Inverter

i550

Inverter i550 - Cabinet

Ethernet Buses EN

PRELIMINARY

SW-Version 02.00

Lenze

Contents

1 About this manual

This manual
- applies to the i550 inverter series
- contains important technical information for the commissioning of EtherCAT, PROFINET and EtherNet/IP

i	PROFINET and EtherCAT are supported from SW Version 02.00 EtherNet/IP is supported from SW Version 02.01

1.1 Purpose of this manual

The purpose of this manual is to provide all necessary information for the setup of EtherCAT, PROFINET and EtherNet/IP of the i550 inverters.

1.2 Reference to other documents

Beside the operation manual the following documents are also required for the safe installation, commissioning and operation of the inverter:
- i500 Operation manual
- i500 Reference/Commissioning manual
- i500 Mounting and switch-on instructions
- Easy Starter software documentation

i	For general commissioning and setup refer to the i500 Operation manual or the i500 Reference/Commissioning manual.

1.3 Copyright

1.3.1 Manufacturer's address

Lenze Drives GmbH
Postfach 10 13 52, D-31763 Hameln
Breslauer Straße 3, D-32699 Extertal
Germany
HR Lemgo B 6478

Phone: +49 5154 82-0
Fax: +49 5154 82-2800
Email: lenze@lenze.com
Web: www.lenze.com

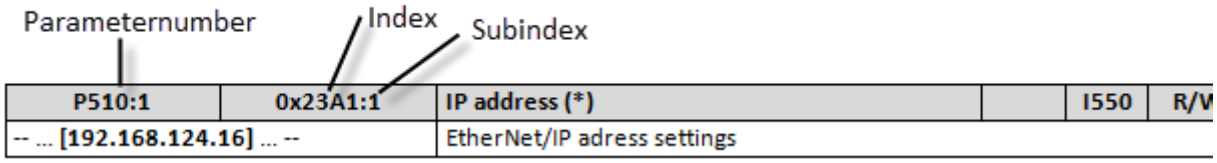
1.4 Disclaimer of liability

The descriptions, technical data and illustrations in this operation manual are subject to change without prior notice.
The procedural notes and circuit details described in this operation manual are only proposals. It is up to the user to check whether they can be adapted to the particular applications. Lenze does not take any responsibility for the suitability of the procedures and circuit proposals described.

1.5 Parameter description

Every parameter has a hexadecimal index number. Parameters which are visible on the keypad have also a parameter number. In the Easy starter the parameter number and the hexadecimal index are visible. Every parameter can have subindex.

Example	Parameter number	Index
Base Frequency	P303.02	0x2B01:002
Control select	P200.00	0x2824:000



Parameter witch are not visible on the keypad are marked in the manual as P (Without number)

Parameters or selections with marking (*) are not available on all control unit types.
Example:

P510:1	0x23A1:1	IP address (*)			I550
-- ... [192.168.124.16] ... --		EtherNet/IP adress settings			

2 Device Overview

2.1 i550 Control Unit Overview

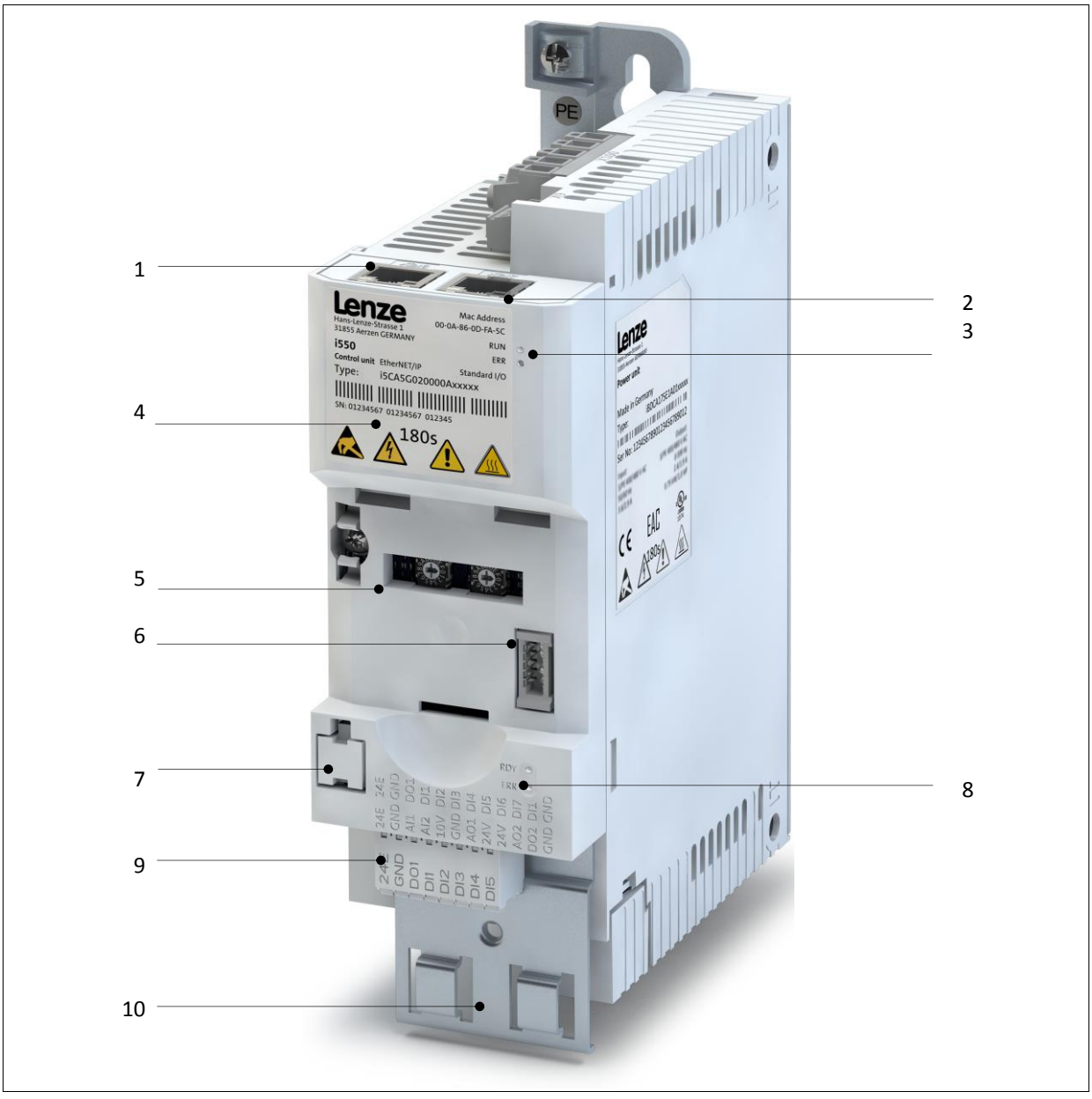


Fig. 1: i550 inverter overview

Position numbers

1) X2x6 - Fieldbus Connector 1 (EtherCAT: IN)
2) X2x7 - Fieldbus Connector 2 (EtherCAT: OUT)
3) Fieldbus Module LEDs
4) Type Plate
5) Rotary selector bus address (EtherNet/IP and EtherCAT only)
6) X16 – Interface for diagnostic module
7) X20 – Memory module
8) Inverter status LEDs
9) X3 – Control terminal (standard I/O)
10) Shield connection plate

2.2 Rotary Selector (EtherNet/IP, EtherCAT only)

The two rotary selectors can select a hexadecimal number in the range of 0x00 to 0xFF (= 0…255). They configure the last part of the device IP address (EtherNet/IP) or Identifier (EtherCAT).

Rotary Switch

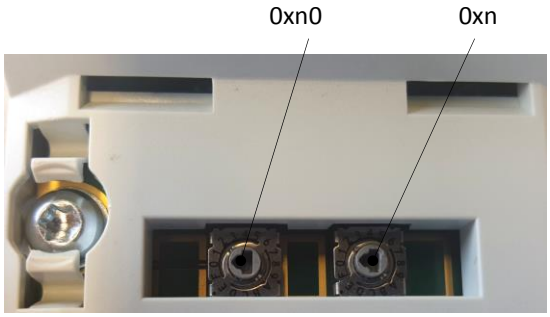


Fig. 2: Rotary Switches

EtherNet/IP

Rotary selector setting	Description
0x00	Use unmodified value configured in IP address parameter
0x01…0xFE	Use IP address value configured in IP address and overwrite last byte with selected value
0xFF	Set default IP address

The configured value is applied on next device restart or on network restart command 0x23A0:0 (P508:0).

EtherCAT

Rotary selector setting	Description
0x00	Use value configured in identifier parameter
0x01…0xFF	Use this value as identifier The value is immediately valid

The configured value is applied on next device restart or on network restart command 0x2360:0 (P508:0).

Adress/Identifier setting:

EtherNet/IP IP address:	0x23A1:1 (P510:1)
EtherCAT Identifier:	0x2361:4 (P510:4)

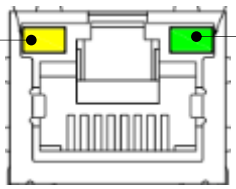
Active Adress/Identifier:

EtherNet/IP IP address:	0x2382:1 (P511:1)
EtherCAT Explicit device identifier:	0x2362:4 (P511:4)

2.3 Fieldbus connector LED

Activity LED
PROFINET and EtherNet/IP :
Flickering if link is active

EtherCAT:
Not used



Link LED
PROFINET, EtherNet/IP and EtherCAT :
On if link detected

EtherCAT Link/Activity LED:
Flickering if link is detected and active

Fig. 3: Fieldbus connector LED

2.4 Status LED

See the chapter of the corresponding fieldbus

3 Fieldbus General

Enable Network
In order to control the drive from the network the 0x2631:37 (P400:37) Network enable needs to be set (Either setting “TRUE” or mapping to a digital input to trigger signal). Once it is asserted the drive enters the network control mode.

It is important to note that in network control mode the following functions still are active:
0x2361:1 (P400:1) Inverter Enable, 0x2361:1 (P400:2) Run/Stop 0x2361:3 (P400:3) Quick Stop, 0x2361:4 (P400:4) Reset fault, 0x2361:1 (P400:5) DC Brake
All other function triggers are disabled while the drive is in the network control mode.

P400:37	0x2631:37	Network enable	i510	i550	R/W
0:Not connected (Reference see P400:1)		Enables the network for control **State:** TRUE: Network is enabled FALSE: Network is disabled **Note:** If Network is enabled (**Network enable 0x2361:37, P400:37** is HIGH) it is not mandatory that **Inverter enable (0x2361:1, P410:1)** or **Run/Stop (0x2361:2, P410:2)** are assigned to a digital input (DI1-7). They can be set to [1] Constant TRUE to Enable and Run the inverter without using digital inputs (DI1-7).			

Several command words, status and setpoint are available to control the drive from remote:

- CIA402 (Predefined mapping) Used for EtherCAT/CAN
- AC drive profile (Predefined mapping) Used for EtherNet/IP
- Lenze Legacy (Predefined mapping)
- Network IN/Out (Configurable mapping)

See chapter 7 Drive Profile for details.

4 PROFINET

4.1 General Considerations

> **i** PROFINET is supported from SW Version 02.00

PROFINET transmits parameter data, configuration data, diagnostic data, alarm messages, and process data between the host system (IO controller) and the controllers that are part of the fieldbus (IO devices). Depending on their time-critical behavior, data are transmitted via corresponding communication channels.

4.1.1 Communication channels

- The process data channel transmits time critical process data. Process data are transmitted cyclically between the IO controller and the IO devices.
- With the process data the controller is actuated.
- The IO controller can directly access the process data. In the PLC, for instance, the data are directly assigned to the IO area.
- Process data are e.g. setpoints, actual values, control and status words.

> **i** There is one channel for incoming (RX) and one channel for outgoing (TX) information:
> Incoming data (Rx data): Process data from the inverter (IO device) to the IO controller
> Outgoing data (Tx data): Process data from the IO controller to the inverter (IO device)

- Parameter data are transmitted via the acyclic channel.
- Usually the transmission of parameter data is not time-critical.
- The access to the parameter data uses the PROFIdrive profile.
- Examples of parameter data are operating parameters, motor data, and diagnostic information.
- The acyclic channel provides access to all drive parameter objects.
- Parameter changes must be saved via parameter 0x2022:3 (P700:3)

4.1.2 Supported services

Feature	Inverter Drive
Conformance Class	CCB
Option according Conf.Class	Media Redundancy Protocol
Device Class	I/O Device
According PN Specification	V2.2
Safety Channel Support	-
Shared Device	-
Device Access	TCI, I&M0..4
Device Profile Support	-

Supervisor Access	✓
Second controller AR	Yes (initial revision of i550 does not support a second ctrl AR)
Fast Startup	No. (Typical startup time appx. 11 sec.)
Topology support	LLDP, MIB , Station Alias
PN Blink function	✓
Alarm Type	User
Acyclic Services	✓
Additional Ethernet Channel	TCP/IP Channel
Lenze GCI Support	✓
Lenze ESDCP Support	✓
Power over Ethernet PoE	-
External 24V Power Supply	Yes (Supply the Control Unit over X3 24E/GND Pins)
Optical Fiber support	-

4.1.3 Device Description File GSDML

The device description file GSDML can be found in the Lenze download area.

4.2 Configuration and identification (I&M)

P508:0	0x2380:0	Activate network (*)		i550	R/W
0: No action/no error 1: Restart with actual values 2: Restart with default values 5: Stop network communication 10: In progress 11: Action cancelled 12: Fault		Controls network activities.			
P510:1	**0x2381:1**	**IP address (*)**		**i550**	**R/W**
-- ... **[0.0.0.0]** ... --		Defines the IP address of the I/O device.			
P510:2	**0x2381:2**	**Subnet (*)**		**i550**	**R/W**
-- ... **[0.0.0.0]** ... --		Defines the subnet mask of the I/O device.			
P510:3	**0x2381:3**	**Gateway (*)**		**i550**	**R/W**
-- ... **[0.0.0.0]** ... --		Defines the gateway address of the I/O device.			
P510:4	**0x2381:4**	**Station name (*)**		**i550**	**R/W**
-- ... **[--]** ... --		Defines the station name of the I/O device.			

P	0x2381:5	I&M1 System designation (*)		i550	R/W
-- ... **[--]** ... --		Defines the I&M1 System designation of the I/O device.			
P	**0x2381:6**	**I&M1 Installation site (*)**		**i550**	**R/W**
-- ... **[--]** ... --		Defines the I&M1 Installation site of the I/O device.			

P	0x2381:7	I&M2 Installation date (*)		i550	R/W
-- ... **[--]** ... --		Defines the I&M2 Installation date of the I/O device.			
P	**0x2381:8**	**I&M3 additional information (*)**		**i550**	**R/W**
-- ... **[--]** ... --		Defines the I&M3 Additional information of the I/O device.			
P	**0x2381:9**	**I&M4 signature code (*)**		**i550**	**R/W**
-- ... **[--]** ... --		Defines the I&M4 Signature code of the I/O device.			

4.3 Diagnostics

Active settings can be checked in the Parameters listed below:

P511:1	0x2382:1	Active IP address (*)		i550	R
-- ... **[Actual value]** ... --		Displays the active IP address of the I/O device.			
P511:2	**0x2382:2**	**Active subnet (*)**		**i550**	**R**
-- ... **[Actual value]** ... --		Displays the active subnet mask of the I/O device.			
P511:3	**0x2382:3**	**Active Gateway (*)**		**i550**	**R**
-- ... **[Actual value]** ... --		Displays the active gateway address of the I/O device.			
P511:4	**0x2382:4**	**Active station name (*)**		**i550**	**R**
-- ... **[Actual value]** ... --		Displays the active station name of the I/O device.			
P511:5	**0x2382:5**	**MAC Address (*)**		**i550**	**R**
-- ... **[Actual value]** ... --		Displays the MAC address of the I/O device.			
P516:0	**0x2388:0**	**Network status (*)**		**i550**	**R**
Bit # description: 0: Initialized 1: Online 2: Connected 3: IP address error 4: Hardware fault 6: Watchdog elapsed 7: Protocol error		Displays the bus status of the I/O device.			
P517:1	**0x2389:1**	**Network error1 (*)**		**i550**	**R**
0: No error 2: Unit ID unknown 3: Max. units exceeded 4: Invalid size 5: Unit type unknown 6: Runtime plug 7: Invalid argument 8: Service pending 9: Stack not ready 10: Command unknown 11: Invalid address descriptor		Displays the bus error1 of the I/O device.			
P517:2	**0x2389:2**	**Network error2 (*)**		**i550**	**R**
Bit # description: 7: IP address error 8: Station name problem 9: DataExch left 10: Stack boot error 11: Stack online error		Displays the bus error2 of the I/O device.			

