CANopen Specification
NA-9161
User Manual

Version 1.08

2013 CREVIS Co.,Ltd

DOCUMENT CHANGE SUMMARY				
REV.	**PAGES**	**REMARKS**	**DATE**	**Editor**
1.0	New Document	Draft	2011/3/15	JE Kang
1.01	14	IO LED	2011/10/21	JE Kang
1.02		Add your experience	2012/1/13	JE Kang
1.03		Modify the wrong letters and value	2012/2/15	JE Kang
		Changed Cover	2012/2/28	JE Kang
1.04		Add Cable and Connector	2012/5/10	JE Kang
1.05	15	Modify the LED Status	2012/7/25	JE Kang
1.06		Changed Crevis TEL	2013/4/4	JE KANG
1.07		Environment Spec. 50℃→55℃ (UL Temp)	2013/7/3	JE Kang
1.08		Modify the Pin Description	2014/05/08	YMKIM

TABLE OF CONTENTS

1. IMPORTANT NOTES

Solid state equipment has operational characteristics differing from those of electromechanical equipment.
Safety Guidelines for the Application, Installation and Maintenance of Solid State Controls describes some important differences between solid state equipment and hard-wired electromechanical devices.
Because of this difference, and also because of the wide variety of uses for solid state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

In no event will CREVIS be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.
The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, CREVIS cannot assume responsibility or liability for actual use based on the examples and diagrams.

Warning!

✓ **If you don't follow the directions, it could cause a personal injury, damage to the equipment or explosion**

- Do not assemble the products and wire with power applied to the system. Else it may cause an electric arc, which can result into unexpected and potentially dangerous action by field devices. Arching is explosion risk in hazardous locations. Be sure that the area is non-hazardous or remove system power appropriately before assembling or wiring the modules.
- Do not touch any terminal blocks or IO modules when system is running. Else it may cause the unit to an electric shock or malfunction.
- Keep away from the strange metallic materials not related to the unit and wiring works should be controlled by the electric expert engineer. Else it may cause the unit to a fire, electric shock or malfunction.

Caution!

✓ **If you disobey the instructions, there may be possibility of personal injury, damage to equipment or explosion. Please follow below Instructions.**

- Check the rated voltage and terminal array before wiring. Avoid the circumstances over 55℃ of temperature. Avoid placing it directly in the sunlight.
- Avoid the place under circumstances over 85% of humidity.
- Do not place Modules near by the inflammable material. Else it may cause a fire.
- Do not permit any vibration approaching it directly.
- Go through module specification carefully, ensure inputs, output connections are made with the specifications. Use standard cables for wiring.
- Use Product under pollution degree 2 environment.

1.1. Safety Instruction

1.1.1 Symbols

DANGER 	Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death property damage, or economic loss
IMPORTANT	Identifies information that is critical for successful application and understanding of the product
ATTENTION 	Identifies information about practices or circumstances that can lead to personal injury, property damage, or economic loss. Attentions help you to identity a hazard, avoid a hazard, and recognize the consequences

1.1.2 Safety Notes

DANGER 	The modules are equipped with electronic components that may be destroyed by electrostatic discharge. When handling the modules, ensure that the environment (persons, workplace and packing) is well grounded. Avoid touching conductive components, e.g. FnBUS Pin.

1.1.3 Certification

c-UL-us UL Listed Industrial Control Equipment, certified for U.S. and Canada
See UL File E235505

FCC

CE Certificate
EN 61000-6-2; Industrial Immunity
EN 61000-6-4; Industrial Emissions

2. DEVICE PROFILE

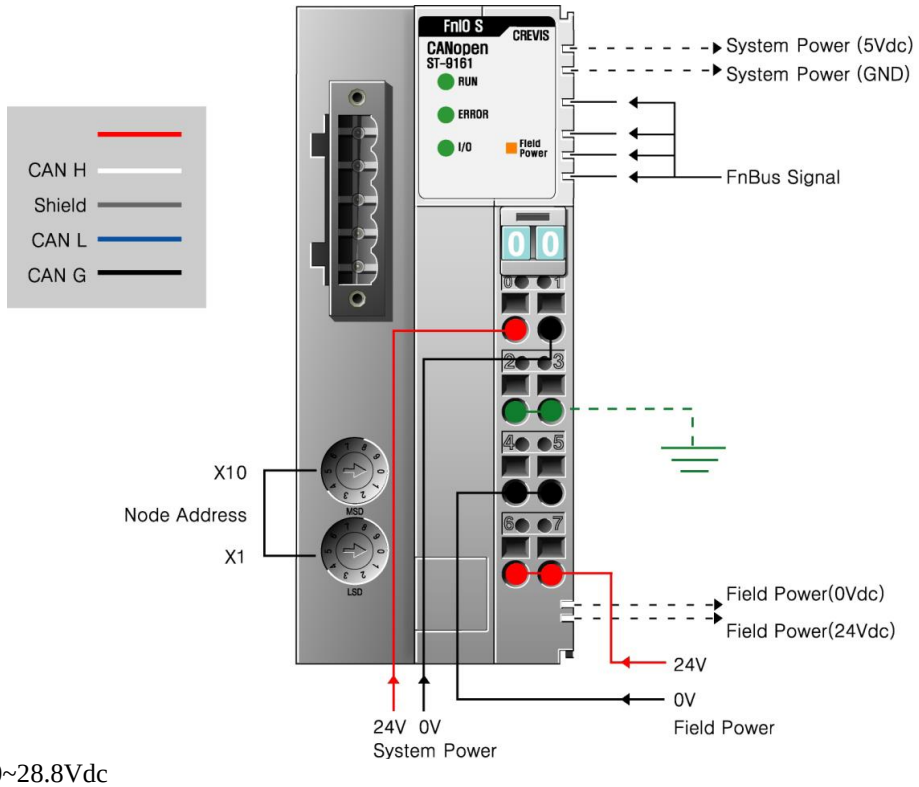
- Communications Adapter
- Device Type: 401D

3. CANOPEN COMMUNICATION

Communication Interface Specifications	
Number of Network modules	Max. 99 slot
Expansion I/O module	Max. 32 slot
Peripheral signals	Input 64byte / Output 64 byte
Indicators	1 green : CAN-RUN status indicator 1 red : CAN-ERR status indicator 1 red/green : FnBus status indicator 1 green : Field Power supply status indicator
Communication Rate	10 Kbps ...1 Mbps
Max. bus length	Depending on Baud rate
Number of PDOs available	8 Transmit PDOs / 8 Receive PDOs
Number of SDOs available	1 Standard SDOs

