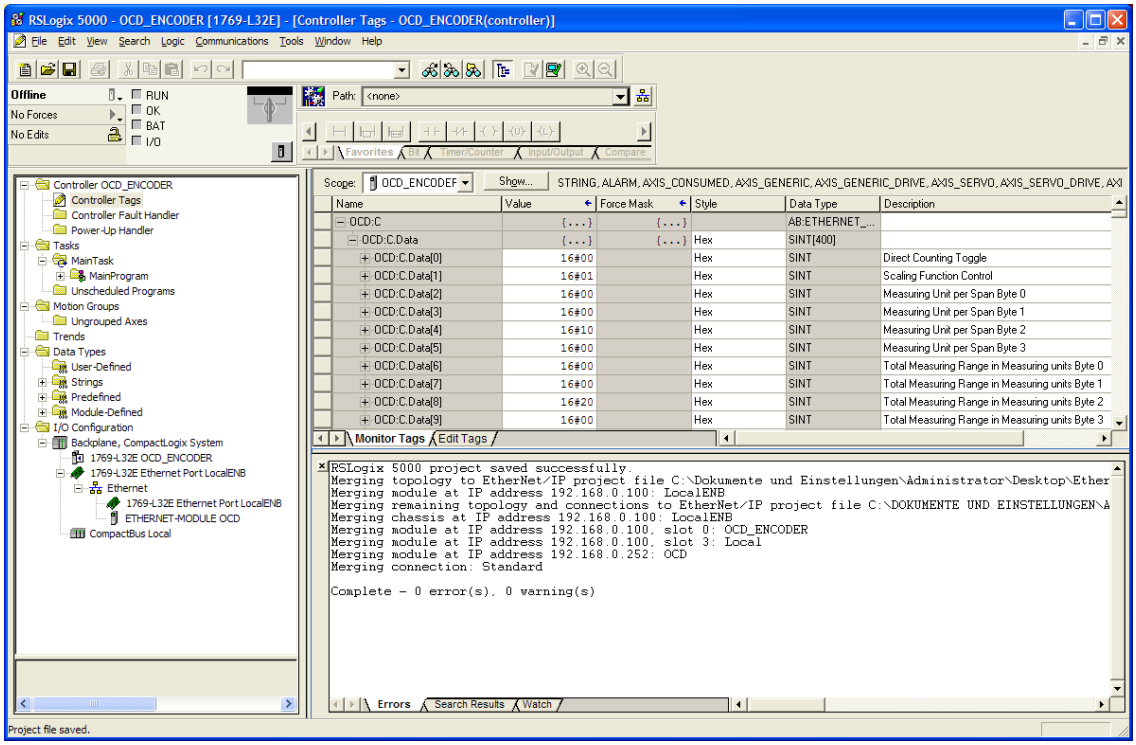


Name	Value	Force Mask	Style	Data Type	Description
OCD:C.Data	{...}	{...}	Hex	SINT[400]	
OCD:C.Data[0]	16#00		Hex	SINT	Direction Counting Toggle
OCD:C.Data[1]	16#01		Hex	SINT	Scaling Function Control
OCD:C.Data[2]	16#00		Hex	SINT	Measuring Units per Span byte 0 (LSB)
OCD:C.Data[3]	16#20		Hex	SINT	Measuring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

- If the value is 00 then the standard configuration will be used
- If the parameter is 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.