12: Stack state error 13: Stack revision error 14: Initialization problem 15: Stack init error	

4.4 Errors and Error Reaction

P515:1	0x2859:1	Watchdog elapsed (*)		i550	R/W
2:Trouble (Reference see P310:1)		Configuration of the watchdog fault reaction			
P515:3	0x2859:3	Invalid confg by Master (*)		i550	R/W
2:Trouble (Reference see P310:1)		Configuration of the invalid configuration reaction			
P515:4	0x2859:4	Stack initialization error (*)		i550	R/W
2:Trouble (Reference see P310:1)		Configuration of the fieldbus stack initialization error			
P515:5	0x2859:5	Invalid process data (*)		i550	R/W
2:Trouble (Reference see P310:1)		Configuration of the invalid process data reaction			

4.5 PDO Mapping

The PDO mapping is done in the master configuration and automatically sent to the inverter. The actual mapping can be checked in 0x24E0 and 0x24E1. The data format is 0xAAAABBCC (AAAA=Index, BB=Subindex, CC=Length)

4.6 PROFINET quick start

1. Power the device externally either by mains or external 24V supply and connect the inverter to the PC using the USB diagnostic adapter.
2. Set address and station name:
   The station name, the IP address, subnet mask, and gateway address can either be assigned by the PLC-controller via PROFINET or set manually in the Easy starter.

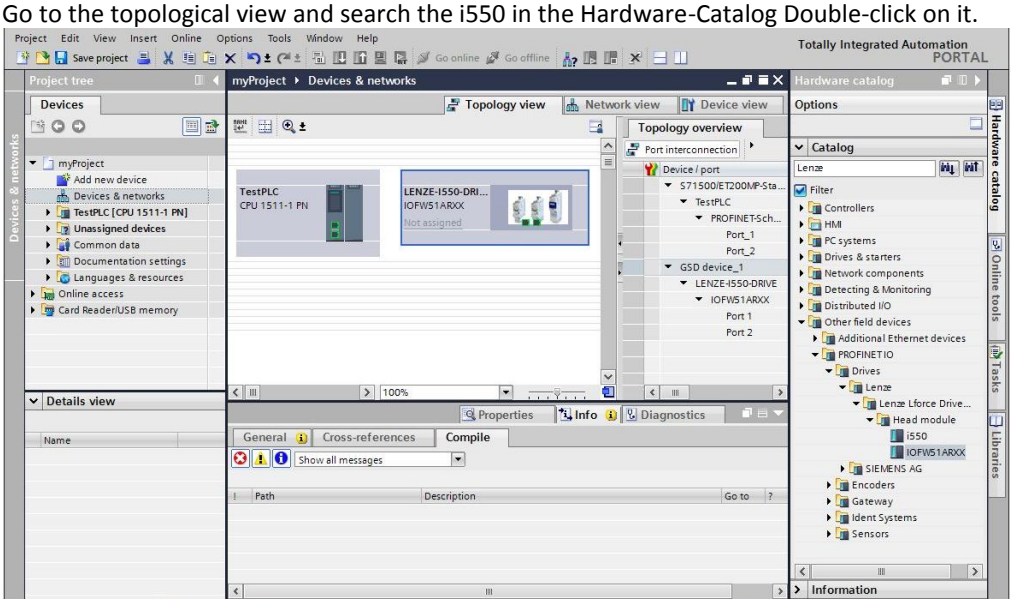
   If the settings are done with Easy Starter set the following parameters:
   a. Communication parameters:

Name	Parameter	Example:
IP Address	0x2381:1 (P510:1)	192.168.178.2
Subnet	0x2381:2 (P510:2)	255.255.255.0
Gateway	0x2381:3 (P510:3)	(Normally not needed)
Station name	0x2381:4 (P510:4)	i550_MainDrive

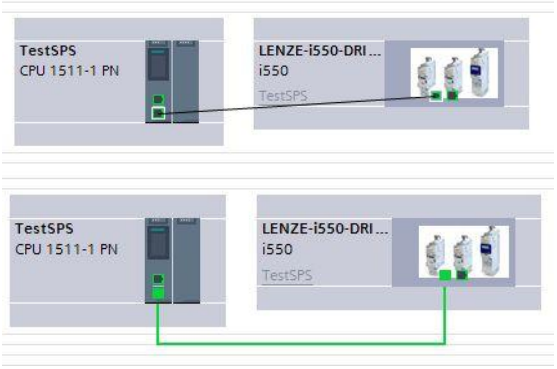
   b. Save the parameter with 0x2022:3 (P700:3) and power cycle the drive that the configuration takes effect.

   (The settings can also be done later directly by PLC Master as explained under point 7)
3. For network control the following setup needs to be done:
   - Set 0x2631:37 (P400:37) “Network enable” to “TRUE” [1]
   - Set 0x2860:1 (P201:1) “Default frequency setpoint” to “Network Frequency Setpoint” [5].
   Save the parameter with 0x2022:3 (P700:3).
4. Install the GSDML-file in the Master PLC to use the i550 device over PROFINET.

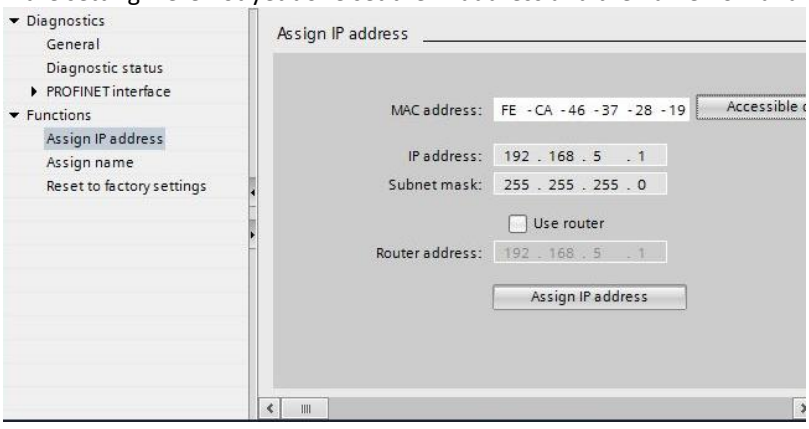
5. Siemens TIA Portal: Add the i550 to the project:
   Go to the topological view and search the i550 in the Hardware-Catalog Double-click on it.



6. Go back to topological view and connect the i550 to the PLC using the mouse:



7. Select the correct IP address and station name if they were set before with Easy Starter.
   If the setting were not yet done set the IP address and the name now and assign them.



8. Process data configuration
   The process data configuration must be configured in the PROFINET configuration tool.
   The default configuration in the i550 GSDML -file is:

PLC to Drive:

Lenze control word (NetWordIN1)	0x4008:1 (P590:1)
Network frequency setpoint 0.01Hz	0x400B:5 (P592:5))
16Bit selectable OUT-Data	The selected parameter to be transferred is selected with the parameter of this module, default set to 0x2601:2 (P202:2)

Drive to PLC:

Lenze status word (NetWordOUT1)	0x400A:1 (P591:1)
Actual Speed [0.01 Hz]	0x400C:6 (P593:6)
Actual motor current [0.1A]	0x2D88:0 (P104:0)

Device overview

Module	Rack	Slot	I address	Q address	Type	Article no.
LENZE-I550-DRIVE	0	0			IOFW51ARXX	IOFW51ARXX
IOFW51ARXX	0	0 X1			LENZE-I550-DRIVE	
L-Controlword 0x4008:01 _1	0	1		256...257	L-Controlword 0x4008:01	
Net freq. 0.01Hz 0x400B:05_1	0	2		258...259	Net freq. 0.01Hz 0x400B:05	
16Bit selectable OUT-Data_1	0	3		260...261	16Bit selectable OUT-Data	
L-Statusword 0x400A:01_1	0	4	258...259		L-Statusword 0x400A:01	
Act.freq. 0.01Hz 0x400C:06_1	0	5	260...261		Act.freq. 0.01Hz 0x400C:06	
Motor current A 0x2D88:00_1	0	6	256...257		Motor current A 0x2D88:00	

The configuration of the process data is automatically sent to the drive. Also the Bit-configuration of NetWordIN1 and NetWordOUT1.

The Bit configuration can be adjusted using the “parameterization” (Baugruppeneinstellung) of the corresponding module. The default setting for the Lenze control word (NetWordIN1) and Lenze status word (NetWordOUT1) are as follow:

Lenze control word (NetWordIN1)	
Bits	**Function**
0	Not connected
1	Not connected
2	Quick Stop
3	Run Forward
4	Invert Rotation
5	DC Brake
6	Not connected
7	Reset Fault
8	Preset Select 1 (bit 0)
9	Preset Select 2 (bit 1)
10	Not connected
11	Not connected
12	Not connected
13	Not connected
14	Not connected
15	Not connected

Lenze status word (NetWordOUT1)	
Bits	**Function**
0	Ready to run
1	Inverse Rotation
2	Running
3	Fault
4	Safe Torque Off
5	Quick stop active
6	Actual speed = 0
7	Inverter warning
8	Mechanical brake release
9	At current limit
10	At speed setpoint
11	Not connected
12	Not connected
13	Not connected
14	Not connected
15	Not connected

9. Start/Stop the drive
   - With the default IO-setting DI1 need to be asserted (Run/Stop)
   - Set “Network frequency setpoint 0.01Hz”, (Example 1234 = 12.34 Hz)
   - Set Bit 3 of “Lenze control word (NetWordIN1)” to start the drive

> Per default the watchdog timeout response for the communication is set to trouble (0x2859:1, P515:1).

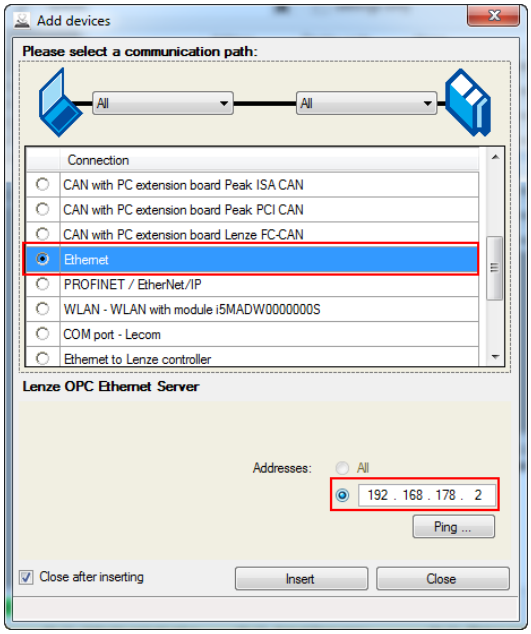
4.7 PROFINET Connection to Engineering Tools

1. Power the device externally either by mains or external 24V supply and connect the inverter to the PC using the USB diagnostic adapter.
2. Set address and station name:
   The station name, the IP address, subnet mask, and gateway address can either be assigned by the PLC-controller via PROFINET or set manually in the Easy Starter.

   If the settings are done with Easy Starter set the following parameters:
   a. Communication parameters:

Name	Parameter	Example:
IP Address	0x2381:1 (P510:1)	192.168.178.2
Subnet	0x2381:2 (P510:2)	255.255.255.0
Gateway	0x2381:3 (P510:3)	(Normally not needed)
Station name	0x2381:4 (P510:4)	i550_MainDrive

   b. Save the parameter with 0x2022:3 (P700:3) and power cycle the drive that the configuration takes effect.
3. Connect the PC to the i500 PROFINET with an Ethernet cable.
4. Set the IP address of the network adapter in your PC to the same range than the inverter IP address.
   Example:
   IP Address 192.168.178.1
   Subnet 255.255.255.0
   Select the connection “Ethernet” in the Easy Starter
   Set the correct IP address of the device and add it. Now the objects of the inverter are accessible in Easy Starter.



4.8 Troubleshooting

4.8.1 PROFINET LED status display

LED	Status	Meaning
BUS RDY	Off	No Master Connection
		PLC in STOP
		PLC in RUN (DataExchange)
BUS ERR	Off	No Error
		Profinet Identification Cycle
		Profinet Init Error (Stack, IP, Stationname)
		Timeout DataExchange

4.8.2 PROFINET specific messages

Error Code (Hex)	Error Code (Dec)	Fault Type	Tool text	Description
0x8190	33168	Configurable (0x2859:1)	Watchdog timeout	PROFINET watchdog expired (Cyclic communication connection interrupted).
0x8286	33414	Configurable (0x2859:3)	PDO map error	PROFINET PDO mapping error (fault with cyclic data configuration)
0x8192	33170	Configurable (0x2859:4)	Communic.error	PROFINET communication stack initialization error (wrong station name / IP address during initialization)
0x8193	33171	Configurable (0x2859:5)	Inv cycl data	PROFINET invalid cyclic data (IO data marked bad)

5 EtherCAT

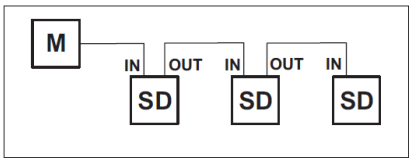
5.1 General Considerations

EtherCAT is supported from SW Version 02.00

5.1.1 Network topology

An EtherCAT frame is sent through a pair of wires from the master to the slaves. The frame is forwarded from slave to slave until it has passed through all the devices. Finally, the last slave returns the frame to the master through a second pair of wires. In this way, EtherCAT always forms a logic ring topology, irrespective of the topology used.

Line topology



M = master
SD = slave device

The devices are interconnected successively.

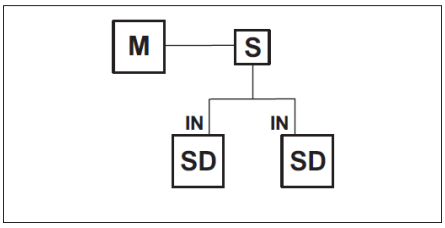
In order to ensure trouble-free operation, it is required to assign and wire the EtherCAT inputs (IN) and EtherCAT outputs (OUT) correctly!

The receiving line is plugged into socket X2x6 (IN), the forwarding line into socket X2x7 (OUT).

The direction of data transmission is from the master to the slaves.

The termination of the last EtherCAT node is effected automatically by the slave.

Switch topology



M = master
S = switch
SD = slave device

The wiring can also be carried out in a star structure via an appropriate switch. For this, observe the additional runtimes.

5.1.2 Supported services

Feature	Inverter Drive
Freerun / Config / Run Mode	✓
Access Mode logical Write (W) logical ReadWrite (RW)	RW
No. of FMMU's	3 *
No. of SM	4
CANopen over EtherCAT (CoE)	✓
Topology Addressing	✓
Explicit Device Identification Mode	✓
Second Slave Address	only via EEPROM
DC synchronization	-
Safety over EtherCAT (FSoE)	-

* Required for mapped mode

5.2 Configuration

The EtherCAT nodes are normally addressed via a fixed 16-bit address defined by the EtherCAT master. During start-up, the master assigns this address to each node, depending on the physical order in the EtherCAT network. The address is not saved and is lost when the device is switched off.

Fixed-address addressing
With the fixed-address addressing, the slaves are addressed via the station address distributed by the master in the start-up phase. In the EtherCAT bus topology in the »PLC Designer«, the first slave gets the address 1001, the second slave the address 1002, etc. The EtherCAT addresses cannot be changed. The EtherCAT address of the master is 0. Master objects with this address can also be accessed via CoE.

Auto-increment addressing
The auto-increment addressing is based on the bus topology. Each slave can be addressed by means of its physical position within the fieldbus.
Slave 1 = address 0
Slave 2 = address -1
Slave 3 = address -2 etc.
The master transmits a telegram to the slave address. Each slave increments the address during the telegram cycle. A slave to whom a telegram is addressed recognizes the telegram by means of the address 0.