General Specification	
System Power	Supply voltage : 24Vdc nominal Voltage range : 11~28.8Vdc,
Power dissipation	Nominal 24Vdc@ 100mA
Current for I/O module	Max 5Vdc@ 1.5A
Isolation	Network to Logic : Isolation Logic to Field power : Isolation Logic to System power : Non-isolation
Field Power	Supply voltage : 24Vdc nominal Voltage range : 11~28.8Vdc
Current in jumper contacts	DC 10A maximum capacity
Weight	155g
Module Size	42 x 99 x 70 (W x H x L)
Environment Condition	
Operating Temperature	-20℃~55℃
Storage Temperature	-40℃~85℃
Relative Humidity	5% ~ 90% non-condensing
Operating Altitude	2000m
Mounting	DIN rail

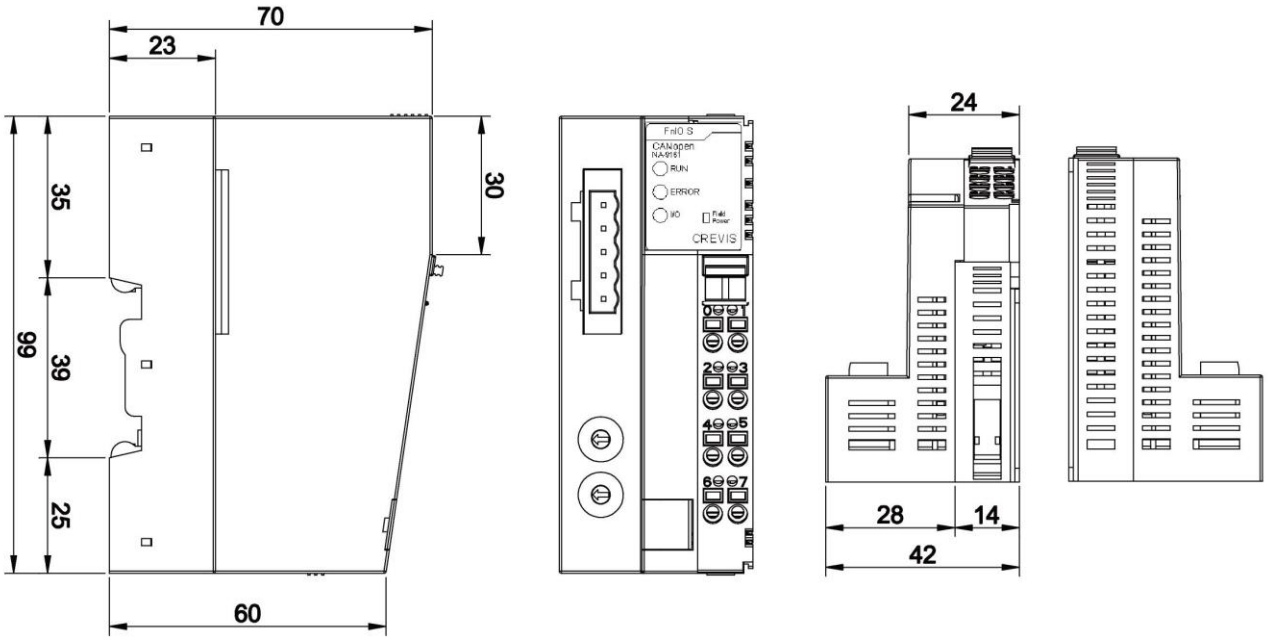
4. WIRE MAPPING



Field Power is 10~28.8Vdc
System power is 24Vdc (± 20%)

4.1 DIMENSION

(mm)

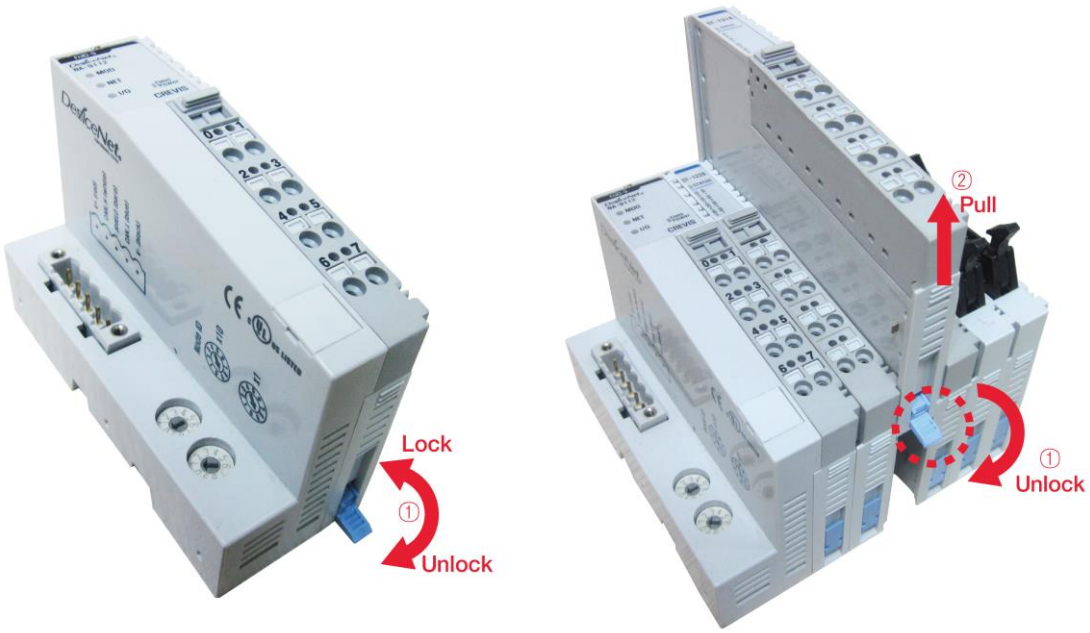


5. MECHANICAL SET UP

5.1 Total Expansion

The number of the module assembly that can be connected is 32. So the maximum length is 426mm Exception ST-2748 is excepted to calculate maximum length because that is double width module.