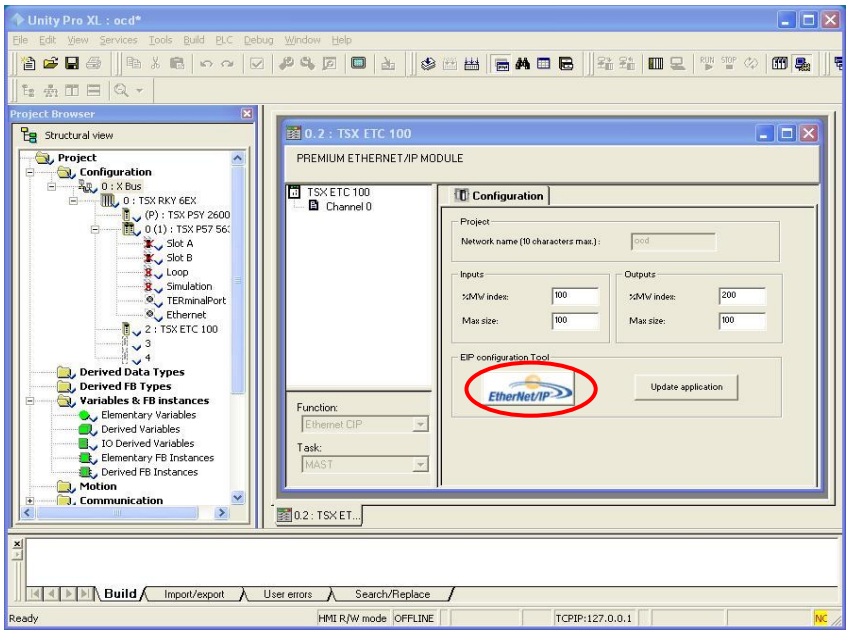


If everything is running then, in the “Errors tab” the message 0 error(s) should appear.