Explicit device identification:
This mechanism is necessary if the device is part of a 'Hot Connect'-group, to prevent cable swapping or the device is used within a modular machine application of Lenze.
To identify the device explicitly one of the following mechanisms can be used within the i-series devices:

- A master can use the 'Configured station alias' address of the slave. For the i-series this is possible by setting the relevant register within the configuration EPM with direct access from the master.
- Setting an identification value with the rotary switch. With a value >0 this value is valid and used to identify the device.
- If the switch is set to 0 the identification value can be set with the parameter 0x2361:4 (P510:4). A value >0 explicitly identifies the device.

The active device identification value is displayed in the parameter 0x2362:4 (P511:4). The identification value (/ Station alias) has to be unique in the network. Using one of the identification mechanisms the master references the devices with these values.

See chapter 2.2 Rotary Selector (EtherNet/IP, EtherCAT only) on page 7 for details about rotary switch.

P508:0	0x2360:0	Activate network (*)		i550	R/W
0: No action/no error 1: Restart with actual values 10: In progress 11: Action cancelled 12: Fault		Controls network activities			
P	**0x1000:0**	**Device type (*)**		**i550**	**R**
-- ... **[Actual value]** ... --		Device type			
P	**0x1008:0**	**Manufacturer device name (*)**	**i510**	**i550**	**R**
-- ... **[Actual value]** ... --		Manufacturer device name			
P	**0x1009:0**	**Manufacturer HW version (*)**		**i550**	**R**
-- ... **[Actual value]** ... --		Manufacturer Hardware revision			
P	**0x100A:0**	**Manufacturer software version (*)**	**i510**	**i550**	**R**
-- ... **[Actual value]** ... --		Manufacturer Software revision			
P	**0x1018:1**	**Vendor ID (*)**	**i510**	**i550**	**R**
-- ... **[Actual value]** ... --		Object identification: Revision: Vendor ID Lenze is 0x0000003B = 59			
P	**0x1018:2**	**Product code (*)**	**i510**	**i550**	**R**
-- ... **[Actual value]** ... --		Object identification: Product Code			
P	**0x1018:3**	**Revision (*)**	**i510**	**i550**	**R**
-- ... **[Actual value]** ... --		Object identification: Revision			
P	**0x1018:4**	**Serial number (*)**	**i510**	**i550**	**R**
-- ... **[Actual value]** ... --		Object identification: Serial number of the device. Note that it does not represent the Lenze serial number (0x2000:2)			

P509:0	0x2363:0	Switch setting (*)		i550	R
-- ... **[Actual value]** ... --		Displays the current rotary switch setting. The active identification is displayed in 0x2362:4.			
P510:4	**0x2361:4**	**Device identifier (*)**		**i550**	**R/W**
-- ... **[0]** ... --		Set the explicit device identification value of the device with an object. (valid if rotary switches are set to 0)			

5.3 Diagnostics

Active settings can be checked in the Parameters listed below:

P511:4	**0x2362:4**	**Active device identifier (*)**		**i550**	**R**
-- ... **[Actual value]** ... --		Displays the active explicit device identifier value			
P511:6	**0x2362:6**	**Active station address (*)**		**i550**	**R**
-- ... **[Actual value]** ... --		Display the active EtherCAT station address of the device			
P511:7	**0x2362:7**	**Active Tx length (*)**		**i550**	**R**
-- ... **[Actual value]** ... --		Display the length of transmitted cyclic data in bytes			
P511:8	**0x2362:8**	**Active Rx length (*)**		**i550**	**R**
-- ... **[Actual value]** ... --		Display the length of received cyclic data in bytes			
P516:0	**0x2368:0**	**Network status (*)**		**i550**	**R**
1: Initialisation 2: Pre-Operational 3: Bootstrap 4: SafeOperational 8: Operational		Display the communication status of the device			
P517:0	**0x2369:0**	**Bus error (*)**		**i550**	**R**
-- ... **[Actual value]** ... --		Indicates a bus error			

5.4 Errors and Error Reaction

P515:1	**0x2859:1**	**Watchdog elapsed (*)**		**i550**	**R/W**
2:Trouble (Reference see P310:1)		Fault reaction for fieldbus : Watchdog expired			
P515:3	**0x2859:3**	**Invalid confg by Master (*)**		**i550**	**R/W**
2:Trouble (Reference see P310:1)		Fault reaction for fieldbus : Invalid Configuration by Master			
P515:4	**0x2859:4**	**Stack initialization error (*)**		**i550**	**R/W**
2:Trouble (Reference see P310:1)		Fault reaction for fieldbus : Fieldbus stack initialization error			
P515:5	**0x2859:5**	**Invalid process data (*)**		**i550**	**R/W**
2:Trouble (Reference see P310:1)		Fault reaction for fieldbus: Invalid process data			

5.5 Cyclic Data Access

Process data are transmitted by means of EtherCAT datagrams via the process data channel.

General:

- The i500 is controlled by means of the process data.
- The transmission of process data is time-critical.
- Process data are transferred cyclically between the Controller (EtherCAT master) and the inverters (slaves) (Continuous exchange of current input and output data).
- The master can directly access the process data. In the PLC for instance, the data are directly stored in the I/O area.
- The max. length of the process data per direction is 32 bytes (8bit/byte)
- Process data are not saved in the i500.
- Content of the process data is defined with the mapping
- Process data are for instance setpoints, actual values, control words, and status words.

Configuration:

- All sync managers are configured for cyclic and mailbox communication. Configuration displayed in 0x1C00:1-4.
- For the communication the data configuration has to be set up. Use the assignment objects 0x1C12 and 0x1C13
- Basic setup for the sync manager is done in the objects 0x1C32 and 0x1C33
- For the assignment the available objects for configuration include velocity mode and free configuration are used. Choose the mapping objects 0x1603 and 0x1A03 for the fixed setup velocity mode and the objects 0x1605 and 0x1A05 to select a dynamic mapping. The content can be chosen from all mappable objects.

RPDO:

P	**0x1603:1**	**(vl) RPDO-->A: Object 1 (*)**		**i550**	**R**
-- ... **[0x60400010]** ... --		Mapped to Control word CiA402 (0x6040)			
P	**0x1603:2**	**(vl) RPDO-->A: Object 2 (*)**		**i550**	**R**
-- ... **[0x60420010]** ... --		Mapped to Target velocity CiA402 (0x6042)			
P	**0x1605:1-16**	**(User) RPDO-->A: Free config. Object 1-16**		**i550**	
-- ... **[0]** ... --					

TPDO:

P	**0x1A03:1**	**(vl) A-->TPDO: Object 1 (*)**		**i550**	**R**
-- ... **[0x60410010]** ... --		Mapped to Status word CiA402 (0x6041)			
P	**0x1A03:2**	**(vl) A-->TPDO: Object 2 (*)**		**i550**	**R**
-- ... **[0x60440010]** ... --		Mapped to Actual velocity CiA402 (0x6044)			
P	**0x1A03:3**	**(vl) A-->TPDO: Object 3 (*)**		**i550**	**R**
-- ... **[0x603F0010]** ... --		Mapped to Error Code (0x603F)			
P	**0x1A05:1-16**	**(User) A-->TPDO: Free config. object 1-16**		**i550**	
-- ... **[0]** ... --					

Sync Manager:

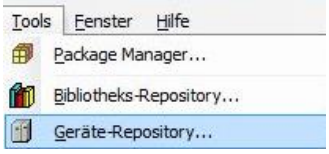
P	0x1C00:1	SM1: Communication type (*)		i550	R
1: Receive mailbox					
P	0x1C00:2	SM2: Communication type (*)		i550	R
2: Transmit mailbox					
P	0x1C00:3	SM3: Communication type (*)		i550	R
3: Transit process data					
P	0x1C00:4	SM4: Communication type (*)		i550	R
4: Receive process data					

P	0x1C12:1	SM2 (RPDO-->Device): Assignment PDO 1 (*)		i550	
-- ... **[0x1603]** ... --					
P	0x1C12:2	SM2 (RPDO-->Device): Assignment PDO 2 (*)		i550	
-- ... **[--]** ... --					
P	0x1C13:1	SM3 (RPDO-->Device): Assignment PDO 1 (*)		i550	
-- ... **[0x1A03]** ... --					
P	0x1C13:2	SM3 (RPDO-->Device): Assignment PDO 2 (*)		i550	
-- ... **[--]** ... --					

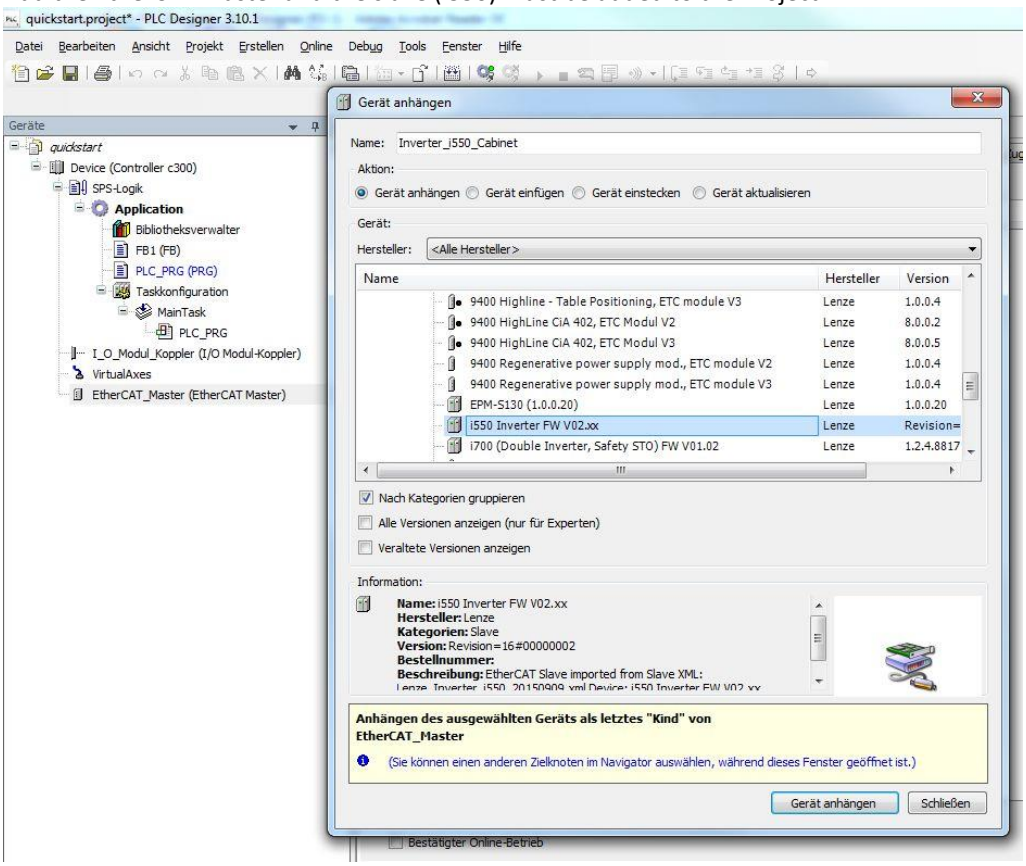
P	0x1C32:1	SM2 (RPDO-->Device): Sync type (*)		i550	R/W
0: Free Run					
P	0x1C32:2	SM2 (RPDO-->Device): Cycle time (*)		i550	R
-- ... **[Actual value]** ... -- ns					
P	0x1C32:3	SM2 (RPDO-->Device): Shift time (*)		i550	R
-- ... **[Actual value]** ... -- ns					
P	0x1C32:4	SM2 (RPDO-->Device): Supported sync types (*)		i550	R
-- ... **[Actual value]** ... --					
P	0x1C32:5	SM2 (RPDO-->Device): Min. cycle time (*)		i550	R
-- ... **[Actual value]** ... -- ns					
P	0x1C33:1	SM3 (TPDO-->Device): Sync type (*)		i550	R/W
0:Free Run (Reference see I0x1C32:1)					
P	0x1C33:2	SM3 (TPDO-->Device): Cycle time (*)		i550	R
-- ... **[Actual value]** ... -- ns					
P	0x1C33:3	SM3 (TPDO-->Device): Shift time (*)		i550	R
-- ... **[Actual value]** ... -- ns					
P	0x1C33:4	SM3 (TPDO-->Device): Supported sync types (*)		i550	R
-- ... **[Actual value]** ... --					
P	0x1C33:5	SM3 (TPDO-->Device): Min. cycle time (*)		i550	R
-- ... **[Actual value]** ... -- ns					

5.6 EtherCAT quick start (Lenze Controller)

1. For network control the following setup needs to be done:
   - Select CiA velocity mode for operation, 0x6060:0 = Velocity mode (VL) [2]
   Save the parameter with 0x2022:3 (P700:3).
2. To use the i550 device over EtherCAT the ESI-file needs to be installed.



3. Add the EtherCAT master and the slave (i550) must be added to the Project.



4. The automatic configuration of the identifier of the slave can be used. (Alternatively an individual identifier can be set with Rotary Switch / Parameter)
5. The default configuration in 0x1C12:1 and 0x1C13:1 selects velocity mode with mapping objects 0x1603 and 0x1A03.The fixed setup for velocity mode (CiA402) is:

PLC → Inverter

RPDO A: Object1	0x6040:0	CiA402 Controlword
RPDO A: Object2	0x6042:0	CiA402 Target velocity

Inverter → PLC

TPDO A: Object1	0x6041:0	CiA402 Statusword
TPDO A: Object2	0x6044:0	CiA402 Velocity actual value
TPDO A: Object3	0x603F:0	Error Code

6. Download configuration and start the communication

See chapter 7.1 CIA402 on page 48 for details about control/status word

5.7 Connection to Engineering Tools

The Lenze Controller provides a gateway function to establish an online connection to a field device via EtherCAT.

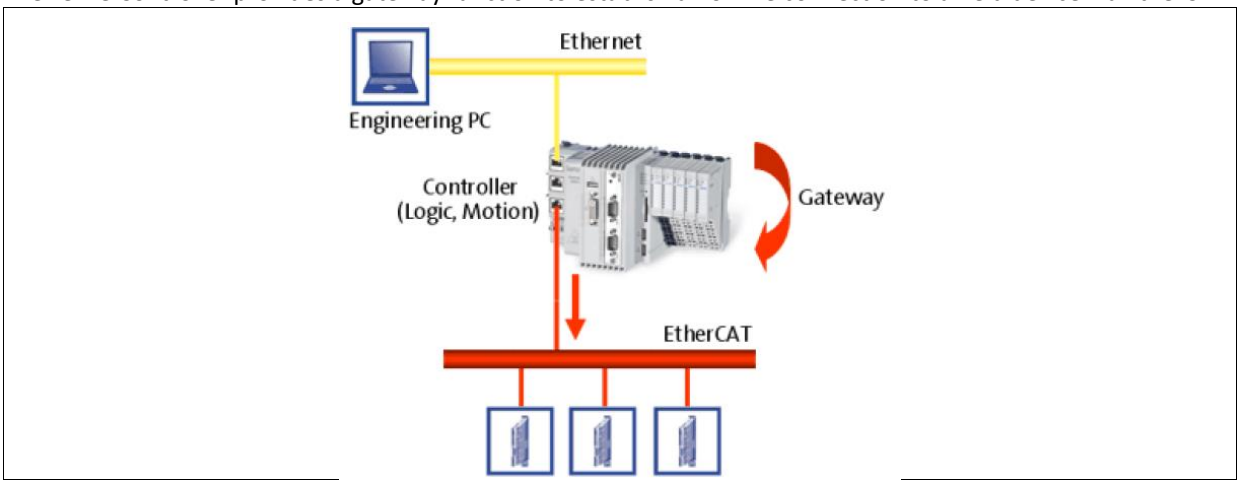
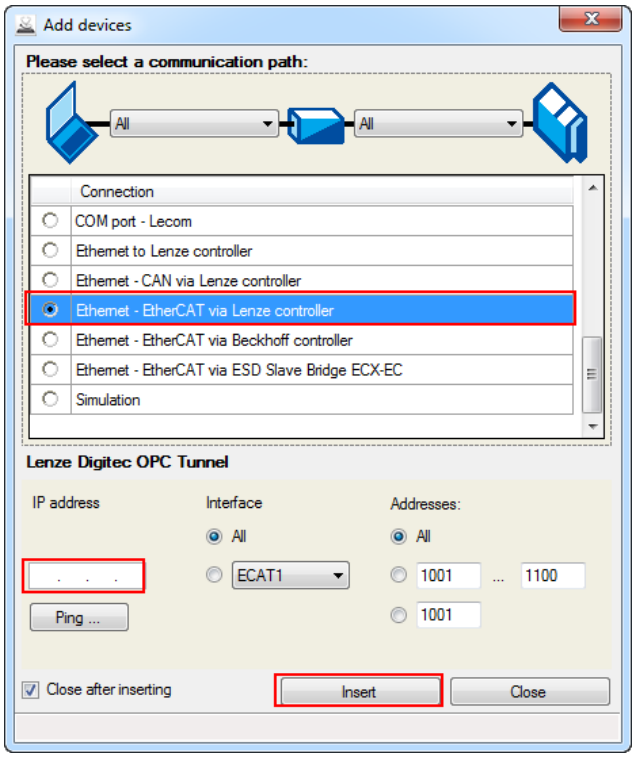