5.2 Plugging and Removal of the Components.

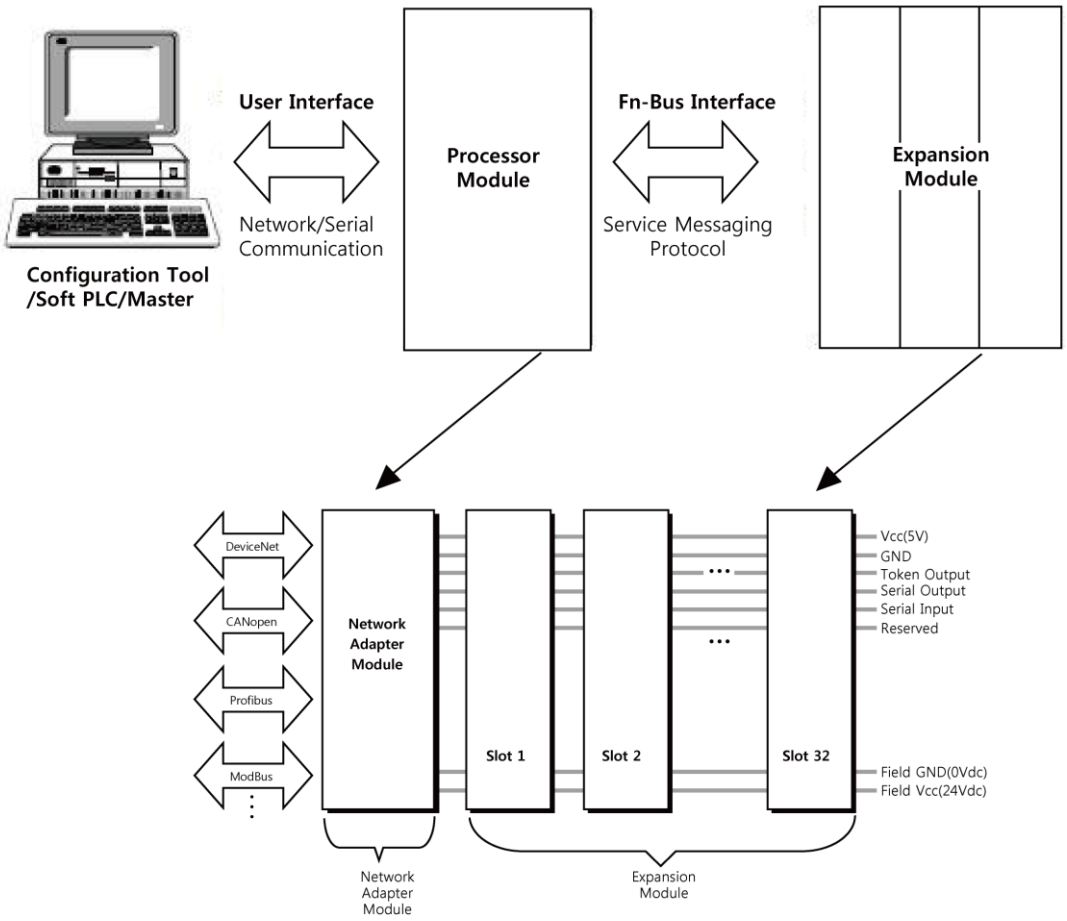


As above figure in order to safeguard the FnIO module from jamming, it should be fixed onto the DIN rail with locking level. To do so, fold on the upper of the locking lever.
To pull out the FnIO module, unfold the locking lever as below figure.

DANGER 	Before work is done on the components, the voltage supply must be turned off.

6. CANOPEN Electrical Interface

6.1. FnBus System



• Network Adapter Module

The Network Adapter Module forms the link between the field bus and the field devices with the Expansion Modules.

The connection to different field bus systems can be established by each of the corresponding Network Adapter Module, e.g. for SyncNet, PROFIBUS, CANopen, DeviceNet, Ethernet/IP, CC-Link, MODBUS/Serial, MODBUS/TCP etc.

• Expansion Module

The Expansion Modules are supported a variety of input and output field devices.

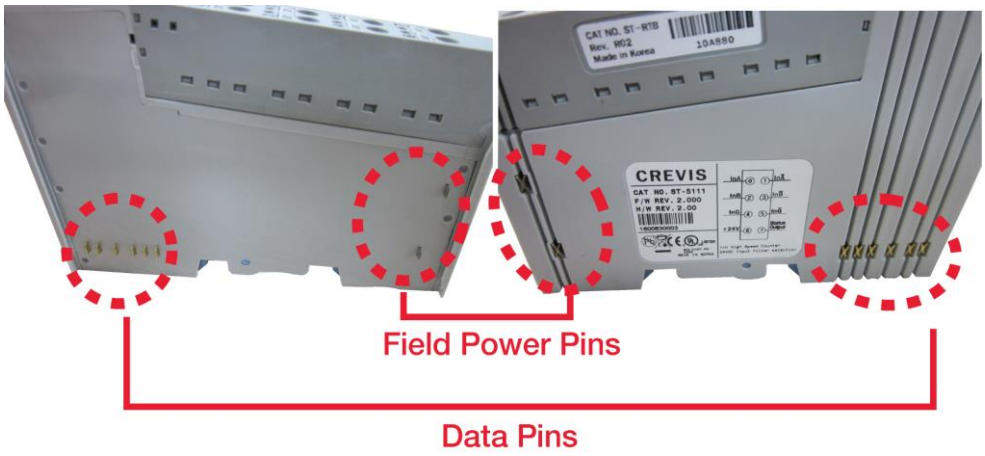
There are digital and analog input/output modules and special function modules.

• Two types of FnBus Message

- Service Messaging
- I/O Messaging

6.2. FnBus Pin Description

Communication between the NA series and the expansion module as well as system / field power supply of the bus modules is carried out via the internal bus. It is comprised of 6 data pin and 2 field power pin.



No.	Name	Description
1	Vcc	System supply voltage (5V dc).
2	GND	System Ground.
3	Token Output	Token output port of Processor module.
4	Serial Output	Transmitter output port of Processor module.
5	Serial Input	Receiver input port of Processor module.
6	Reserved	Reserved for bypass Token.
7	Field GND	Field Ground.
8	Field Vcc	Field supply voltage (24Vdc).

DANGER 	Do not touch data and field power pins in order to avoid soiling and damage by ESD noise.

7. CONFIGURATION OF THE FIELD BUS NODE

7.1 Network Address and Baud rate

Before starting Adapter operation the node number (node ID) and the Network adapter's baud rate must be set. These settings are made by means of 2 rotary switches on the Adapter.

7.1.1 Node ID

The coupler's node ID is set with Rotary switches.
The node ID can be set in the range from 1 to 99(the ID 0 is not allowed).