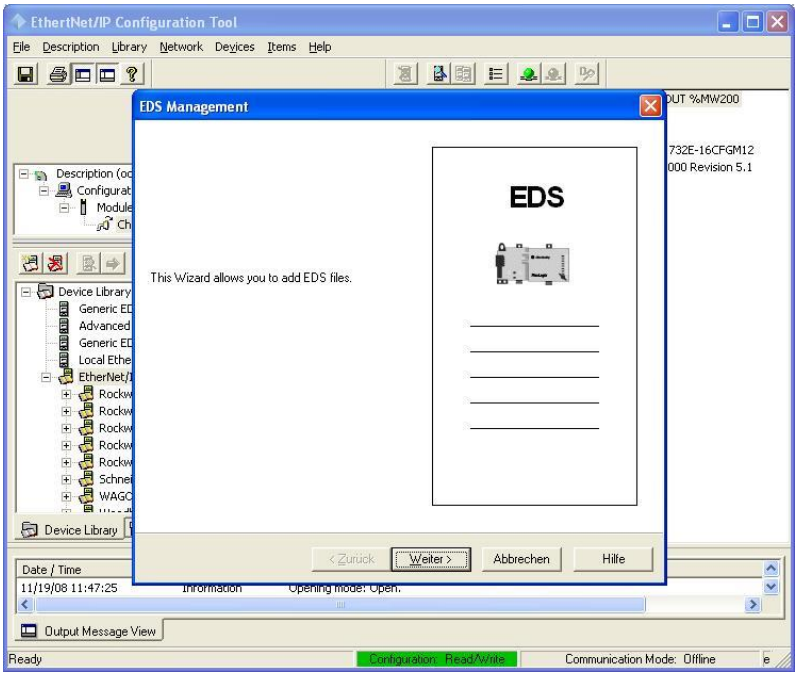
7.2 Schneider configuration tools

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

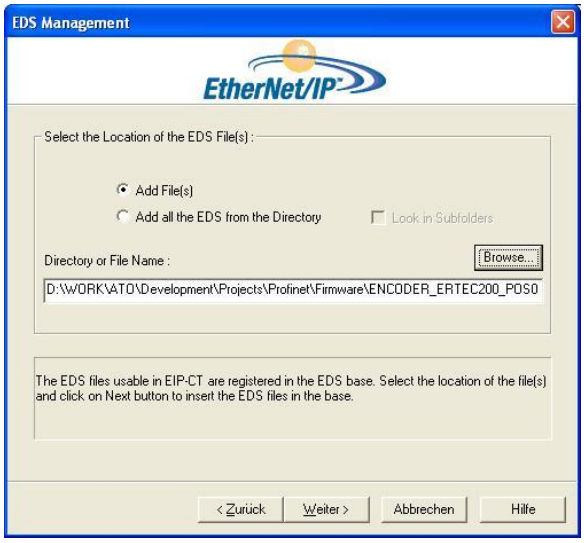


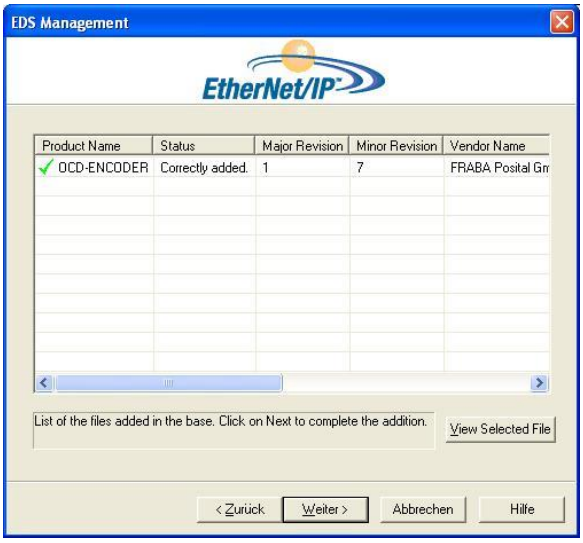
7.2.1 Setting configuration

In the first time it is necessary to install the EDS-File with the wizard.

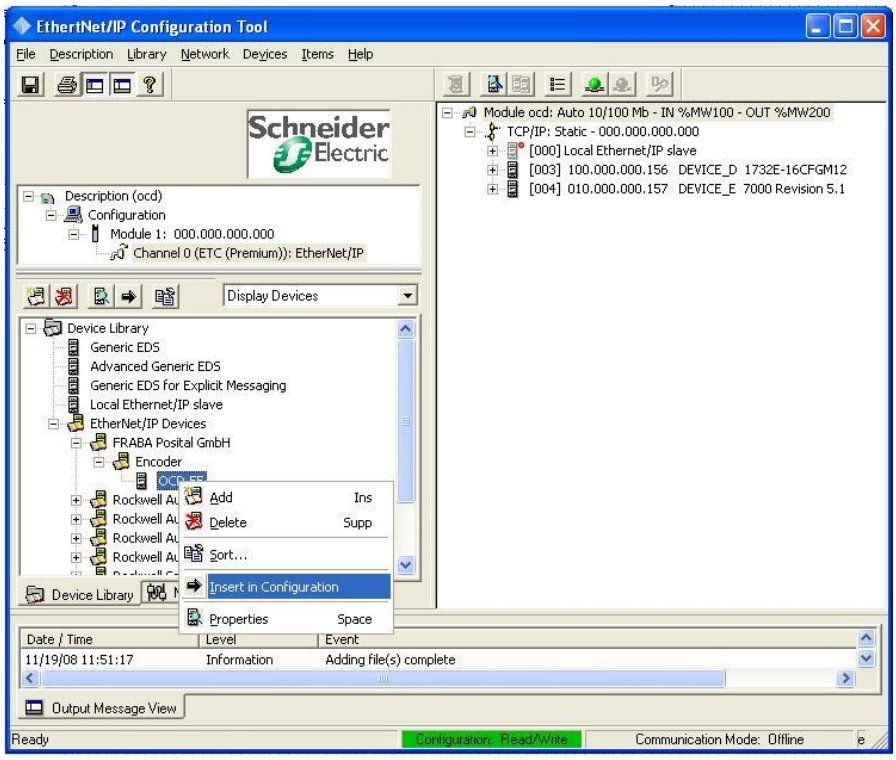


Select the EDS-File, available on our web side, and follow the wizard to the end.