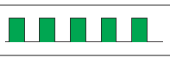
Fig. 4: Example: EtherCAT bus system with a Lenze Controller 3231 C as gateway

1. Before the Controller can be used as gateway the inverters must be defined in the controller first. To do so the inverters have to be added in the PLC designer. Afterwards the project has to be loaded in the controller. (See 5.6 EtherCAT quick start (Lenze Controller) on page 24)
2. Start Easy starter and select the connection “Ethernet – EtherCAT via Lenze controller.
   Enter the IP address of the controller. (By clicking the Ping button, you can carry out a simple test which verifies whether the controller can actually be reached via the IP address set)
   Click “Insert”



5.8 Troubleshooting

5.8.1 EtherCAT LED status display

LED	Status	Meaning
RUN	Off	EtherCAT status "Init"
		EtherCAT state "Pre-Operational"
		EtherCAT state "Safe-Operational"
		EtherCAT state "Operational"
		EtherCAT state "Bootstrap"
L/A	Off	No EtherCAT connection
		EtherCAT communication active
		EtherCAT connection available

LED	Status	Meaning
ERROR	Off	No Error
	Single Flash	Local Error, devices state change to SafeOp
	Double Flash	Sync Manager watchdog timeout
		EtherCAT invalid configuration

5.8.2 EtherCAT specific messages

Error Code (Hex)	Error Code (Dec)	Fault Type	Tool text	Description
0x8190	33168	Configurable (0x2859:1)	Watchdog timeout	EtherCAT watchdog expired (Communication connection detected a timeout)
0x8286	33414	Configurable (0x2859:3)	PDO map error	EtherCAT PDO mapping error (fault with mapping data, wrong configuration or objects selected)
0x8192	33170	Configurable (0x2859:4)	Communic.error	EtherCAT communication stack initialization error (wrong SM settings received)
0x8193	33171	Configurable (0x2859:5)	Inv cycl data	EtherCAT invalid cyclic data (internal mapping deactivated during operation)

6 EtherNet/IP

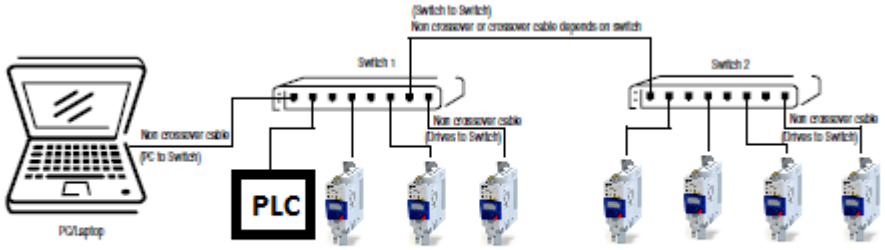
EtherNet/IP is supported from SW Version 02.01

Lenze implementation of CIP follows the standard and supports the two main types of EtherNet/IP communication: Explicit Messaging (Parameter data transfer) and Implicit Messaging (Cyclic data transfer)

6.1 General Considerations

6.1.1 Network topology

Typically, an EtherNet/IP network is made up of segments containing point-to-point connections in a star configuration.



MultiCast Configuration
By default the iSeries drive automatically generates the multicast address used for I/O messaging. The default multicast I/O packets will be propagated over the local subnet only. The user is allowed to explicitly set the drive's multicast address and TTL values but this feature should be used carefully. The TTL and Mcast Config attributes in the TCP/IP object are also implemented.

6.1.2 Supported services

Feature	Inverter Drive
IGMP Implementation	IGMP v2
TCP/IP Sockets	Add. TCP/IP socket connections supported
CIP Connections	up to 8
Cabling	STP (ISO/IEC 11801 or EN 50173) CAT 5e
Half duplex / full duplex Baud rate	10 / 100 MBit/s
Auto detection of data rates (auto negotiation)	✓
Supported baudrates: 10 BaseT, 100 BaseT	✓
Scalable amount of input and output process data words.	✓

Parameter access data channel	✓
AC DRIVE profile	✓
Minimum Fieldbus cycle time (Request Packet Interval - RPI)	4 ms
Fieldbus cycle time steps	4-128 msec
Pass through time Frame to axis ms	<= 6 ms
Port to Port latency	<= 125 us

Feature	**Inverter Drive**
Internet Protocol (IP version 4) (RFC 791)	✓
DLR (Device Level Ring Protocol)	✓
DLR supervisor mode	-
User Datagram Protocol (UDP) (RFC 768)	✓
Transmission Control Protocol (TCP) (RFC 793)	✓
Address Resolution Protocol (ARP) (RFC 826)	✓
Internet Control Messaging Protocol (ICMP) (RFC 792)	✓
ACD (Address Conflict Detection)	✓
QoS (Quality of Service)	✓
IEEE 802.3 (Ethernet) (RFC 894)	✓

6.2 Configuration

Each node on the network must have an individual address, if two or more nodes have duplicate addresses the network will not function correctly. After changing this setting, the drive must be power cycled or the EtherNet/IP module must be reset in order for the new IP address to take effect.

See chapter 2.2 Rotary Selector (EtherNet/IP, EtherCAT only) on page 7 for details about rotary switch.

P508:0	**0x23A0:0**	**Activate network (*)**		**i550**	**R/W**
0: No action/no error 1: Restart with actual values 2: Restart with default values 5: Stop network communication 10: In progress 11: Action cancelled 12: Fault		Controls Network activities			
P510:1	**0x23A1:1**	**IP address (*)**		**i550**	**R/W**
-- ... **[192.168.124.16]** ... --		EtherNet/IP address settings			

P510:2	0x23A1:2	Subnet (*)		i550	R/W
-- ... **[255.255.255.0]** ... --		EtherNet/IP Subnet mask			
P510:3	**0x23A1:3**	**Gateway (*)**		**i550**	**R/W**
-- ... **[0.0.0.0]** ... --		EtherNet/IP Gateway address			
P510:4	**0x23A1:4**	**Host name (*)**		**i550**	**R/W**
-- ... **[--]** ... --		Setting the host name Note: As DNS is not supported the hostname only serves to identify the device.			
P510:5	**0x23A1:5**	**IP configuration (*)**		**i550**	**R/W**
0: Stored IP **1: BOOTP** 2: DHCP		Specifies the way getting/acquiring the used Network address configuration.			
P510:6	**0x23A1:6**	**Multicast TTL (*)**		**i550**	**R/W**
-- ... **[1]** ... --		Multicast TTL By default mechanism, this MC IP address is calculated by Ethernet/IP algorithm from defined unicast IP address. If desired, this can be set manually using the following drive parameters.			
P510:7	**0x23A1:7**	**Multicast allocation control (*)**		**i550**	**R/W**
0: Default Allocation Algorithm 1: NumMcast/Mcast start addr		Allocation Control			
P510:8	**0x23A1:8**	**Mcast IP address (*)**		**i550**	**R/W**
-- ... **[239.64.2.192]** ... --		IP Address setting for Multicast			
P510:9	**0x23A1:9**	**Multicast number (*)**		**i550**	**R/W**
1 ... **[1]** ... 8					
P510:10	**0x23A1:10**	**Timeout (*)**		**i550**	**R/W**
500 ... **[10000]** ... 65535 ms		Ethernet timeout in ms			
P512:1	**0x23A4:1**	**Port 1 enum (*)**		**i550**	**R/W**
0: Auto-Negotiation 1: 10Mbps 2: 100Mbps 5: 10Mbps/Half Duplex 6: 10Mbps/Full Duplex 7: 100Mbps/Half Duplex 8: 100Mbps/Full Duplex		Ethernet Port 1 Settings			
P512:2	**0x23A4:2**	**Port 2 enum (*)**		**i550**	**R/W**
0: Auto-Negotiation 1: 10Mbps 2: 100Mbps 5: 10Mbps/Half Duplex 6: 10Mbps/Full Duplex 7: 100Mbps/Half Duplex 8: 100Mbps/Full Duplex		Ethernet Port 2 Settings			
P513:0	**0x23A6:0**	**Quality of service (*)**		**i550**	**R**
0: 802.1Q Tag disable 1: 802.1Q Tag enable		Enable/disable VLAN tagging 802.1Q			
P514:0	**0x23A7:0**	**Address conflict detection (*)**		**i550**	**R/W**
0: Disabled **1: Enabled**		Enable or disable address conflict detection			

6.3 Diagnosis

Active settings can be checked in the Parameters listed below:

P511:1	**0x23A2:1**	**Active IP address (*)**		**i550**	**R**
-- ... **[Actual value]** ... --		Current EtherNet/IP address settings			
P511:2	**0x23A2:2**	**Active subnet (*)**		**i550**	**R**
-- ... **[Actual value]** ... --		Current EtherNet/IP Subnet mask			
P511:3	**0x23A2:3**	**Active Gateway (*)**		**i550**	**R**
-- ... **[Actual value]** ... --		Current EtherNet/IP Gateway Address			
P511:5	**0x23A2:5**	**MAC Address (*)**		**i550**	**R**
-- ... **[Actual value]** ... --		MAC ID			
P511:6	**0x23A2:6**	**Active Mcast address (*)**		**i550**	**R**
-- ... **[Actual value]** ... --		Current Multicast address settings			
P509:0	**0x23A3:0**	**DIP switch position (*)**		**i550**	**R**
-- ... **[Actual value]** ... --		Displays the setting of the EtherNet/IP module's address switch. 0:Software Setting from IP configuration 1-254:IP address - XXX.XXX.XXX.[SWITCH] 255:Factory Default of communication objects			
P519:1	**0x23A5:1**	**Port 1 display (*)**		**i550**	**R**
0: Not connected 1: 10Mbps/Half Duplex 2: 10Mbps/Full Duplex 3: 100Mbps/Half Duplex 4: 100Mbps/Full Duplex		Display port 1 status			
P519:2	**0x23A5:2**	**Port 2 display (*)**		**i550**	**R**
0: Not connected 1: 10Mbps/Half Duplex 2: 10Mbps/Full Duplex 3: 100Mbps/Half Duplex 4: 100Mbps/Full Duplex		Display port 2 status			
P516:0	**0x23A8:0**	**CIP module status (*)**		**i550**	**R**
-- ... **[Actual value]** ... --		Display of current CIP module status info			
P517:0	**0x23A9:0**	**CIP network status (*)**		**i550**	**R**
-- ... **[Actual value]** ... --		Display of current CIP network status			

6.4 Error and Error Reaction

Implicit Messaging Timeout
Class 1 connections are monitored by watchdog timeout given in the forward open command while establishing a connection. Reaction is defined in 0x2859:1.

Explicit Messaging Timeout
To support controllers that are not capable of class 1 connections the iSeries drive supports the watchdog on class 3 connection messages and in addition an overall EtherNet/IP timeout on unconnected services 0x23A1:10. For a connected class 3 connection the timeout is configured by the client, the reaction within the drive can be set in 0x2859:6.

P515:1	0x2859:1	Watchdog elapsed (*)		i550	R/W
2:Trouble (Reference see P310:1)					
P515:3	**0x2859:3**	**Invalid confg by Master (*)**		**i550**	**R/W**
2:Trouble (Reference see P310:1)					
P515:4	**0x2859:4**	**Stack initialization error (*)**		**i550**	**R/W**
2:Trouble (Reference see P310:1)					
P515:5	**0x2859:5**	**Invalid process data (*)**		**i550**	**R/W**
2:Trouble (Reference see P310:1)					
P515:6	**0x2859:6**	**Timeout explicit message (*)**		**i550**	**R/W**
1:Warning (Reference see P310:1)					
P515:7	**0x2859:7**	**Communication timeout (*)**		**i550**	**R/W**
1:Warning (Reference see P310:1)					

Logix Controller PROG Mode Support
Device supports RXLogic Prog Mode Support.

6.5 Supported Objects

The drive contains EtherNet/IP Assembly Object Instances that pertain to the following Scanner Connection Parameters:

- Input (actual values such as actual velocity, actual position, etc.)
- Output (enable and reference value going to the drive)
- Configuration

The terms Input and Output refer to the point of view of the scanner device. Output data is produced by the scanner and consumed by the adapter. Input data is produced by the adapter and consumed by the scanner. The drive is an adapter device (to be configured as a AC Drive Adapter in Scanner. All Assembly instances are accessible using Class 3 explicit messages as well as the Class 1 I/O messaging).

The Ethernet Link Object provides the following common services:

- 0x0E Get_Attribute_Single
- 0x10 Set_Attribute_Single

All parameters are able to be modified using Explicit Messaging.
The Set Attribute Single message instruction can be utilized for each parameter that is to be changed over Explicit Messaging.

i	Instance corresponds to the parameter's Index number and Attribute corresponds to drive parameter's subindex number. In the event the desired drive parameter has no index the user may set the attribute = 1.

Assembly Object

Class Code		0x04
Class Attributes	Revision	2
	Max Instance	see below
Class Services	Get Attribute Single	
Class Instances	Input	see below
	Output	see below
	Configuration	130
Instance Services		see below

Following AC Drive Profile supporting objects and respective attribute are implemented on iSeries drives:

Class Attribute services

Service	Service Name	Data Type
1	Get Revision	Data Type = UNIT
2	Get Max. Instance	Data Type = UNIT
3	Get Number of Instances	Data Type = UNIT
4	Get Number Attributes	Data Type = UNIT
5	Get Optional Attributes	Data Type = UNIT
6	Get Max. ID Number Class Attributes	Data Type = UNIT
7	Get Max. ID Number Instance Attributes	Data Type = UNIT

Instance Attribute services

Service	Service Name	Data Type
1	Get No. of Member in List	Data Type = UNIT
2	Get Member List	Data Type = UNIT
3	Get/Set Data	Data Type = UNIT
4	Get Size	Data Type = UNIT

6.5.1 I/O Assemblies

Output assemblies are commonly used for controlling the enable/disable state of the drive and for supplying the velocity or torque reference. Input assemblies are commonly used to monitor the drive status and runtime quantities such as current velocity.

i500 Ethernet/IP implementation supports the I/O assembly object class 0x04. iSeries drives assemblies are static. There are several Input and Output pre-defined assemblies (assembly object instances) that can be used for data exchange. The terms Input and Output refer to the point of view of the scanner. Output data is produced by the scanner and consumed by the adapter. Input data is produced by the adapter and consumed by the scanner. The iSeries drives are always an adapter device. Depending on the assembly number the memory map of the data can have a different size and meaning.

Input Assembly Instance
The Input Assembly Instances supported, in accordance to the CIP Network Library include: Instance 70, 71, 73 and 110.

Input Assembly 73:

Instance	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
73	0	At Reference	Ref From Net	Ctrl From Net	Ready	Running 2 (Rev)	Running 1 (Fwd)	Warning	Faulted
	1	Drive State							
	2	Speed Actual (low byte)							
	3	Speed Actual (high byte)							
	4	Torque Actual (low byte)							
	5	Torque Actual (high byte)							

Important Note on Input Assemblies:
Input assemblies (adapter to scanner) are mapped to the adapter memory from byte 0. There is no preceding 4 byte header like that found in most Allen-Bradley equipment. The inverter does not use preceding header functionality for real time status. So the start address in the assembly memory map is the actual start of the 1st assembly data item. The user should supply the actual assembly length when mapping the input assembly to the controller memory.

Output Assembly Instance
Important Note on Output Assemblies:
Output assemblies (scanner to adapter) are assumed to have a preceding 4 byte header. When mapping the assembly this header will automatically be added to the data stream by most AB PLC/CLC equipment. If using equipment other than an AB for the scanner, the user will have to configure it to send the 4 byte header preceding the actual assembly data. The data in the header should be set to 0. The Output Assembly Instances supported, in accordance to the CIP Network Library include: Output Assembly 21, 22, 23 and 110.