7.1.2 Baud rate

Before starting Rotary switches must be '0' '0'.
The switch in the table above can be stored when '1' 'x'. (X is don't care)

The select S/W can be set in the range from 0 to 8
The Acceptance S/W can be set in the range from 0 to 1

	0	1	2	3	4	5	6	7	8	9
Select S/W	0	1	2	3	4	5	6	7	8	9
Acceptance S/W	'0'->'1'	'0'->'1'	'0'->'1'	'0'->'1'	'0'->'1'	'0'->'1'	'0'->'1'	'0'->'1'	'0'->'1'	0->'1'
Baud rate	1MB	800KB	500KB	250KB	125KB	100KB	50KB	20KB	10KB	Auto baud rate

ATTENTION	NODE ID addresses have to be unique throughout the entire interconnected networks.

7.2 CANopen Cable and connector

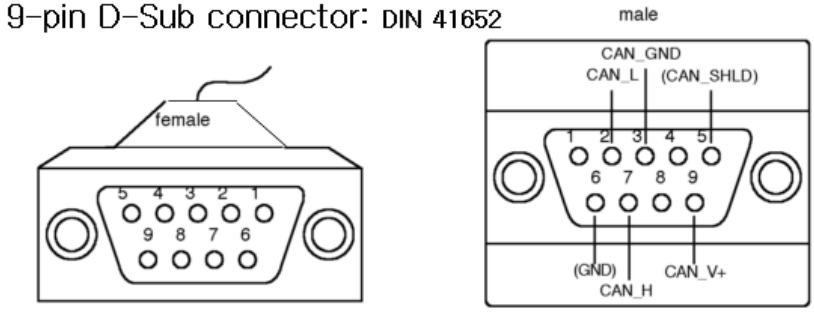
CANopen does not stipulate the physical media, but for the connector pin-assignment is presented.

7.2.1 Wire - Cable

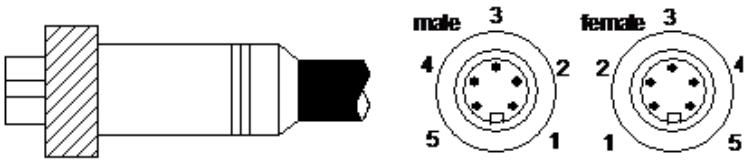
- Twisted pair
- Shielded twisted pair
  *CANopen does not stipulate the wire. Refer to “Appendix B”

7.2.2 Connector

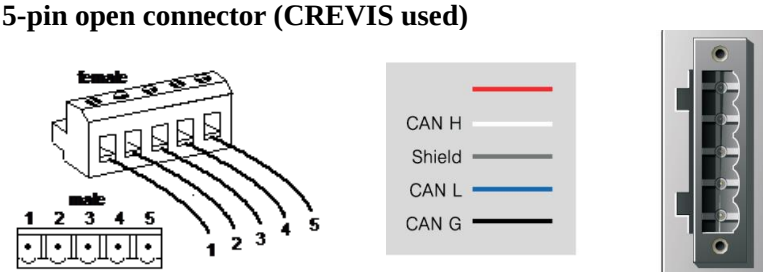
- 9-pin Dsub



- 5-pin mini connector



- **5-pin open connector (CREVIS used)**



Pin	Signal	Description
1	CAN_GND	Ground / 0V / V-
2	CAN_L	CAN_L bus line (dominant low)
3	(CAN_SHILD)	Optional CAN Shield
4	CAN_H	CAN_H bus line (dominant high)
5	(CAN_V+)	Optional CAN external positive supply

- **FJ-45**

8. STATUS INDICATOR LED

8.1 CAN-RUN LED Status

State	LED is:	To indicate:
Not Powered Not On-line	Off	The Device is not on-line or may not be powered - Not completed the Dup-MAC_ID test yet
On-line, STOPED	Single flash Green	The Device is in STOPED state
On-line, PRE-OPERATIONAL	Blinking Green	The Device is in the PRE-OPERATIONAL state
On-line, OPERATIONAL	Green	The Device is in the OPERATIONAL state

8.2 CAN-ERR LED Status

State	LED is :	To indicate :
Not Powered Not On-line	Off	The Device is in working condition
Warning limit reached On-line	Single flash Red	At least one of the error counters of the CAN controller has reached or exceeded the warning level (too many error frames).
Error Control Event On-line	Double flash Red	The guarding monitor has asserted, guarding telegrams are no longer being received. The adapter is pre-operational state.
Sync Error On-line	Triple flash Red	A sync error has occurred. - The adapter is pre-operational (PDOs switch off).
Bus Off	Red	The CAN controller is bus off.

8.3 Field Power LED Status

State	LED is :	To indicate :
Not Supplied Field Power	Off	Not supplied 24V dc field power
Supplied Field Power	Green	Supplied 24V dc field power

8.4 Expansion Module Status LED (I/O)

State	LED is :	To indicate :
Not Powered No Expansion Module	Off	Device has no expansion module or may not be powered
FnBus On-line, Do not Exchanging I/O	Flashing Green	FnBus is normal but does not exchanging I/O data (Passed the expansion module configuration).
FnBus Connection, Run Exchanging IO	Green	Exchanging I/O data
FnBus connection fault during exchanging IO	Red	One or more expansion module occurred in fault state. - Changed expansion module configuration. - FnBus communication failure.
Expansion Configuration Failed	Flashing Red	Failed to initialize expansion module - Detected invalid expansion module ID. - Overflowed Input / Output Size - Too many expansion module - Initial protocol failure - Mismatch vendor code between adapter and expansion module.

9. COMMUNICATION

9.1 Device model

9.1.1 Structure of the device model

Communication. This functional unit makes the communication data objects and the associated functionality for data exchange over the CANopen network available. The network status machine is part of this.
Object directory. This contains all the data objects (application data + parameters) that are accessible from outside and that affect the behavior of communication, application and status machines. The object directory is organized as a two-dimensional table in which the data are addressed by their index and sub-index.
The data exchange with CANopen devices takes place by means of data objects. In the CANopen communication profile, two types of standard object (PDO and SDO) and special objects (for network management etc.) are defined. The NA-9161 supports the following objects:

- ✓ 8 transmit PDOs
- ✓ 8 receive PDOs
- ✓ 1 standard SDO (server)
- ✓ 1 emergency object
- ✓ 1 synchronization object (SYNC, without time stamp)
- ✓ Node guarding
- ✓ NMT objects

Every CANopen device possesses a CANopen object directory in which
The parameters for all the CANopen objects are entered.