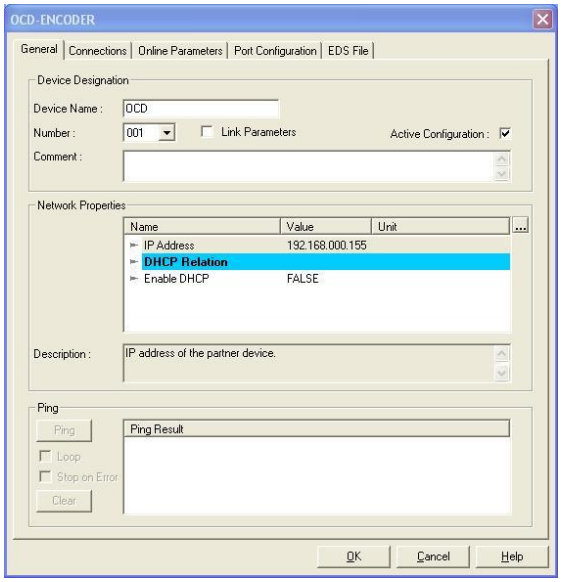




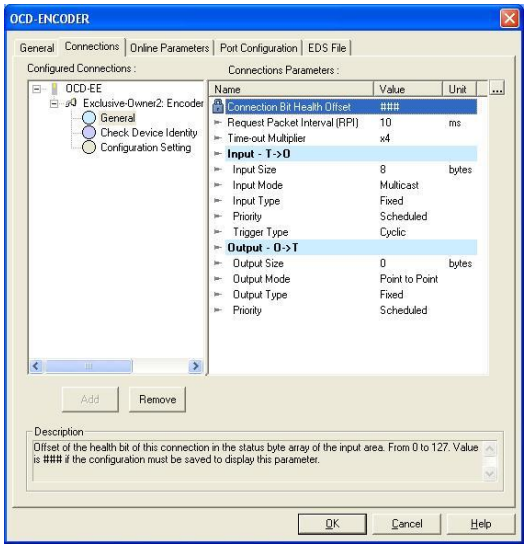
Select the encoder in the Device Library and *Insert in Configuration* (menu opens on right button click of the mouse).



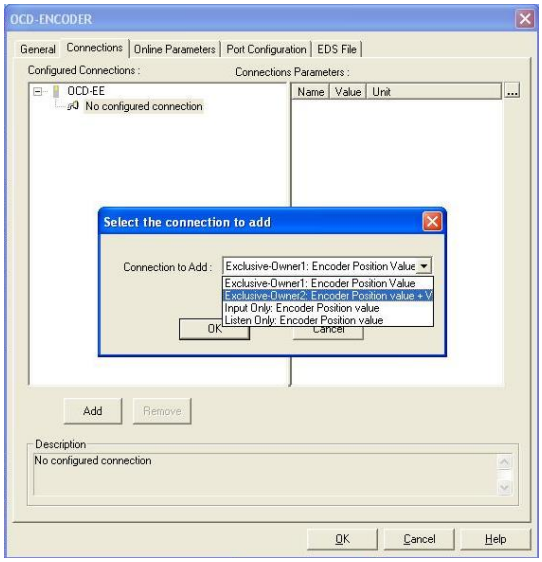
Set the IP-Address of the encoder in the *General*-Tab. Add a connection for reading the position value or the position value and velocity.



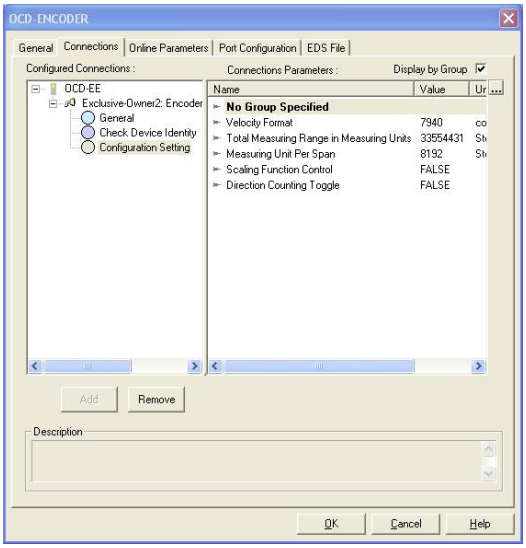
The cycle time (RPI), the input and output configuration can eb checked in tab *Connections* under *General*. The offline parameters are available Under *Configuration*



More details about the different connections are available in chapter 2.