Output Assembly 23:

Instance	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
23	0		NetRef	NetCtrl			Fault Reset	Run Rev	Run Fwd
	1								
	2	Speed Reference (low byte)							
	3	Speed Reference(high byte)							
	4	Torque Reference (low byte)							
	5	Torque Reference (high byte)							

Note:
If bit5 NetCtrl is 1 and Network Enable 0x2631:37 = 114 (Network ControlEnableRequest.Bit): All bits of this control word are processed.
If bit5 NetCRl is 0 or Network Enable 0x2631:37 is not asserted: Control bits 0, 1, 12, 13, 14, 15 are NOT processed; their states are ignored and the drive is in local control with functions triggered by settings in 0x2631 (P400)

6.5.2 Identity Object

The Identity Object provides a known, public interface to a device's identification and general data. There must be at least one instance of this object for each device.
The Identity Object instances parameters that may be read over Explicit Messaging. The iSeries drive shall support the following Identity Object parameters:

Class Code		0x01
Class Attributes	Revision	1
	Max Instance	1
	Number Instance	
	Max ID# class attribute	
	Max ID# instance attribute	
Class Services	Get Attribute Single	
Instance Attributes	Vendor ID	Lenze
	Device Type	2 (AC Drive)
	Product Code	550
	Revision	1
	Major	1
	Minor	1
	Product Name	"IOFW51AGXX"
Instance Services	Get Attribute All	
	Get Attribute Single	
	Reset	

Table 1: Identity Object

6.5.3 Compatibility with Other Products

The i500 EtherNet/IP module is a CIP slave adapter and can be controlled by any CIP Generic master capable of either class 1 or class 3 messaging.

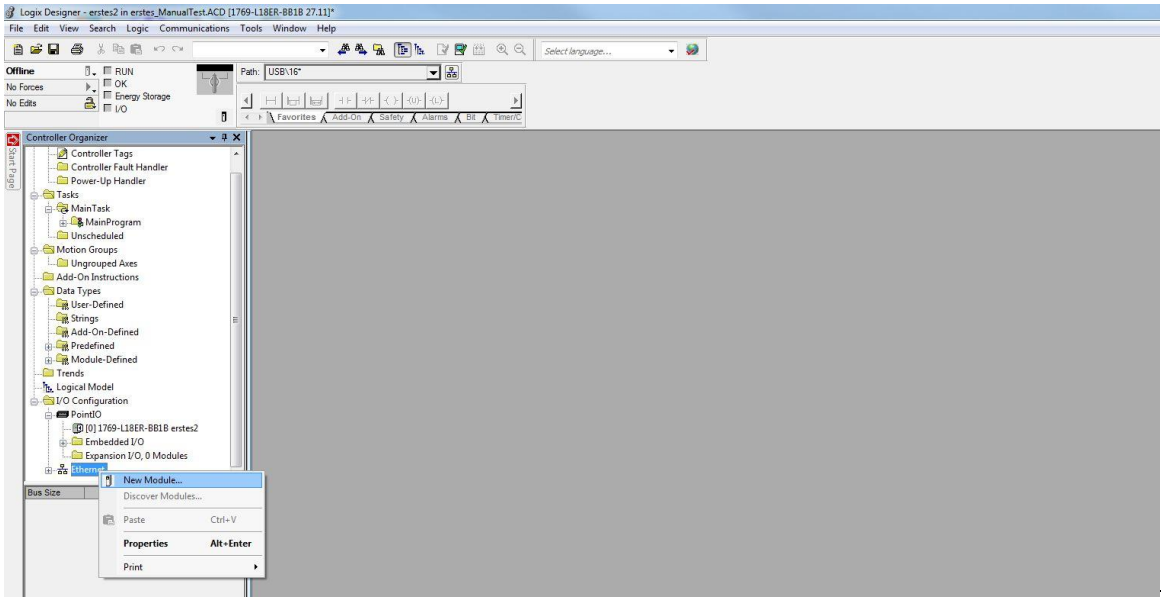
6.6 EtherNet/IP quick start

1. Set the IP address, Subnet and Gateway of the inverter with Parameter 0x23A1:XX and the Rotary Switch
2. For network control the following setup needs to be done:
   - Set 0x2631:37 (P400:37)"Network enable" to P400:37 to 114 Network ControlEnableRequest.Bit"
   - Set 0x2860:1 (P201:1) "Default frequency setpoint" to "Network Frequency Setpoint" [5].
3. Save the parameter with 0x2022:3 (P700:3) and power cycle the drive completely that the configuration takes effect.
4. Configuring the Network Master
   Master Support Files
   Some EtherNet/IP master configuration software utilizes Electronic Data Sheet (eds) files to configure the network profile and communications with the relevant devices. Some use this file for automatic tag generation. The iSeries eds file is available on the Lenze website.
   Note: AB controllers do not require the eds file to add the drive to their configuration.
5. For cyclic data exchange / acyclic data exchange refer to the sections above.

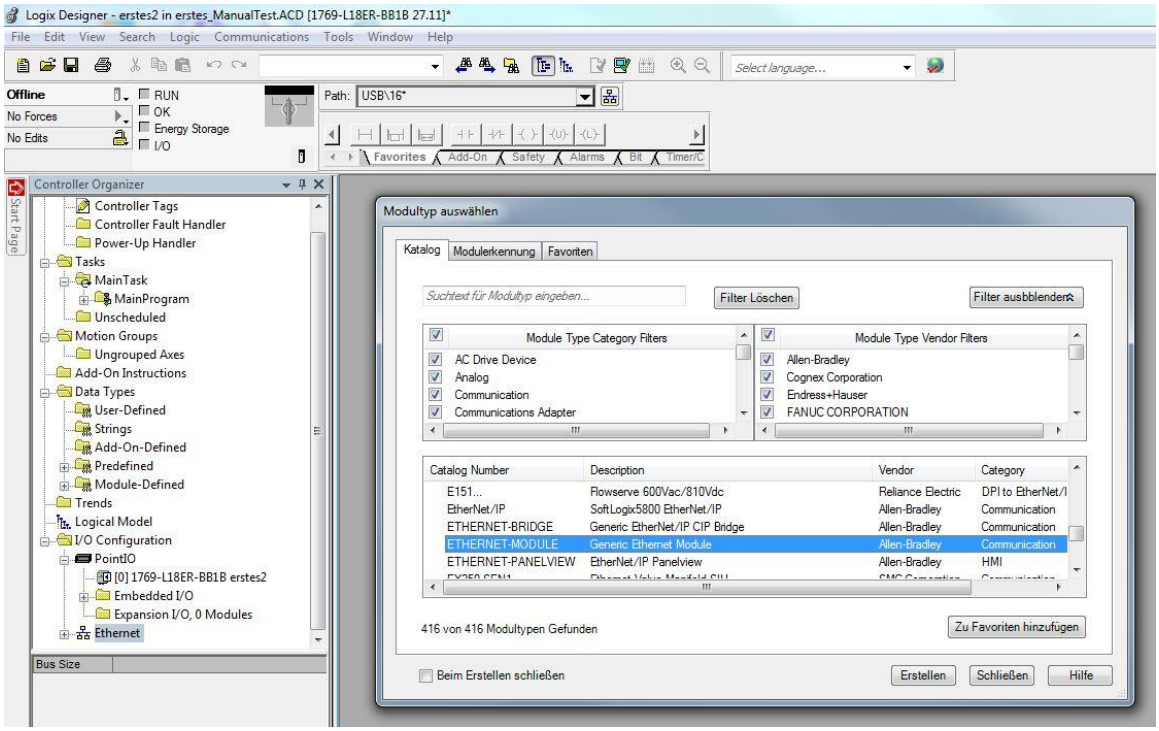
6.7 Cyclic Data Access (Implicit Messaging)

To map the drive to an EtherNet/IP scanner in RSLogix 5000 for implicit messaging
Implicit Messaging:

1. Click the [I/O Configuration] folder in the left-hand navigation window
2. Click the appropriate Ethernet Port folder, [1769-L32E Ethernet Port] in this example.
3. Right click on the [Ethernet] network icon and select [New Module] under the Ethernet port of the Controller's EtherNet/IP scanner.
   Note: The project in RSLogix 5000 must be in the offline state to add the drive to the configuration.



4. Select "ETHERNET MODULE Generic Ethernet Module" and click "Create".



5. For this example we will use Output Assembly 23 to control the drive and Input Assembly 73 to read status from the drive.

Output Assembly 23:

Output Assembly 23:

Instance	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
23	0		NetRef	NetCtrl			Fault Reset	Run Rev	Run Fwd
	1								
	2	Speed Reference (low byte)							
	3	Speed Reference(high byte)							
	4	Torque Reference (low byte)							
	5	Torque Reference (high byte)							

Input Assembly 73:

Instance	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
73	0	At Reference	Ref From Net	Ctrl From Net	Ready	Running 2 (Rev)	Running 1 (Fwd)	Warning	Faulted
	1	Drive State							
	2	Speed Actual (low byte)							
	3	Speed Actual (high byte)							
	4	Torque Actual (low byte)							
	5	Torque Actual (high byte)							

Note:

If bit5 NetCtrl is 1 and Network Enable 0x2631:37 = 114 (Network ControlEnableRequest.Bit): All bits of this control word are processed.

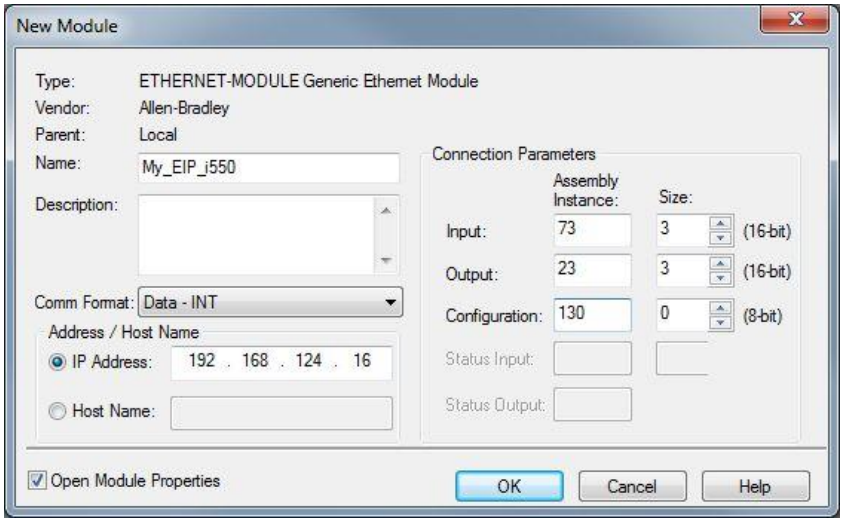
If bit5 NetCRl is 0 or Network Enable 0x2631:37 is not asserted: Control bits 0, 1, 12, 13, 14, 15 are NOT processed; their states are ignored and the drive is in local control with functions triggered by settings in 0x2631 (P400)

6. Enter a name for the drive -usually relating to the process (i.e. booster_pump_4, or an equipment tag number such as PP105).
7. Enter the IP address of the drive. Ensure that it is on the same subnet as the PLC (the first 3 octets of the IP address match).
8. Enter “Data – INT” for the Comm format.
   Enter the desired Input and Output Assembly numbers and their corresponding lengths. Remember the size must be set to the number of words that actually make up the assembly you want to use.
   For Configuration enter assembly instance 130 and a size of 0. This value is required.

Note:

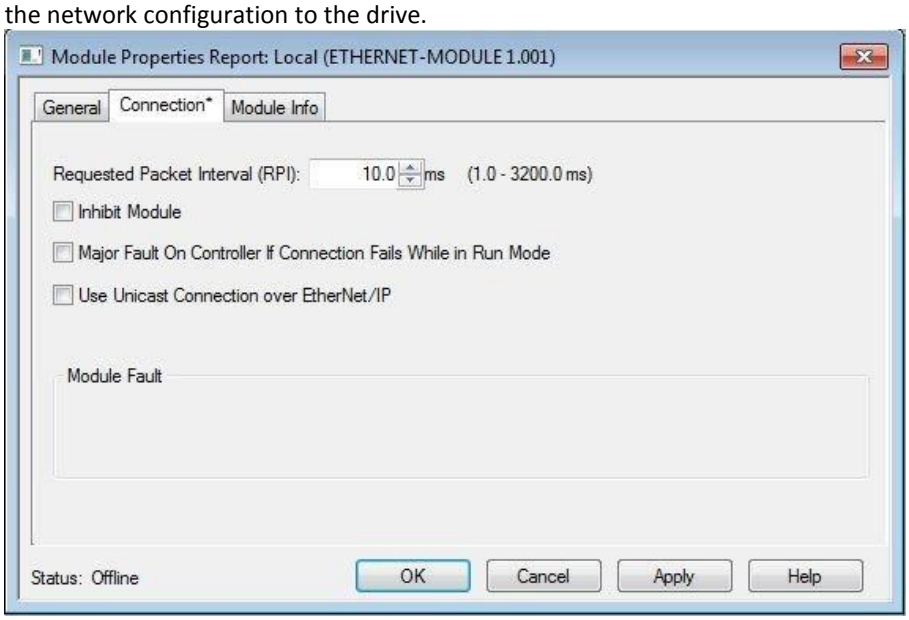
Bits 5 and 6 of byte 0 in assembly 23 MUST be asserted for the drive to accept control and speed reference commands from the network!

9. In this example we use "My_EIP_i550". Next change the Comm Format to "Data- INT". This is necessary as the data in the assemblies of this example are in 16 bit integer words. Enter the drive’s IP address. Finally, enter "73" for the Input Assembly and size = "3". Enter "23" for the Output Assembly also with size = "3" and enter "130" as the Configuration Assembly with size = "0". Now click "OK"

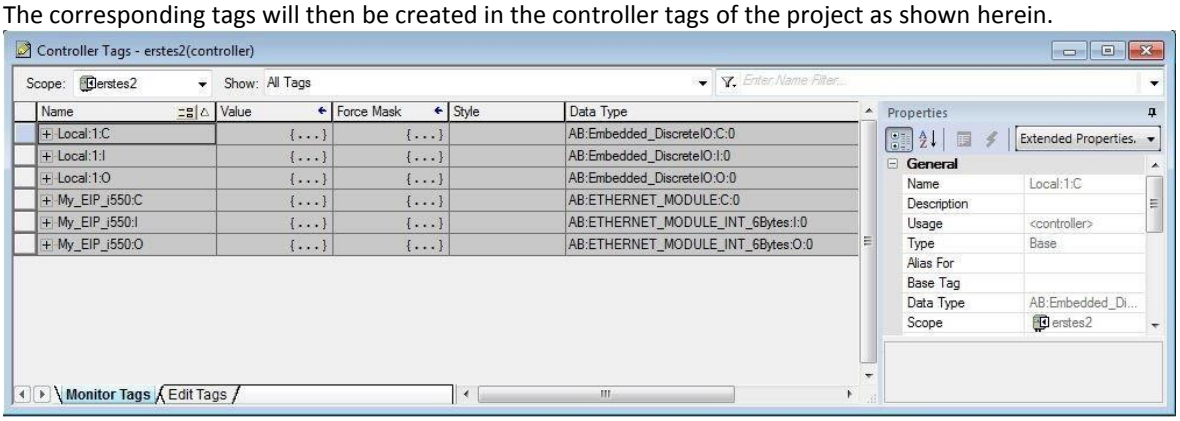


Under the connection tag enter the desired RPI rate. This is how frequently the drive will be polled by the PLC. The minimum value is 4.0 milliseconds.

10. From this screen you can also optionally set the controller to fault if the EtherNet/IP connection is lost to the drive while the controller is running. This selection is the [Major Fault On Controller If Connection Fails While in Run Mode].
11. Now for this example: deselect "Use Unicast Connection over Ethernet/IP". This feature is new for RSLogix™ 5000 version 20 and not present in previous revisions. The iSeries drive does support Unicast Connection over EtherNet/IP and usage of this feature will result in overall faster network performance; however usage of this feature can make troubleshooting difficult without configuration efforts in managed switches. For this example, we will keep the default Request Packet Interval (RPI) of 10.0 milliseconds as for most inverter applications it is not necessary to poll the drive more frequently than 10.0 milliseconds. Select OK to finish the network configuration to the drive.