9.2 PDO (Process Data)

9.2.1 Introduction

In many fieldbus systems the entire process image is continuously transferred - usually in a more or less cyclic manner. CANopen is not limited to this communication principle, since the multi-master bus access protocol allows CAN to offer other methods.
The process data in CANopen is divided into segments with a maximum of 8 bytes. These segments are known as process data objects (PDOs). The PDOs each corresponds to a CAN telegram, whose specific CAN identifier is used to allocate them and to determine their priority.
The PDOs are named from the point of view of the Network Adaptor: receive PDOs (RxPDOs) are received by the coupler and contain output data, while transmit PDOs (TxPDOs) are sent by the coupler and contain input data.

9.2.2 PDO Mapping

CANopen specifies the data assignment for the first two PDOs in the device profile for input/output groups (DS401) (“default mapping”). The first PDO is provided for digital inputs (TxPDO1) or outputs (RxPDO1). The first 4 analogue inputs or outputs are located in the second PDO. These PDOs are accordingly occupied by the Network Adaptors - if, for instance, no digital output terminals are plugged in, RxPDO1 remains empty.

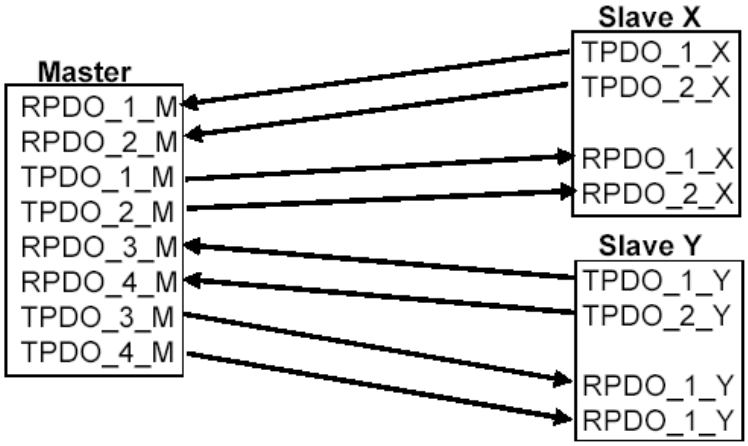
Once the first PDOs have been occupied, the next PDOs are filled with process data in the following sequence:
1. Digital I/O (1-byte)
2. Digital I/O (2-byte)
3. Analogue I/O

9.2.3 PDO Identifier

For the first two PDOs (PDO1 + PDO2) CANopen provides default identifiers depending on the node address, but all other PDOs must have identifiers assigned to them. The principle of the default identifiers is explained in the section on "Network Management", and there is a list of all the CANopen default identifiers in the appendix.

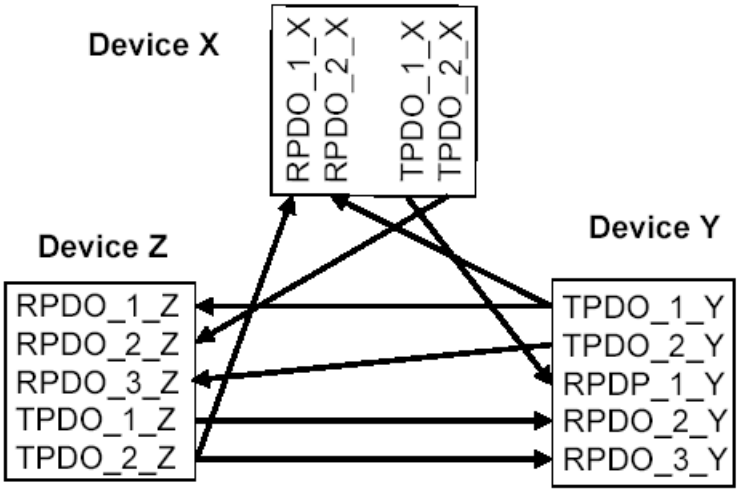
Pre-Define Connection Set

In the system of default identifiers, all the nodes (here: slaves) communicate with one central station (the master), since slave nodes do not listen by default to the send identifier of other slave nodes:



PDO Linking

If the consumer-producer model of CANopen PDOs is to be used for direct data exchange between nodes (without a master), the distribution of identifiers must be appropriately adapted, so that the TxPDO identifier of the producer agrees with the RxPDO identifier of the consumer:



This procedure is known as PDO linking. It permits, for example, easy construction of electronic drives in which several slave axes simultaneously listen to the actual value in the master axis TxPDO.

9.2.4 PDO Communication Type

Event driven

The “event” is the alteration of an input value, the data being transmitted immediately after this change. The event-driven flow can make optimal use of the bus bandwidth, since instead of the whole process image it is only the changes in it that are transmitted. A short reaction time is achieved at the same time, since when an input value changes it is not necessary to wait for the next interrogation from a master.

Polling

The PDOs can also be polled by data request telegrams (remote frames). In this way it is possible to get the input process image of event-driven inputs onto the bus, even when the inputs have not changed, for instance by a monitoring or diagnostic device brought into the network while it is running.
The Crevis CANopen bus Adapter supports the interrogation of PDOs by means of remote frames.

Synchronized

It is not only for drive applications that it is worthwhile to synchronize the determination of the input information and the setting the outputs. For this purpose CANopen provides the SYNC object, a CAN telegram of high priority but containing no user data, whose reception is used by the synchronized nodes as a trigger for reading the inputs or for setting the outputs:

PDO transmission type

The “PDO transmission type” parameter specifies how the transmission of the PDO is triggered, or how received PDOs are handled:

Transmission type	**PDO transmission**				
	cyclic	acyclic	synchronous	asynchronous	RTR only
0		X	X		
1-240	X		X		
241-251	reserved				
252			X		X
253				X	X
254				X	
255				X	

Synchronous

Transmission type 0 is only useful for RxPDOs: the PDO is only used when the next SYNC telegram is received. In transmission types 1-240 the PDO is cyclically transmitted or expected: after every “nth” SYNC (n = 1...240).
Since transmission types can be combined on a coupler as well as in the network, it is possible, for example, for a fast cycle to be agreed for digital inputs (n = 1), whereas the data for analogue inputs is transmitted in a slower cycle (e.g. n = 10). The cycle time (SYNC rate) can be monitored (object 0x1006), so that if the SYNC fails the Adapter switches its outputs into the fault state.

Asynchronous

The transmission types 254 + 255 are asynchronous, but may also be event-driven. In transmission type 254, the event is specific to the manufacturer, whereas for type 255 it is defined in the device profile. Since the Beckhoff CANopen Network Adaptors support device profile DS401 no distinction is made here between the two transmission types.