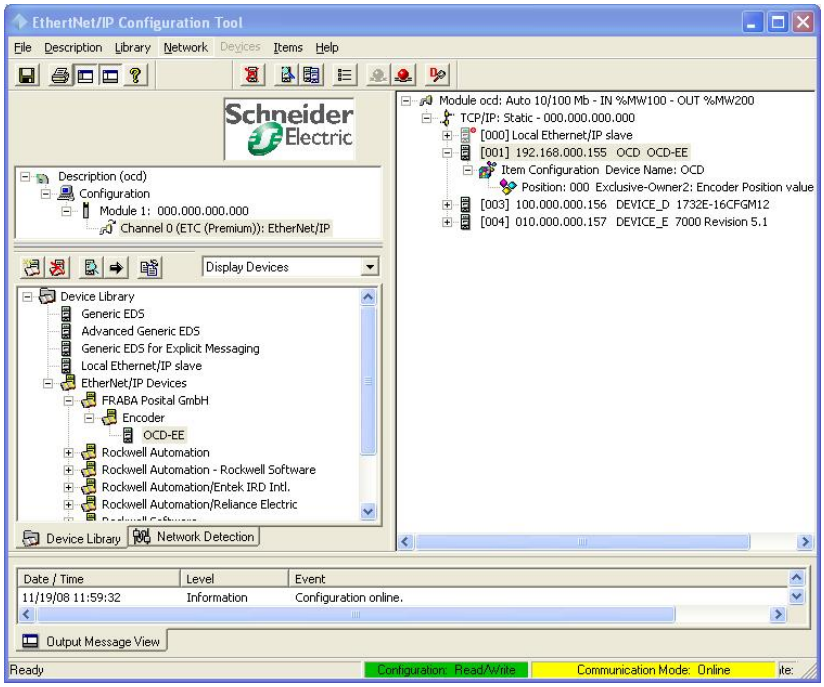


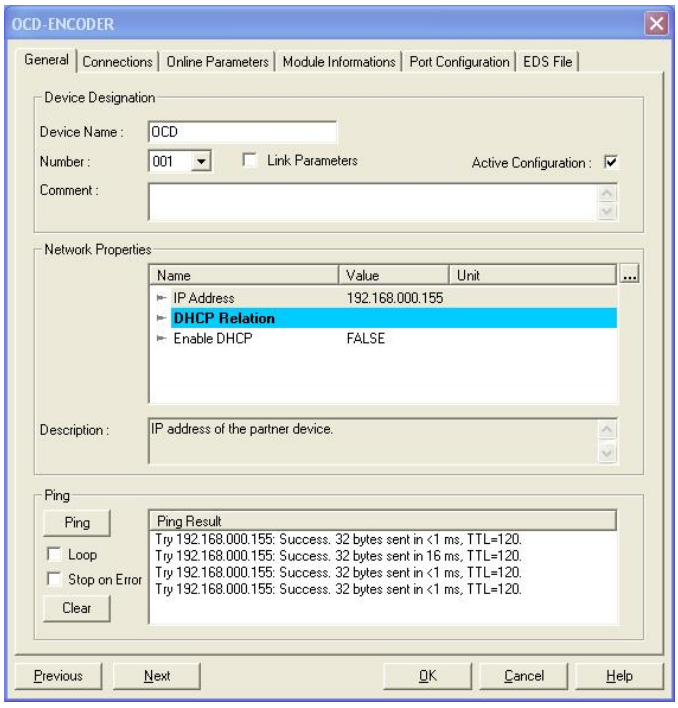
Setting, that will be used after the PLC goes in the Run state.