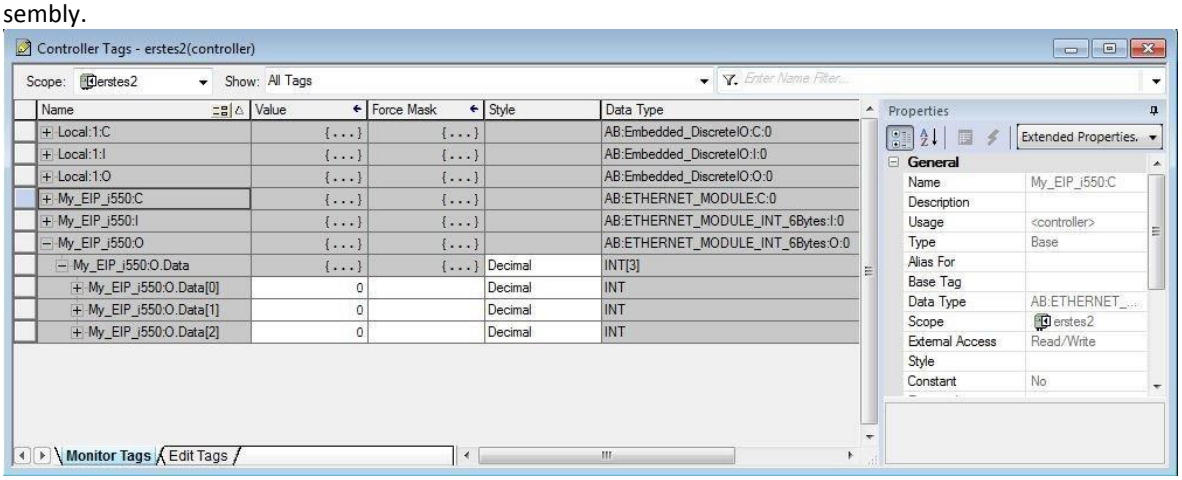


12. The corresponding tags will then be created in the controller tags of the project as shown herein.

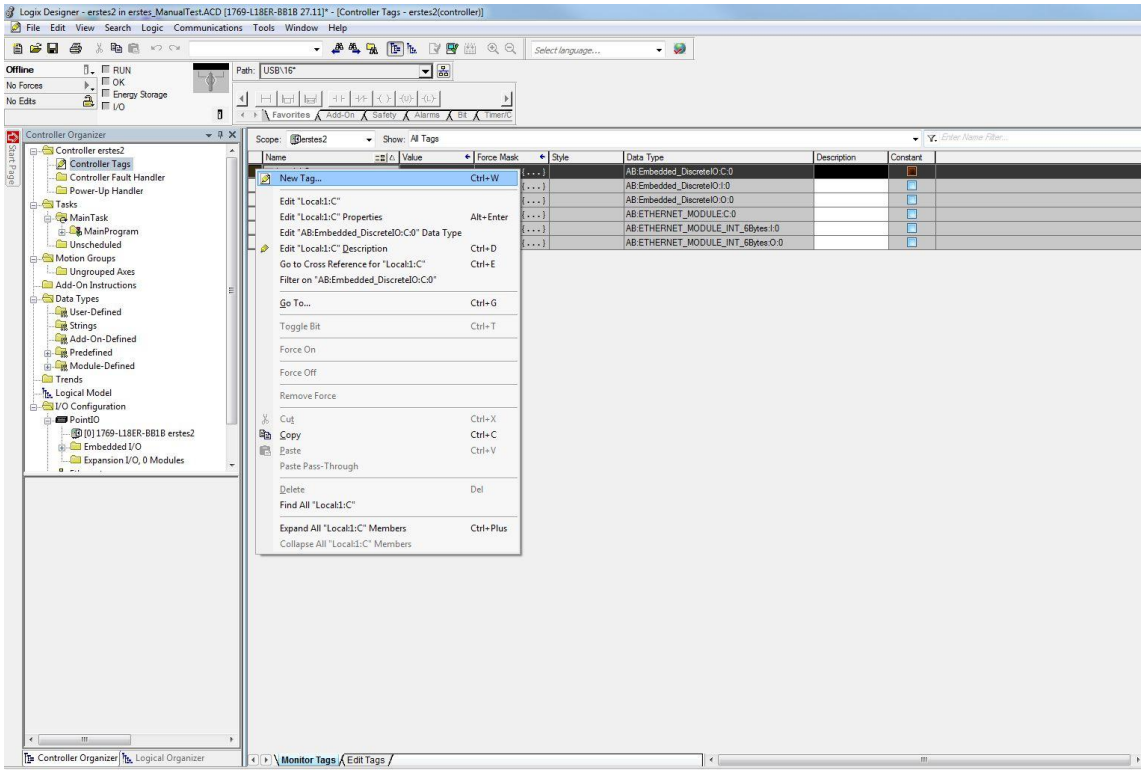


To understand this from the above configuration we named the drive “My_i550_Drive”. There are three sets of tags labeled “My_i550_Drive”:
[:C] for the Configuration assembly (1)
[:I] for the Input Assembly (101 in this example)
[:O] for the Output assembly (100 in this example)

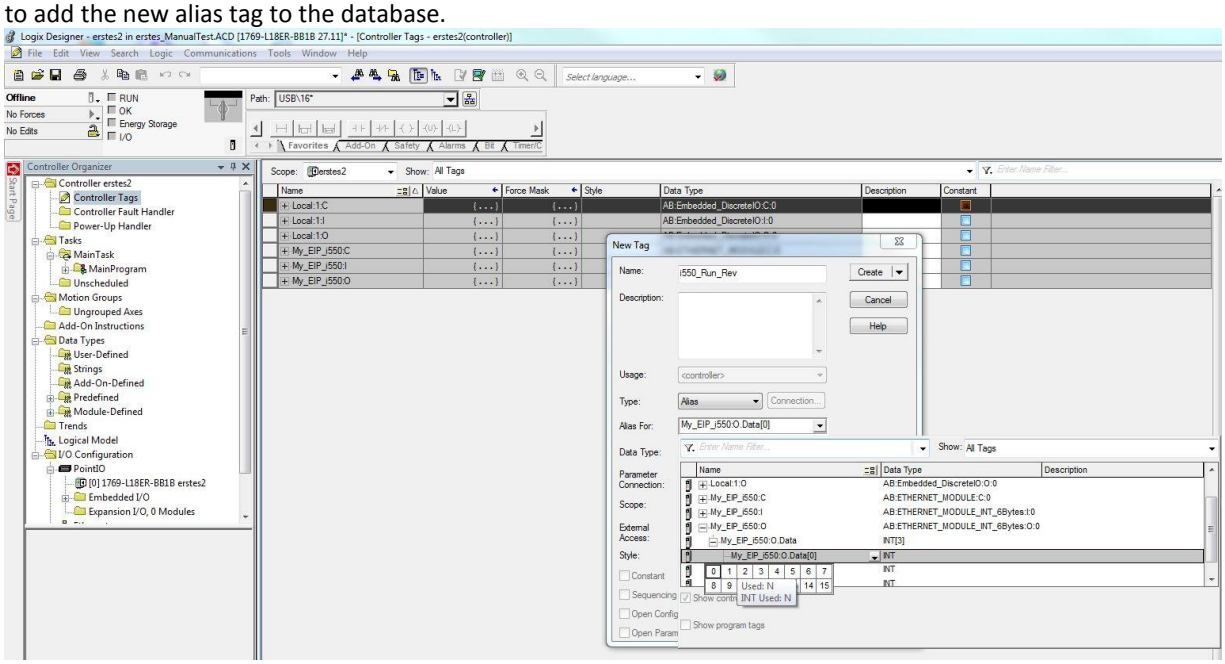
13. Click on the [+] and expand the [My_i550_Drive:O] data to reveal all four words that make up the Output assembly.



14. The example RSLogix™ 5000 project contains aliasing for all bits in the assemblies. The process below is the basic workflow to create an alias tag in RSLogix™ 5000. To begin, select "Controller Tags" from the navigation tree. Now right click next to any tag and select "New Tag…”



15. Enter a meaningful name for the tag. Change the Type to "Alias". Then in the Alias For field browse to the desired bit or word. In this example, Run Rev is the drive output assembly word 0 bit 1. Finally, select "Create" to add the new alias tag to the database.



16. Implicit Messaging Timeout
Class 1 connections are monitored by watchdog timeout given in the forward open command while establishing a connection. It is often desirable to set a fault timeout condition to prevent the drive from operating in a runaway condition or loss of communications. In case the class#1 connection expires, the defined reaction in 0bject 0x2859 raised.

Saving the Configuration in the Controller

After adding the scanner (or bridge) and the adapter to the I/O configuration, the configuration must be downloaded to the controller. The configuration should also be saved to a file on your computer.

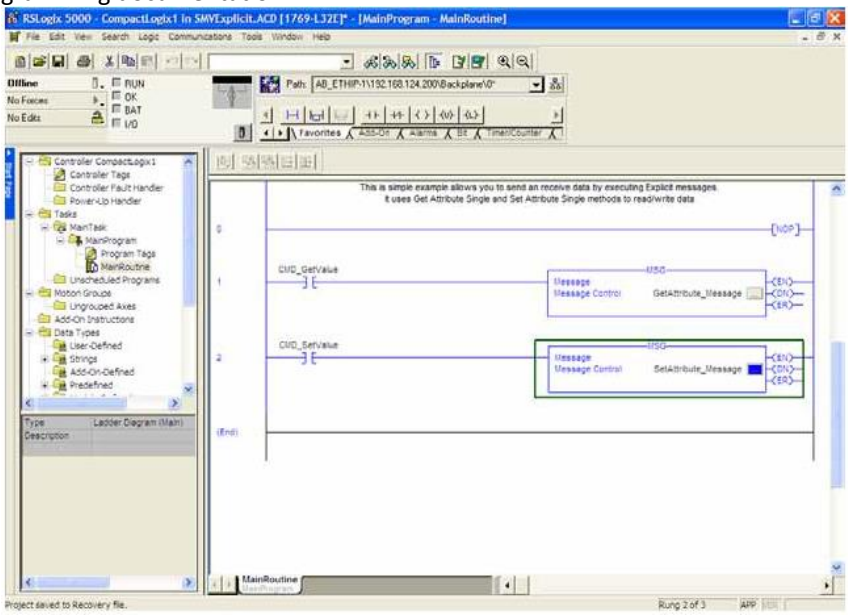
1. On the top toolbar, click [Communications] then select [Download] from the pull down menu. The Download dialog box will open.
2. Click [Download] to download the configuration to the controller. When the download is successfully completed, RSLogix enters online mode and the I/O OK box in the upper-left part of the screen is green.
3. On the top toolbar, click [File] then select [Save] from the pull down menu. If this is the first time the project is saved, then the [Save As] dialog box will open. Navigate to a folder, type a file name and then click [Save] to save the configuration to a file on your computer.

6.8 Acyclic Data Access (Explicit Messaging)

- Acyclic / non-cyclic / Service access provides a method for the network master to access any drive or module parameter.
- This kind of parameter access is typically used for monitoring or low priority non-scheduled parameter access. However it can also be used to control the drive by writing assembly data.
- The iSeries drive EtherNet/IP module supports several different methods of doing this.

Explicit Messaging:

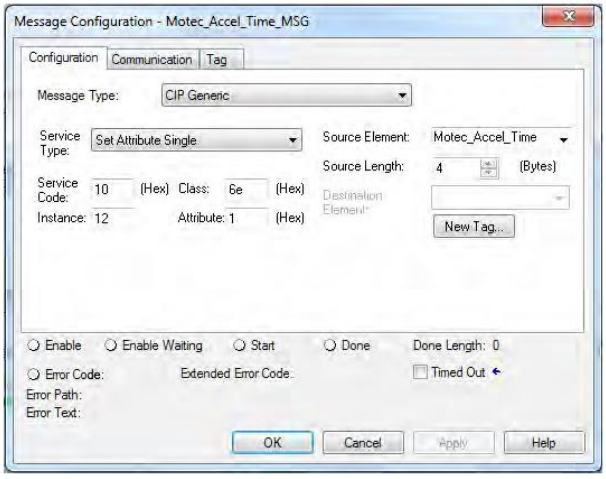
An explicit message is a logical instruction in the PLC's program used for messaging. It can be used to read/write to either a parameter setting or an assembly's data. In the case of CompactLogix, ControlLogix and SoftLogix the MSG instruction provides the capabilities described in this section. For other PLC types, consult that PLC's programming documentation.



General drive variables (parameters and subindexes) are in class 0x6e. The Instance is the parameter's index number and the attribute is the subindex number. If there is no subindex, set attribute to 0. A attribute of 1 is supported for those clients that don't support an attribute value of 0. These variables are all data type SINT (8 bit/ 1 byte objects), INT (16 bit/2 byte objects), or DINT (32 bit/4 byte objects) and must have the source length set to coincide. You can check the length of the object in the drive’s programming manual:

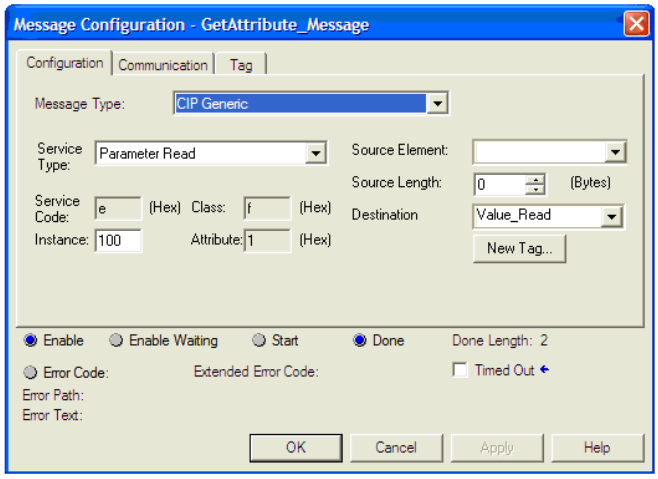
To write a parameter value to the iSeries drive using EtherNet/IP explicit messaging set the following:

Message Type = CIP Generic
Class = 6e (Hex)
Attribute = parameter's subindex number (or 1 if no subindex)
Service Code = 10 (Parameter Write)
Instance = the parameter's index number in the drive
Source Element = the variable in the PLC used as the source of the data for a write
For a write of a parameter value set the source length = the length of the drive parameter (in bytes)



To read a parameter value from the iSeries drive using EtherNet/IP explicit messaging set the following:

Message Type = CIP Generic
Class = 6E (Hex)
Attribute = parameter's subindex number (or 1 if no subindex)
Service Code = e (Parameter Read)
Instance = the parameter's index number in the drive
Destination = the target variable data from the drive will be copied to in the PLC
Make sure the tag used as the destination is the same format and length as the drive parameter.



CIP generic master

For CIP generic masters that cannot support class 1 (implicit) messaging and are limited to class 3 explicit messaging it is possible to write/read the assembly data via explicit messaging.
To write assembly data to the i500 using EtherNet/IP explicit messaging set the following:

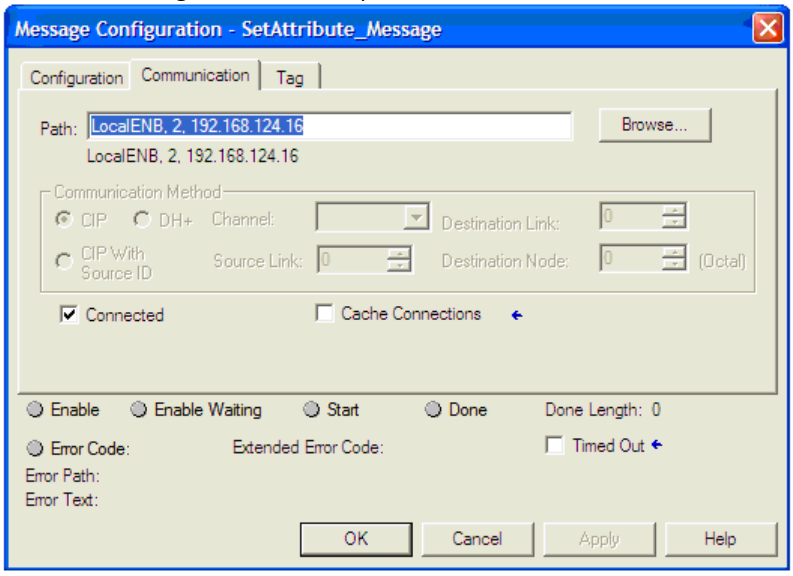
Message Type = CIP Generic
Class = 4 (Hex)
Attribute = 3

Service Code = 10 (Set Attribute Single)
Instance = the assembly number in the drive desired (i.e. Assembly 23 would be 23)
Source Element = variable in the PLC used as the source of the data for a write (must be in INT format)
When writing an Assembly, set the source length equal to the same number of bytes contained in the desired assembly (i.e., Assembly 23 contains 3 words which equals 6 bytes).