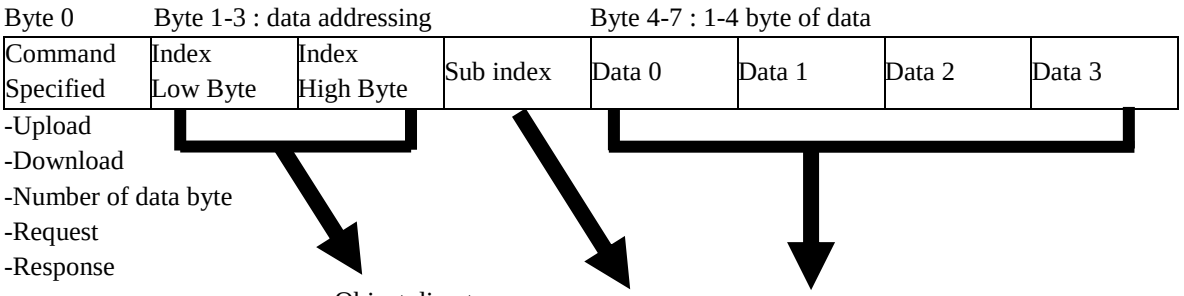
Inhibit Time

The “inhibit time” parameter can be used to implement a “transmit filter” that does not increase the reaction time for relatively new input alterations, but is active for changes that follow immediately afterwards. The inhibit time (transmit delay time) specifies the minimum length of time that must be allowed to elapse between the transmission of two of the same telegrams. If the inhibit time is used, the maximum bus loading can be determined, so that the worst case latency can then be found.

9.3 SDO (Service Data)

9.3.1 Introduction

The parameters listed in the object directory are read and written by means of service data objects. These SDOs are multiplexed domains, i.e. structures of any size that have a multiplexer (address). The multiplexer consists of a 16-bit index and an 8-bit sub-index that address the corresponding entries in the object directory.



Object directory

Index	Sub-index	Data

The CANopen Network Adaptors are servers for the SDO, which means that at the request of a client they make data available (upload), or they receive data from the client (download).
This involves a handshake between the client and the server. When the size of the parameter to be transferred is not more than 4 bytes, a single handshake is sufficient (one telegram pair).
For a download, the client sends the data together with its index and sub-index, and the server confirms reception. For an upload, the client requests the data by transmitting the index and sub-index of the desired parameter and the server sends the parameter (including index and sub-index) in its answer telegram. The same pair of identifiers is used for both upload and download. The telegrams, which are always 8 bytes long, encode the various services in the first data byte.
All parameters with the exception of objects 1008h, 1009h and 100Ah (device name, hardware and software versions) are only at most 4 bytes long, so this description is restricted to transmission in expedited transfer.

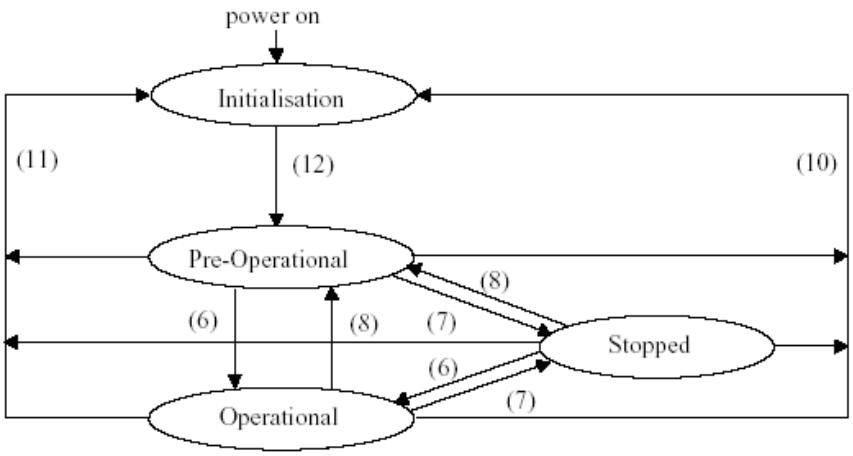
9.4 Emergency (Error Message)

Emergency messages are always sent in the event of a critical error situation having occurred/overcome in the device, or if important information has to be communicated to other devices.
Structure and meaning of the entries in the emergency object are explained in the table .EMCY-CODE., they are coded in the bus message in a 'Low byte' / 'High byte' order.
An emergency object is also sent, after an error is remedied (Error Code = 0x0000, the Error Register and the Additional Code behave as described in the table .EMCY-CODE.).
Following Power On an emergency object is sent if the loaded settings are the default settings. This occurs for two reasons:
• No settings have yet been saved (Index 0x1010).
• The saved setting was discarded by the Network Adaptor, because modules were connected or disconnected.

9.5 NMT (Network Management)

9.5.1 Network Start-up

CANopen defines a state machine that controls the functionality of a device. Transition between the individual states is initiated by internal events or services from the NMT master. These devices states can be connected to application processes.



In ***Initialization*** state, the CANopen data structures of a node are initialized by the application. The CiA DS-301 standard defines various mandatory OD entries for this task as well as specific communication objects required for that. In the minimum device configuration, the identifier for these communication objects must correspond to the so-called **Pre-Defined Connection-Set.** The device profiles define further settings for the applicable device class. The pre-defined settings for identifier for emergency, PDOs and SDOs are calculated based on the node address (Node ID) that can be in the range from 1 to 99, added to a base identifier that determines the function of the individual object.
After ***Initialization*** is completed the node automatically switches into **PRE-OPERATIONAL** (12) state. The NMT master will be informed about this state change with the BOOT-UP message sent by the corresponding node. In this state it is not possible to communicate with the node using PDOs. However, the node can be configured over the CAN bus using SDOs in **PRE-OPERATIONAL** state. NMT services and Life Guarding are also available in this state.
The application as well as the available resources of the CANopen device determine to what extend configuration over the CAN bus with the help of SDOs must take place. For example, if the CANopen device does not provide a non-volatile memory to store mapping and communication parameters for PDOs and these parameters differ from the default values, then these parameters must be transmitted to the node over the network after initialization is completed.
After the configuration of these parameters by the application or the NMT master is completed, the NMT service *Start_Remote_Node* (6) can be used to render the node from **PRE-OPERATIONAL** state into **OPERATIONAL** state. This state change also causes the initial transmission of all TPDOs independently of whether an event for it is present. Each subsequent transmission of PDOs then always takes place as a function of an event.
All CANopen devices also support the Stop_Remote_Node (7), Enter_PRE-OPERATIONAL_State (8), Reset_Node (10), *Reset_Communication* (11) services. *Reset_Node* is used to reset the application-specific data and the communication parameter of the node. The CANopen data structures are loaded with their initial values. Data stored in a non-volatile memory are rejected. This state change is comparable with an initial operation of the node.
If the NMT service *Reset_Communication* is used to change the state of a node, then loading initial values exclusive for the communication parameters in the CANopen stack takes place.
No communication via PDO and SDO is possible if the device is in **STOPPED** state. Only NMT services, Node Guarding, Life Guarding as well as Heartbeat are possible in this state.