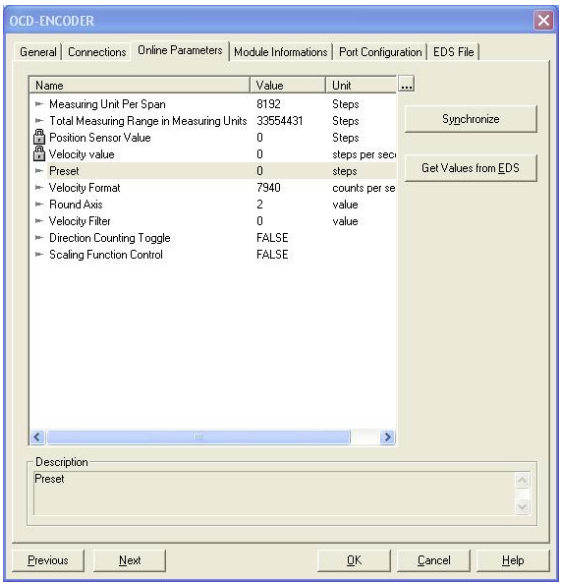
7.2.2 Online configuration

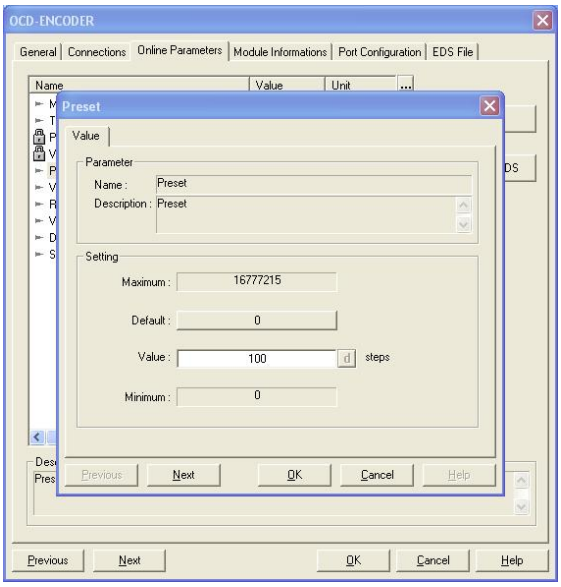
If the last steps were successful the encoder can go in the status online. In the configuration window in tab General it is possible to test the encoder connection with sending Ping commands to the encoder.

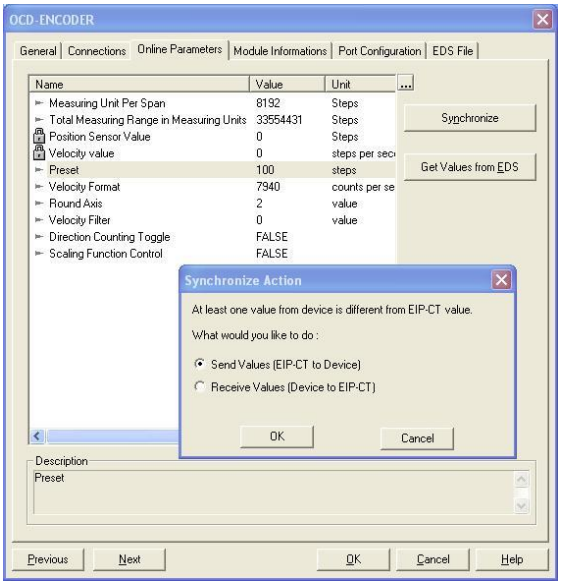




In the tab *Online Parameters* you can change the parameters. In a separate window it is possible to use the wished parameter. There are the minimum, maximum and default values available. With the *Synchronize* button it is possible to send the parameters to the encoder or to read them out from the encoder.