To read assembly data from the iSeries drive using EtherNet/IP explicit messaging set the following:
Message Type = CIP Generic
Class = 4 (Hex)
Attribute = 3
Service Code = e (Get Attribute Single)
Instance = the assembly number in the drive desired (i.e. Assembly 73 would be 73)
Destination = the target variable data from the drive will be copied to in the PLC
Make sure the tag used as the destination is an array in INT format with the same length as the desired assembly.

Explicit Message Path

For any explicit message the path will need to be set to route the message out the controller's Ethernet port to the IP address of the Drive. This path will differ with the PLC used. Consult the PLC manufacturer for assistance with determining how to set this path if needed.



Explicit Messaging Timeout

It is often desirable to set a fault timeout condition to prevent the drive from operating in a runaway condition. To do this while controlling the drive via explicit messaging, set Overall Ethernet Timeout Time 0x23A1:10 and reaction accordingly 0x2859:7.

Torque scale (Class 2a, instance 1, attribute 18) is related to actual torque command by the following equation: Torque reference in Scaled Torque = Nm * 2TorqueScale. By setting TorqueScale = 0, Torque Reference (Assembly 23 word 2) is actual NM commanded torque (=NM *20= NM*1=NM). Loading a value of 2 to Motec_Torque_Reference thereby commands a torque limit from the drive of 2 NM.
Drive_Mode (Class2a, instance1, attribute 6) has only two valid settings: "3" for Torque Mode, or "1" for Velocity Mode.
Both Drive Mode and TorqueScale are AC drive profile objects. They are defined in the CIP library. These objects are in class 2a with the Instance =1, and the attribute number as listed above. These variables are all data type SINT and must have the source length set to 1 byte.

Torque control mode is not implemented in inverter SW version 2.0

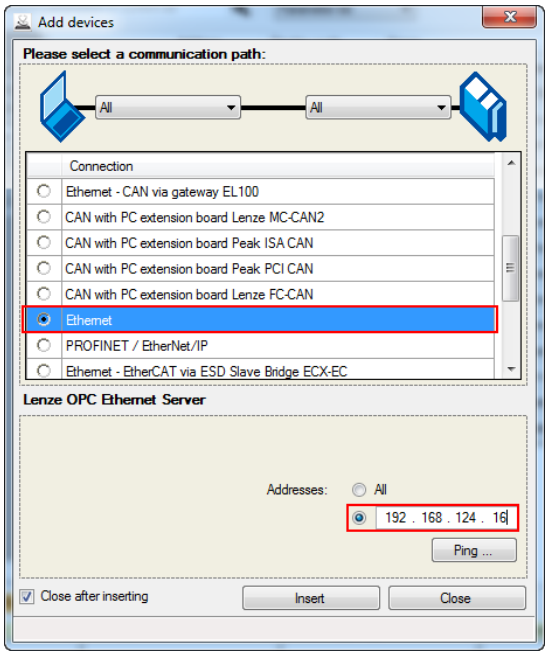
6.9 EtherNet/IP Connection to Engineering Tools

1. Power the device externally either by mains or external 24V supply and connect the inverter to the PC using the USB diagnostic adapter.
2. Set address and station name:
   The station name, the IP address, subnet mask, and gateway address can either be assigned by the PLC-controller via EtherNet/IP or set manually in the Easy Starter.

   If the settings are done with Easy Starter set the following parameters:
   a. Communication parameters:

Name	Parameter	Default:
IP Address	0x23A1:1 (P510:1)	192.168.124.16
Subnet	0x23A1:2 (P510:2)	255.255.255.0
Gateway	0x23A1:3 (P510:3)	0.0.0.0

   Save the parameter with 0x2022:3 (P700:3) and power cycle the drive that the configuration takes effect.
3. Connect the PC to the i500 EtherNet/IP with an Ethernet cable.
4. Set the IP address of the network adapter in your PC to the same range than the inverter IP address.
   Example:
   IP Address 192.168.124.1
   Subnet 255.255.255.0
   Select the connection “Ethernet” in the Easy Starter
   Set the correct IP address of the device and add it. Now the objects of the inverter are accessible in Easy Starter.



6.10 Troubleshooting

6.10.1 EtherNet/IP LED status display

The indication of module status is a single bicolor (red/green) indicator that represents the state of the entire product per the standard CIP definition.

LED	Colour / status		Description
MS	**Green**	**Red**	
	Off	Off	CIP module status: "Nonexistent" The communication module is not being supplied with voltage.
	Off	On	 CIP module status: "Major Unrecoverable Fault" The communication module has a fault that cannot be rectified. The status is set if the pending status determining device error shows the "System fault" response.
	Off	Blinking	 CIP module status: "Major Recoverable Fault" The communication module has a fault that can be rectified. The status is set if the pending status determining device error shows the "Fault", "Trouble", "Quick stop by trouble", "Warning locked", or "Warning" response.
	On	Off	 CIP module status: "Operational" The communication module is working perfectly.
	Blinking	Off	 CIP module status: "Standby" The communication module has not been completely configured or the configuration is defective.
	Blinking	Blinking	 CIP module status: "Device Self Testing" The communication module is currently undergoing a self-test.

LED	Colour / status		Description
NS	Green	Red	
	Off	Off	CIP network status: "No IP Address" The communication module is not being supplied with voltage or has not been given an IP address.
	Off	On	 CIP network status: "Duplicate IP" The communication module is unable to gain access to the fieldbus (IP address conflict).
	Off	Blinking	 CIP network status: "Connection Timeout" A time-out is executed.
	On	Off	 CIP network status: "Connected" The communication module is working perfectly and has established a connection to the scanner.
	Blinking	Off	 CIP network status: "No Connections" The communication module ... • is working correctly; • has been assigned an IP address; • has not been integrated into the network by the scanner yet.
	Blinking	Blinking	 CIP network status: "Self-Test" The communication module is currently undergoing a self-test.

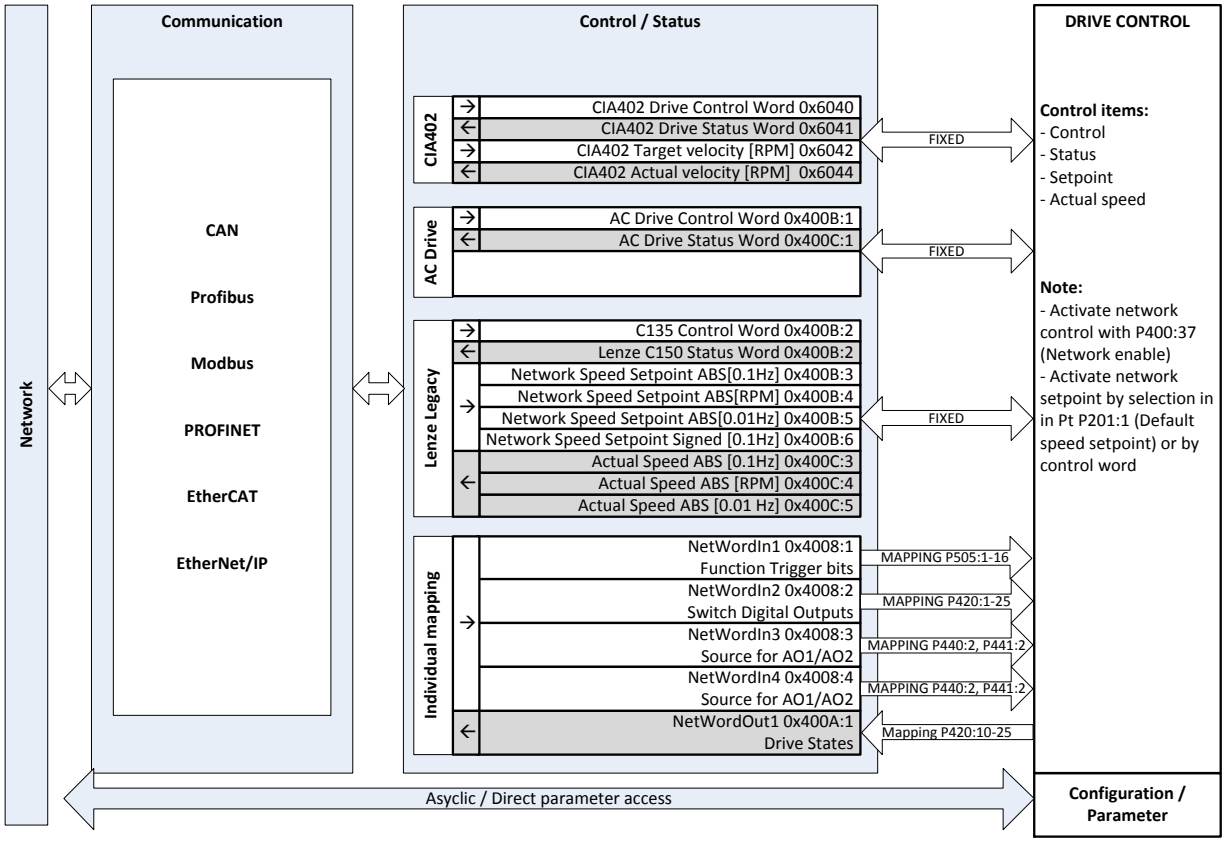
6.10.2 EtherNet/IP specific messages

Error Code (Hex)	Error Code (Dec)	Fault Type	Tool text	Description
0x8190	33168	Configurable (0x2859:1)	Watchdog timeout	EtherNet/IP watchdog expired (Communication connection detected a timeout)
0x8286	33414	Configurable (0x2859:3)	PDO map error	EtherNet/IP PDO mapping error (fault with I/O data configuration)
0x8192	33170	Configurable (0x2859:4)	Net. Init. error	EtherNet/IP communication stack initialization error (wrong setting during initialization)
0x8193	33171	Configurable (0x2859:5)	Inv cycl data	EtherNet/IP invalid cyclic data (exchanged cyclic data is static, not updated)
0x8112	33042	Configurable (0x2859:6)	TO expl. msg	EtherNet/IP timeout explicit message (parameter communication connection expired)
0x8114	33044	Configurable (0x2859:7)	TO overall comm	EtherNet/IP overall communication timeout

7 Drive Profile

Several command words, status and setpoint are available to control the drive from remote:

- CIA402 (Predefined mapping) Used for EtherCAT/CAN
- AC drive profile (Predefined mapping) Used for EtherNet/IP
- Lenze Legacy (Predefined mapping)
- Network IN/Out (Individual mapping)



7.1 CIA402

This chapter describes the CIA402 Format

> **i** This Format is normally used for EtherCAT and CAN

> **i** With default settings the Lenze-State machine is active. For full compatibility with CIA402 state machine the CiA402 mode needs to be selected in 0x6060. (Selection: [2] “Velocity mode CIA402”)
> For detailed description about CiA402 state machine refer to CiA402 documents.

7.1.1 Control word

0x6040 CIA402 Drive Control Word

Bit	Function	Note
0	Switch on	0 = Switch inverter OFF 1 = Switch inverter ON
1	Enable voltage	0 = Disable Voltage 1 = Enable Voltage
2	Activate quick stop	0 = Quick stop active 1 = Quick stop not active
3	Enable operation	0 = Controller inhibit 1 = No controller inhibit
4	Operation mode specific	
5	Operation mode specific	
6	Operation mode specific	
7	Fault reset	Transition from 0 to 1 resets fault
8	n/a	
9	Operation mode specific	
10	Reserved	
11	Reserved	
12	Reserved	
13	Reserved	
14	Holding brake release	1 = Releases holding brake
15	Reserved	

7.1.2 Status word

0x6041:0 P780:0 CIA402 Drive Status Word

Bit	Function	Note
0	Ready to switch on	
1	Switched on	
2	Operation enabled	
3	Fault active	
4	Voltage enabled	
5	Quick stop	0 = Quick stop active 1 = Quick stop not active
6	Switch on disabled	
7	Warning active	
8	Deactivate RPDOs	
9	Remote	Network control mode active
10	Target reached	Target speed reached
11	Internal limit active	Internal limit of speed setpoint active
12	Reserved	
13	Reserved	
14	Brake released	
15	STO not active	

7.1.3 Speed setpoint / Actual Speed

P781:0	0x6042:0	Target velocity vl	i510	i550	R/W
-- ... **[0]** ... -- rpm		CiA402 network speed setpoint			

P783:0	0x6044:0	Velocity actual value vl	i510	i550	R
-- ... **[Actual value]** ... -- rpm		CiA402 actual speed			

7.2 Legacy Lenze Format

This chapter describes the Legacy Lenze Format.

This Format was used in other Lenze products such as 8200Vector and 8400

7.2.1 Control word C135

0x400B:2 P592:2 Lenze Legacy C135 control word

Bits	Function	Comments
0	Setpoint Selection bit 0	Bits 0,1 decoding: 0 = Flexible (Default setpoint is active) 1 = Preset Setpoint #1 2 = Preset Setpoint #2 3 = Preset Setpoint #3
1	Setpoint Selection bit 1	
2	Rotation (0-CW/1-CCW)	
3	Activate quick stop	0 = Not Active 1 = Active
4	Reserved	
5	Reserved	
6	Reserved	
7	Reserved	
8	Reserved	
9	Disable (1-active/0-inactiv)	0 = Controller released 1 = Controller Inhibited
10	Network user fault	
11	Fault reset (0→1)	0->1 edge causes TRIP reset
12	Reserved	
13	Reserved	
14	DC Brake Active	0 = Not Active 1 = Active
15	Reserved	

7.2.2 Status word

0x400C:2 P593:2 Lenze Legacy C150 status word

Bits	Function	Comments
0	Active Parameter Set	0 = Parameter Set 1 or 3 active 1 = Parameter Set 2 or 4 active
1	Power stage inhibited	0 = enabled 1 = inhibited
2	Current/torque limit reached	Current limit reached Torque limit reached (in Torque mode)
3	Frequency setpoint reached	
4	Ramp generator	Input = Output
5	Below Frequency threshold	Below Frequency threshold Qmin Index 0x4005 (f < 0x4005) Qmin
6	Actual frequency = 0	
7	Inhibit (1-activ/0-inactiv)	0 = Drive Enabled 1 = Drive Inhibited
8	Encode status bit 0	0000 = Initializing 0001 = Mains Voltage Off 0010 = Switch n Inhibited 0011 = Operation Inhibited 0100 = Flying Restart 0101 = DC Brake Active 0110 = Operation Enabled 0111 = Message Active 1000 = FAULT
9	Encode status bit 1	
10	Encode status bit 2	
11	Encode status bit 3	
12	Over-temperature warning	
13	DC Bus overvoltage	
14	Rotation (0-CW/1-CCW)	
15	Ready for Operation	

0x400C:5 P593:5 Lenze Legacy drive state

Bits	Function	Comments
0	Fault Locked	
1	Fault	
2	Start Pending	
3	Identification Not Done	
4	Inhibit	
5	Stop	
6	Switching On Sequence	
7	Identification in Progress	
8	Running	
9	Acceleration	
10	Deceleration	
11	Deceleration Override	
12	DC Brake	
13	Flying Start	
14	Current Limit	
16	Sleep Mode	

7.2.3 Speed setpoint / Actual Speed

Several speed command formats are also available:

P592:3	0x400B:3	Network frequency setpoint	i510	i550	R/W
0.0 ... **[0.0]** ... 599.0 Hz		Legacy network frequency setpoint Scaling: 0.1 Hz unsigned (direction information comes via control word)			
P592:4	**0x400B:4**	**Network setpoint speed**	**i510**	**i550**	**R/W**
0 ... **[0]** ... 50000 rpm		Legacy network speed setpoint Scaling: RPM unsigned (direction information comes via control word)			
P592:5	**0x400B:5**	**Network setp. frequency**	**i510**	**i550**	**R/W**
0.00 ... **[0.00]** ... 599.00 Hz		Legacy network frequency setpoint Scaling: 0.01 Hz unsigned (direction information comes via control word)			
P592:6	**0x400B:6**	**Network speed setpoint**	**i510**	**i550**	**R/W**
-599.0 ... **[0.0]** ... 599.0 Hz		Legacy network frequency setpoint Scaling: 0.1 Hz signed			