9.5.2 Boot-up Message

After the initialization phase and the self-test, the Network Adaptor sends the boot-up message, a CAN message with no data bytes and with the identifier of the emergency message: CAN-ID = 0x80 + node ID.

9.5.3 Node Guarding

Node Guarding represents a means of node supervision that is initiated by the NMT master. This service is used to request the node's operational state and to determine whether the node is functioning correctly. The NMT master transmits a single Node Guard message to the slave in the form of a remote frame with the CAN identifier 0x700 plus the node address of the NMT slave. As a response to this remote frame, the NMT slave sends a CAN message back containing its current NMT state and a one bit that toggles between two subsequent messages.
Response from the NMT Slave to a Node Guard Remote Frame:

Identifier	DLC	Data
		0
0x700 + Node Address	1	Status Byte

Node State of a CANopen Device

Status Byte	Node State
0x00	BOOT-UP
0x04	STOPED
0x05	OPERATIONAL
0x7F	PRE-OPERATIONAL

Bit 7 of the status byte always starts with a 0 and changes its value after each transmission. The application is responsible for actively toggling this bit. This ensures that the Node Guard response message from a slave is not just stored in one of the Full-CAN channels. Thus the NMT master will get the confirmation from the NMT slave node that the application is still running.

9.5.4 Life Guarding

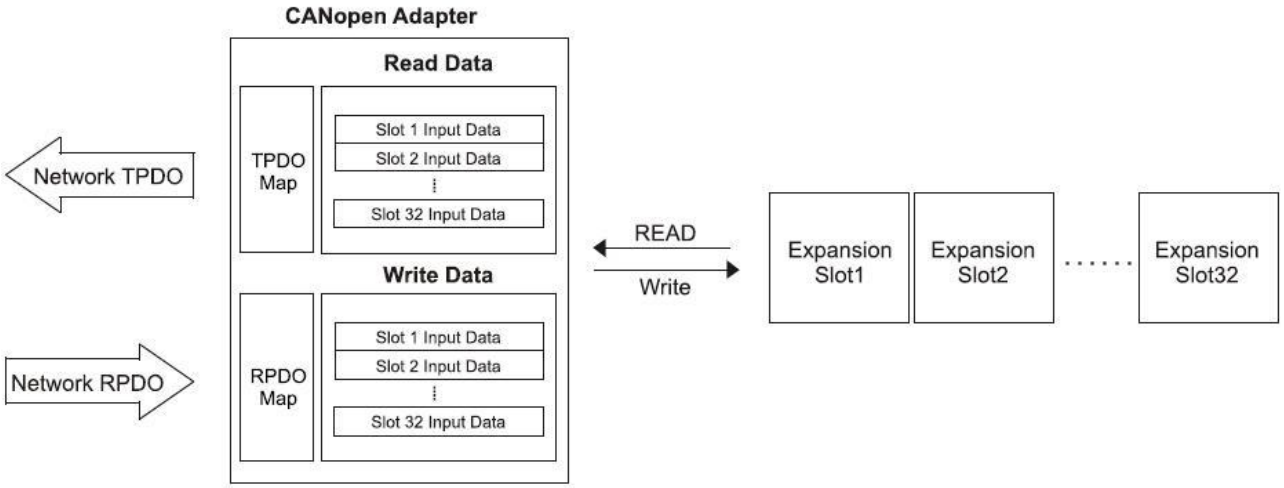
As an alternative to Node Guarding node supervision can also be performed by Life Guarding services. In contrast to the Node Guarding the NMT master cyclically sends a Life Guard message to the slave in the form of a remote frame with the CAN identifier 0x700 plus the node address of the NMT slave. As a response to this remote frame, the NMT slave sends a CAN message back containing its current NMT state and a one bit that toggles between two subsequent messages. With being missing the answer or unexpected status of the slave the NMT masters application is informed. Further the slave can detect the loss of the masters. The Life Guarding is started with the transmission of the first Life Guard message of the masters.
Response from the NMT Slave to a Life Guard Remote Frame

Identifier	DLC	Data
		0
0x700 + Node Address	1	Status Byte

Meaning of the status byte corresponds to that of the Node Guarding message The Life Guarding supervision on the NMT slave node is deactivated, if the Life Guard time (object entry 0x100C in the object dictionary) or the Life time factor (object entry 0x100D in the object dictionary) are equal to zero.

9.6. I/O Process Image Map

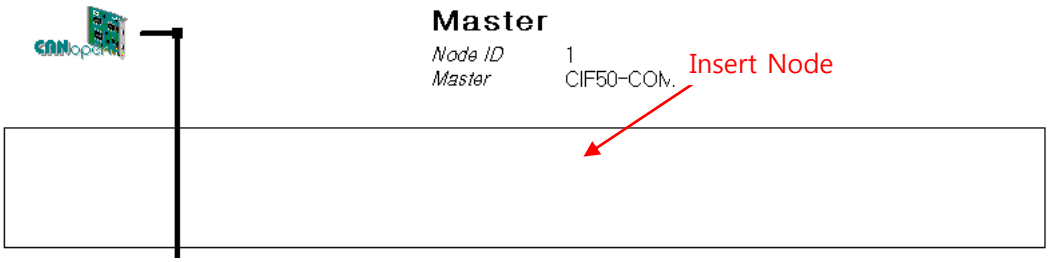
An expansion module may have 3 types of data as I/O data, configuration parameter and memory register.
The data exchange between network adapter and expansion modules is done via an I/O process image data by FnBus protocol. The following figure shows the data flow of process image between network adapter and expansion modules.