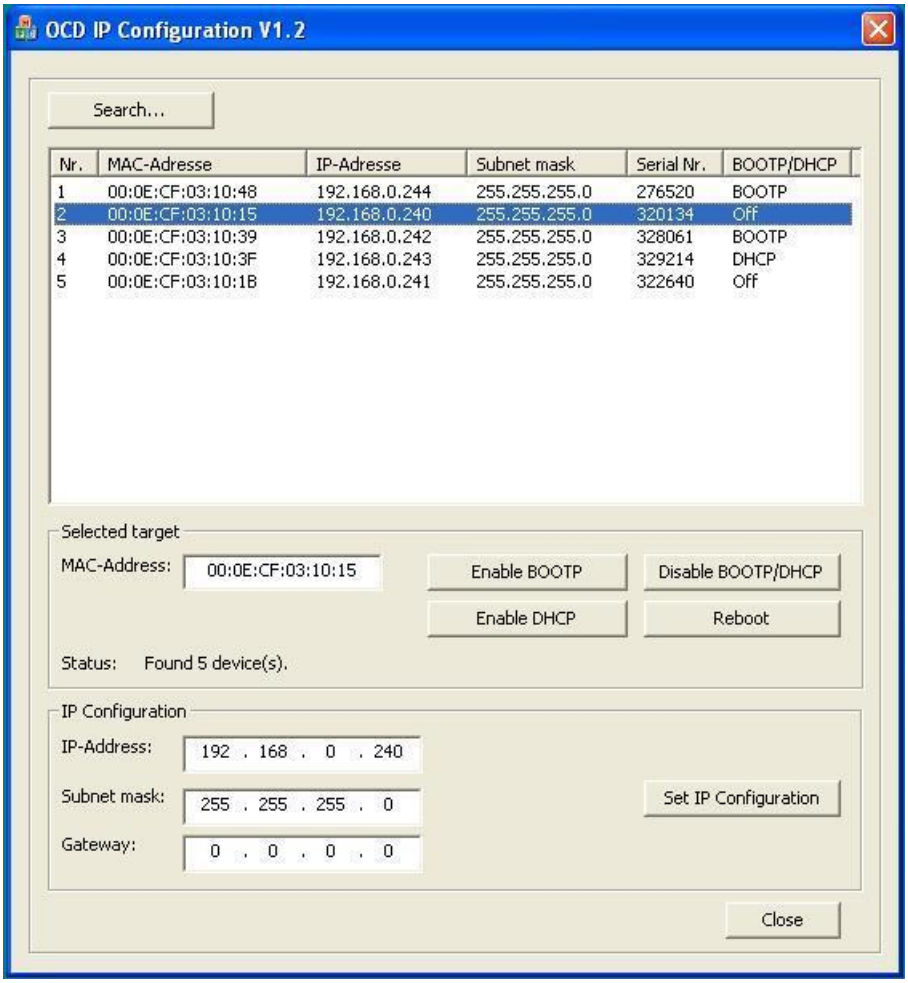


7.3 BOOTP/DHCP and IP configuration tool

For EtherNet/IP encoders it is necessary to know the IP-Address. On our website there is a special tool available for free that allows you to scan the complete network segment for MAC-Addresses of Posital encoders. Our tool will find the devices with deactivated BOOTP and DHCP too.
Connect all devices, turn on the power supply and click "*Search…*". Select the encoder and select the button corresponding to the needed functionality. After changing the status, it is necessary to click "*Search…*" again to get the actual status of the encoders.

With this tool is it possible to change the IP-Adress, the Subnet and Gateway as well.

This program use UDP Port 4000.
Warning: Nake sure that the firewall doesn't block this por



8 FAQ

8.1 Problem: IP Address unknown and BOOTP/DHCP is deactivated
Solution:
Download the tool to read out the IP-Address out of the MAC-Address from our web site: www.posital.com

8.2 Problem: I replaced a rotary encoder in the machine and now the controller cannot start the application. Additionally, the Stat LED is flashing with 4 Hz (4 flashes per seconds).
Solution: Start the BOOTP/DHCP server to set the IP-Address and deactivate BOOTP and DHCP. See chapter 7.1.1

8.3 Problem: The Posital BOOTP/DHCP Configuration Tool doesn't find the Posital encoders.
Solution: Check that TCP Port 4000 is not blocked by the firewall.

8.4 Problem: Cannot deactivate BOOT/P or DHCP
Background: Firewall and/or WLAN block communication.
Solutions:
1. Firewall must not block Port 4000 and 5000
2. Deactivate WLAN and all other network cards.

8.5 Problem: Error message "Invalid identifier"
Solution: Check that the major revision of the encoder uses the same number as the EDS-File. Check the website.

8.6 Problem: IP Address unknown and BOOTP/DHCP is deactivated
Solution: Download the tool to read out the IP-Address out of the MAC-Address from our web site.

8.7 Problem: Stat LED is flashing with 4 Hz
Background: After replacing a rotary encoder in the machine the controller cannot start the application. Additionally, the Stat LED is flashing with 4 Hz
Solution: Start the BOOTP/DHCP server to set the IP-Address and deactivate BOOTP and DHCP.