Several actual speed formats are also available:

P593:3	0x400C:3	Actual frequency Hz	i510	i550	R
-- ... **[Actual value]** ... -- Hz		Legacy actual frequency Scaling 0.1Hz, unsigned			
P593:4	**0x400C:4**	**Actual motor speed RPM**	**i510**	**i550**	**R**
-- ... **[Actual value]** ... -- rpm		Legacy actual speed Scaling RPM, unsigned			
P593:6	**0x400C:6**	**Actual frequency**	**i510**	**i550**	**R**
-- ... **[Actual value]** ... -- Hz		Legacy actual frequency Scaling 0.01Hz, unsigned			

7.3 AC Drive Profile

This chapter describes the AC Drive Format

> **i** This Format is normally used for EtherNet/IP

7.3.1 Control word

0x400B:1 P592:1 AC Drive control word

> **i** Some of the bits will be ignored if bit5 **NetCtrl** bit is not set, see table below for details

Bits	Function	Comments
0	Run forward (CW)	Run Forward - see transition table below for exact logic **NOTE: Bit processed only when NetCtrl = 1**
1	Run reverse (CCW)	Run Reverse - see transition table below for exact logic **NOTE: Bit processed only when NetCtrl = 1**
2	Fault Reset (0 -> 1)	Reset existing fault. Only on transition from 0->1
3	Reserved	
4	Reserved	
5	Control from Network (NetCtrl)	If bit5 NetCtrl is 1 and Network Enable 0x2631:37 = 114 (Network ControlEnableRequest.Bit): All bits of this control word are processed. If bit5 NetCRl is 0 or Network Enable 0x2631:37 is not asserted: Control bits 0, 1, 12, 13, 14, 15 are NOT processed; their states are ignored and the drive is in local control with functions triggered by settings in 0x2631 (P400)
6	Network setpoint source	If NetRef = 1 Network Setpoint becomes active drive setpoint. Network Setpoint could be speed, frequency, PID setpoint or Torque setpoint If NetRef = 0 Network Setpoint uses currently active setpoint coming from default configuration or set by 0x400E (P505)
7	Reserved	
8	Reserved	
9	Reserved	
10	Reserved	
11	Reserved	
12	Inhibit	**NOTE: Bit processed only when NetCtrl = 1**
13	Activate quick stop	**NOTE: Bit processed only when NetCtrl = 1**
14	PID off (1 – off)	**NOTE: Bit processed only when NetCtrl = 1**
15	DC Brake	**NOTE: Bit processed only when NetCtrl = 1**

7.3.2 Status word

0x400C:1 P593:1 AC Drive status word

Bits	Function	Comments
0	Fault/Trip	0 = No Fault 1 = Faulted
1	Warning active	
2	Running forward (CW)	0 = Not running Forward 1 = Running Forward
3	Running reverse (CCW)	0 = Not running Forward 1 = Running Forward
4	Ready	0 = NOT Ready 1 = Ready
5	Control from Network	0 = Local Control 1 = Network Control
6	Reference from Network	0 = Local Reference 1 = Network Reference
7	At Reference	0 = Setpoint not reached 1 = Setpoint reached
8	Profile State bit0	
9	Profile State bit1	
10	Profile State bit2	
11	Profile State bit3	
12	PID active	0 = PID NOT Active 1 = PID Active
13	Torque mode active	0 = NOT in Torque Mode 1 = Torque Mode Active
14	Current Limit reached	0 = NOT in Current Limit 1 = in Current Limit
15	DC Brake Active	0 = DC brake NOT active 1 = DC brake active

Drive State: Lenze CIA402 state machine to Ethernet/IP Drive State conversion table:

CiA402 Plus State	AC Drive Profile Drive State
INIT (0, 1)	0 - Vendor Specific
NOT_READY_TO_SWITCH_ON (2)	1 = Startup
SWITCH_ON_DISABLED (3)	2 = Not_Ready
READY_TO_SWITCH_ON (4) SWITCHED_ON (5)	3 = Ready
OPERATION_ENABLED (6)	4 = Enabled
DISABLE_OPERATION (7) SHUT_DOWN (8) QUICK_STOP (9)	5 = Stopping
FAULT_REACTION_ACTIVE (10)	6 = Fault_Stop
FAULT (11)	7 = Faulted

7.3.3 Speed setpoint / Actual Speed

Several speed command formats are also available:

P592:3	**0x400B:3**	**Network frequency setpoint**	**i510**	**i550**	**R/W**
0.0 ... **[0.0]** ... 599.0 Hz		Legacy network frequency setpoint Scaling: 0.1 Hz unsigned (direction information comes via control word)			
P592:4	**0x400B:4**	**Network setpoint speed**	**i510**	**i550**	**R/W**
0 ... **[0]** ... 50000 rpm		Legacy network speed setpoint Scaling: RPM unsigned (direction information comes via control word)			
P592:5	**0x400B:5**	**Network setp. frequency**	**i510**	**i550**	**R/W**
0.00 ... **[0.00]** ... 599.00 Hz		Legacy network frequency setpoint Scaling: 0.01 Hz unsigned (direction information comes via control word)			
P592:6	**0x400B:6**	**Network speed setpoint**	**i510**	**i550**	**R/W**
-599.0 ... **[0.0]** ... 599.0 Hz		Legacy network frequency setpoint Scaling: 0.1 Hz signed			

Several actual speed formats are also available:

P593:3	**0x400C:3**	**Actual frequency Hz**	**i510**	**i550**	**R**
-- ... **[Actual value]** ... -- Hz		Legacy actual frequency Scaling 0.1Hz, unsigned			
P593:4	**0x400C:4**	**Actual motor speed RPM**	**i510**	**i550**	**R**
-- ... **[Actual value]** ... -- rpm		Legacy actual speed Scaling RPM, unsigned			
P593:6	**0x400C:6**	**Actual frequency**	**i510**	**i550**	**R**
-- ... **[Actual value]** ... -- Hz		Legacy actual frequency Scaling 0.01Hz, unsigned			

7.4 NETword Configuration

Instead of using the predefined command and status word there are general NETWords can be configured.

> Depending on the fieldbus the mapping can be done in the Slave (inverter) or in the Master (PLC).
> Note: If the mapping is done in the Master (Example: PROFIBUS) the mapping in the Slave is overwritten!

Master → Inverter (NETWordIn)

- NETWordIn1: Function trigger bit
  Value: 0x4008:1 (P590:1)
  Configuration: 0x400E:1 (P505:1-16)

- NETWordIn2: Switch digital outputs/relay
  Value: 0x4008:2 (P590:2)
  Configuration: 0x2643:1-3 (P420:1-3)

- NETWordIn3: Source for AO1/AO2
  Value: 0x4008:3 (P590:3)
  Configuration AO1: 0x2639:2 (P440:2)
  Configuration AO1: 0x263A:2 (P441:2)

- NETWordIn4: Source for AO1/AO2
  Value: 0x4008:4 (P590:4)
  Configuration AO1: 0x2639:2 (P440:2)
  Configuration AO1: 0x263A:2 (P441:2)

Inverter → Master (NETWordOut)

- NETWordOut1: Drive Status bits
  Value: 0x400A:1 (P591:1)
  Configuration: 0x2635:10-25 (P420:10-25)

- NETWordOut2: Switched by Sequencer
  Value: 0x400A:2 (P591:2)
  Configuration: Sequencer parameter

7.4.1 NETWordIn configuration

Actual value:

P590:1	0x4008:1	NETWordIN1	i510	i550	R/W
--		Actual value of mappable network in word 1 bit collector (Trigger Function) --> Trigger mapping 0x400E1-16 (P505:1-16)			
P590:2	**0x4008:2**	**NETWordIN2**	**i510**	**i550**	**R/W**
--		Actual value of mappable network in word 2 bit collector (Trigger for digital Outputs) --> Trigger mapping 0x2634:1-3 (P420:1-3)			
P590:3	**0x4008:3**	**NETWordIN3**	**i510**	**i550**	**R/W**
0.0 ... **[0.0]** ... 100.0 %		Actual value of mappable network in word 3 (Source for Analog Output 1 and 2) --> Mapping 0x2639:2 (P440:2), 0x263A:2 (P441:2)			
P590:4	**0x4008:4**	**NETWordIN4**	**i510**	**i550**	**R/W**
0.0 ... **[0.0]** ... 100.0 %		Actual value of mappable network in word 3 (Source for Analog Output 1 and 2) --> Mapping 0x2639:2 (P440:2), 0x263A:2 (P441:2)			

Configuration:

P505:1	0x400E:1	NETWordIN1.00	i510	i550	R/W
0: Not connected 1: Inverter disabled 2: Stop 3: Quick stop 4: Reset fault 5: DC brake 8: Run forward (CW) 9: Run reverse (CCW) 13: Invert rotation 14: AI1 setpoint selection 15: AI2 setpoint selection 17: Network setpoint selection 18: Preset bit0 selection 19: Preset bit1 selection 20: Preset bit2 selection 21: Preset bit3 selection 39: Ramp 2 selection 40: Load parameter set 41: Parameter set 1 selection 42: Parameter set 2 selection 43: User-Netw. fault 1 44: User-Netw. fault 2 45: Process controller off 46: Set PID output to 0 47: PID integrator disabled 48: PID influence ramps active		Function of Network Input Word Bit 0			

P505:2	0x400E:2	NETWordIN1.01	i510	i550	R/W
0:Not connected (Reference see P505:0)		Function of Network Input Word Bit 1			
P505:3	**0x400E:3**	**NETWordIN1.02**	**i510**	**i550**	**R/W**
3:Quick stop (Reference see P505:0)		Function of Network Input Word Bit 2			
P505:4	**0x400E:4**	**NETWordIN1.03**	**i510**	**i550**	**R/W**
8:Run forward (CW) (Reference see P505:0)		Function of Network Input Word Bit 3			
P505:5	**0x400E:5**	**NETWordIN1.04**	**i510**	**i550**	**R/W**
13:Invert rotation (Reference see P505:0)		Function of Network Input Word Bit 4			
P505:6	**0x400E:6**	**NETWordIN1.05**	**i510**	**i550**	**R/W**
5:DC brake (Reference see P505:0)		Function of Network Input Word Bit 5			
P505:7	**0x400E:7**	**NETWordIN1.06**	**i510**	**i550**	**R/W**
0:Not connected (Reference see P505:0)		Function of Network Input Word Bit 6			
P505:8	**0x400E:8**	**NETWordIN1.07**	**i510**	**i550**	**R/W**
4:Reset fault (Reference see P505:0)		Function of Network Input Word Bit 7			
P505:9	**0x400E:9**	**NETWordIN1.08**	**i510**	**i550**	**R/W**
18:Preset bit0 selection (Reference see P505:0)		Function of Network Input Word Bit 8			
P505:10	**0x400E:10**	**NETWordIN1.09**	**i510**	**i550**	**R/W**
19:Preset bit1 selection (Reference see P505:0)		Function of Network Input Word Bit 9			
P505:11	**0x400E:11**	**NETWordIN1.10**	**i510**	**i550**	**R/W**
0:Not connected (Reference see P505:0)		Function of Network Input Word Bit 10			
P505:12	**0x400E:12**	**NETWordIN1.11**	**i510**	**i550**	**R/W**
0:Not connected (Reference see P505:0)		Function of Network Input Word Bit 11			
P505:13	**0x400E:13**	**NETWordIN1.12**	**i510**	**i550**	**R/W**
0:Not connected (Reference see P505:0)		Function of Network Input Word Bit 12			
P505:14	**0x400E:14**	**NETWordIN1.13**	**i510**	**i550**	**R/W**
0:Not connected (Reference see P505:0)		Function of Network Input Word Bit 13			
P505:15	**0x400E:15**	**NETWordIN1.14**	**i510**	**i550**	**R/W**
0:Not connected (Reference see P505:0)		Function of Network Input Word Bit 14			
P505:16	**0x400E:16**	**NETWordIN1.15**	**i510**	**i550**	**R/W**
0:Not connected (Reference see P505:0)		Function of Network Input Word Bit 15			

7.4.2 NETWordOut configuration

Actual value:

P591:1	0x400A:1	NetWordOUT1	i510	i550	R
Bit # description: 0: Mapping bit 0 1: Mapping bit 1 …		Actual value of mappable network out word 1 bit collector (Status bits) --> Trigger mapping 0x2634:1-3 (P420:1-3)			
P591:2	**0x400A:2**	**NetWordOUT2**	**i510**	**i550**	**R**
Bit # description: 0: Mapping bit 0 1: Mapping bit 1 …		No mapping			

Configuration:

P420:10	0x2634:10	NETWordOUT1 - bit 0	i510	i550	R/W
51:Ready for operation (Reference see P420:1)		Function of Network Bit 0			
P420:11	**0x2634:11**	**NETWordOUT1 - bit 1**	**i510**	**i550**	**R/W**
69:Inverse rotation (Reference see P420:1)		Function of Network Bit 1			
P420:12	**0x2634:12**	**NETWordOUT1 - bit 2**	**i510**	**i550**	**R/W**
50:Running (Reference see P420:1)		Function of Network Bit 2			
P420:13	**0x2634:13**	**NETWordOUT1 - bit 3**	**i510**	**i550**	**R/W**
56:Fault (Reference see P420:1)		Function of Network Bit 3			
P420:14	**0x2634:14**	**NETWordOUT1 - bit 4**	**i510**	**i550**	**R/W**
55:Safe Torque Off (Reference see P420:1)		Function of Network Bit 4			
P420:15	**0x2634:15**	**NETWordOUT1 - bit 5**	**i510**	**i550**	**R/W**
54:Quick stop active (Reference see P420:1)		Function of Network Bit 5			
P420:16	**0x2634:16**	**NETWordOUT1 - bit 6**	**i510**	**i550**	**R/W**
71:Actual speed = 0 (Reference see P420:1)		Function of Network Bit 6			
P420:17	**0x2634:17**	**NETWordOUT1 - bit 7**	**i510**	**i550**	**R/W**
58:Device warning (Reference see P420:1)		Function of Network Bit 7			
P420:18	**0x2634:18**	**NETWordOUT1 - bit 8**	**i510**	**i550**	**R/W**
115:Holding brake release (Reference see P420:1)		Function of Network Bit 8			
P420:19	**0x2634:19**	**NETWordOUT1 - bit 9**	**i510**	**i550**	**R/W**
78:At current limit (Reference see P420:1)		Function of Network Bit 9			
P420:20	**0x2634:20**	**NETWordOUT1 - bit 10**	**i510**	**i550**	**R/W**
72:Setpoint speed reached (Reference see P420:1)		Function of Network Bit 10			

P420:21	0x2634:21	NETWordOUT1 - bit 11	i510	i550	R/W
0:Not connected (Reference see P420:1)		Function of Network Bit 11			
P420:22	**0x2634:22**	**NETWordOUT1 - bit 12**	**i510**	**i550**	**R/W**
0:Not connected (Reference see P420:1)		Function of Network Bit 12			
P420:23	**0x2634:23**	**NETWordOUT1 - bit 13**	**i510**	**i550**	**R/W**
0:Not connected (Reference see P420:1)		Function of Network Bit 13			
P420:24	**0x2634:24**	**NETWordOUT1 - bit 14**	**i510**	**i550**	**R/W**
0:Not connected (Reference see P420:1)		Function of Network Bit 14			
P420:25	**0x2634:25**	**NETWordOUT1 - bit 15**	**i510**	**i550**	**R/W**
0:Not connected (Reference see P420:1)		Function of Network Bit 15			

8 Troubleshooting

8.1 Inverter LED status display

The inverter has two LEDs (RDY = READY, ERR = ERROR) on the front cover to identify the status of the inverter:

RDY (Blue)	ERR (Red)	State
–	–	No supply voltage
	–	STO active
		STO active, warning active
	–	Inverter inhibited
		Inverter inhibited, DC Voltage not
		Inverter inhibited, Warning active
		Inverter inhibited, Fault active
	–	Inverter released, drive running OR Quick Stop active
		Inverter released, drive running, Warning active
		Inverter released, Trouble reaction active