8.8 Problem: After Power-up the programmed parameters are lost.
Solution: Use the save command to save all programmed parameters in the non-volatile memory (NVM). Only Preset is saved automatically in the NVM.

8.9 Problem: Parameters from Configuration tool i.e. RSLogix overwrite the saved values of the encoder
Answer: Yes, that is how it is supposed to work. Please refer to FAQ 8.8 to have the parameters saved on the configuration tool.

8.10 Problem: Additional transmission of LLDP frames
Answer: The used stack version will transmit an additional ~1% LLDP frames. This should not be a problem with the total traffic of the network.

8.11 Problem: Problems with Configuration Tool
Answer: Check if the Version 1.4 is in use!

8.12 Problem: How many encoders can work with one Rockwell PLC?
Answer: One encoder represents a traffic of only 100 Ethernet packets per second. The PLCs can manage 6000-20000 Ethernet-packets/s. Rockwell has got two tools:
- EtherNet/IP Capacity Tool
- Integrated Architecture Builder (overdressed for analysis)

8.13 Problem: Rockwell PLC get error code 515 from our encoder
Answer:

1. Check if its cable installation is correctly done
2. Check if the network load is not too big. If so, you can use Unicast instead of Multicast configuration.
3. If the PLC has not worked before with one of our encoders, you can test our sample project from our website.

8.14 Problem: What is the fastest possible cycle time (RPI) on our EtherNet/IP encoder?
Answer: Due to the slow OS VXWorx we can only guarantee an RPI of 10ms. If the PLC uses faster RPIs it could be that the connection cannot start or will disconnect after several hours.
The next version will support DLR and use eCos as OS. It should reach a cycle time (RPI) of 1ms (realized with Profinet).

8.15 Problem: How can an encoder be replaced by a new one without setting the IP-address?
Answer: If a star topology is in use, the Rockwell switch will get the IP address from the DHCP switch and send it to the encoder.
See htttp://www.cisco.com/c/en/us/td/docs/solutions/Verticals/CPwE/CPwE_DIG/CPwE_chapter10.html
Also remember that line or ring topologies solutions with rotary switches can help.

8.16 Problem: How can one set the preset value from the Control Tag Table?
Answer: You download our quick start manual from our website and look into the slides for detailed information.

8.17 Question: What is the default IP address of the encoder?
Answer: 192.168.0.250

9 Glossar

Term	Explanation
10Base-T	Transmission line with 10 Mbit data transmission rate
100Base-T	Transmission line with 100 Mbit data transmission rate
Auto-crossing	Allow to use straight or crossover wiring
Auto-negotiation	Is an Ethernet procedure by which two connected devices choose common transmission parameters, such as speed and duplex mode
Baudrate	Transmission rate; it displays 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.
CIP	**C**ontrol and **I**nformation **P**rotocol
DHCP	**D**ynamic **H**ost **C**onfiguration **P**rotocol is a protocol used by network devices (clients) to obtain the parameters necessary to operate 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**theNet**/IP**
EMC	**E**lectro**m**agnetic **c**ompatibility, there are rules to verifying devices.
ENIP	**E**ther**N**et/**IP**
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	IO 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	Worldwide explicit address of a device. The encoder uses 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**O**D**E, 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.

We do not assume responsibility for technical inaccuracies or omissions. Specifications are subject to change without notice.

10 Revision index

Revision	Date	Revision
First release	5.6.2008	1.0
Delete section device type	2.7.2009	1.7
Delete save single attributes, Cycle of write parameters from 5,000 Mio -> 100.000, 15-30V -> 10-30V, Preset info to save in non-volatile memory, add Port 4000 FAQ	1.9.2009	1.8
Change Cycle of write parameters from 100,000 to 5 Mio	14.1.2010	1.9
Add table Control Bits	17.11.2010	1.10
Add information about filter definition, velocity, Unicast Update FAQ Delete technical data and drawings	10.1.2013	1.11
Reviewed wording, correct typos Updated FAQ Added information about filter definition, Unicast, Round Axis, Scaling Function, Preset, Velocity Format table	27.6.2018	1.12
Scaling function control: detailed explanations	25.7.2019	1.13