9.6.1 Connect node Example

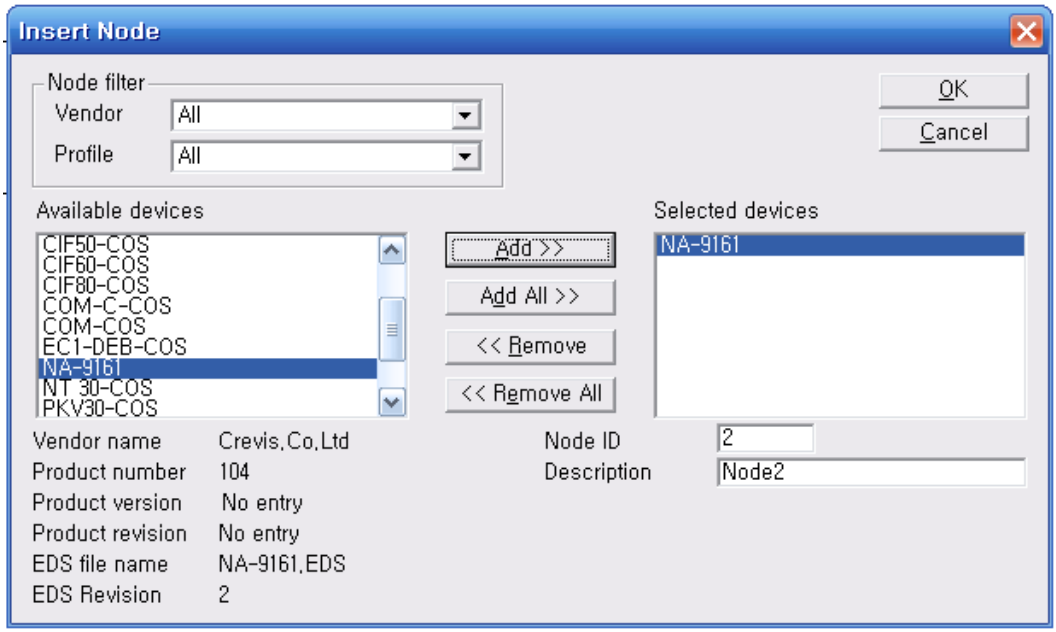
Example of NA-9161 with Sycon.

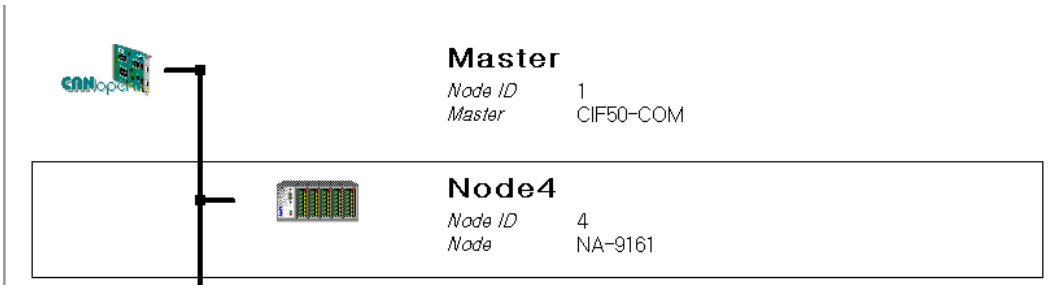
- Insert NA-9161



-EDS file is downloading of homepage.
Copy the file to an EDS folder.

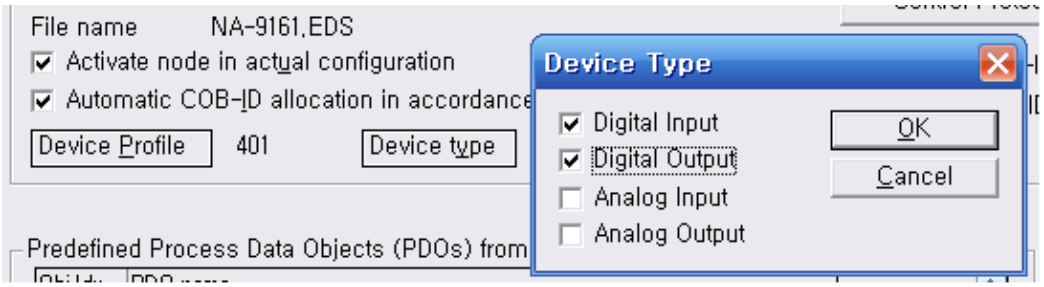
Program Files→ Hilscher→ Sycon→ Fieldbus→ CANopen→ EDS folder

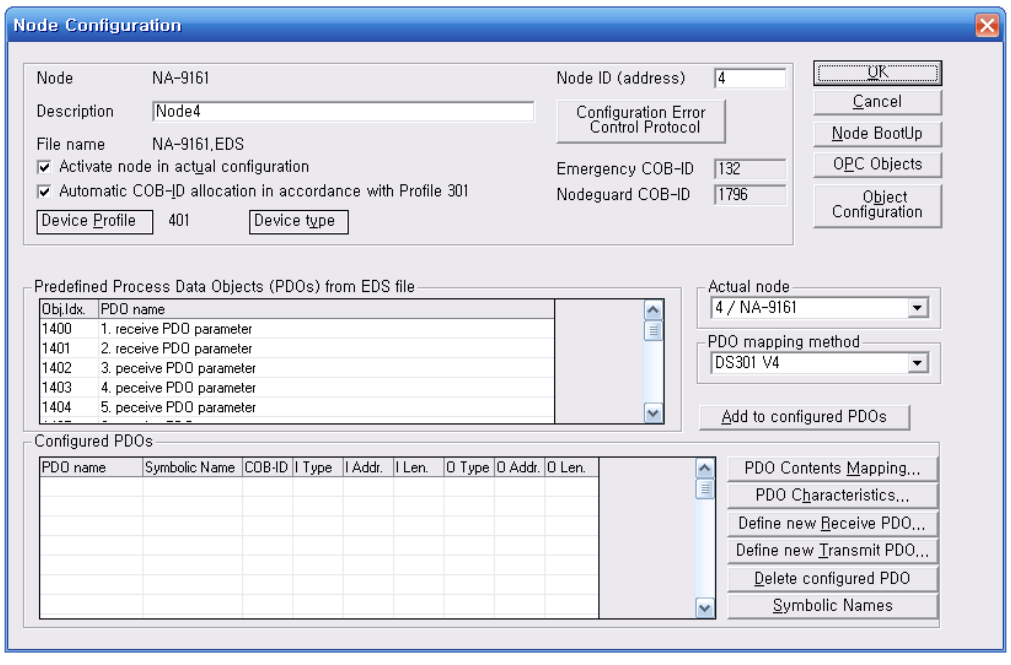


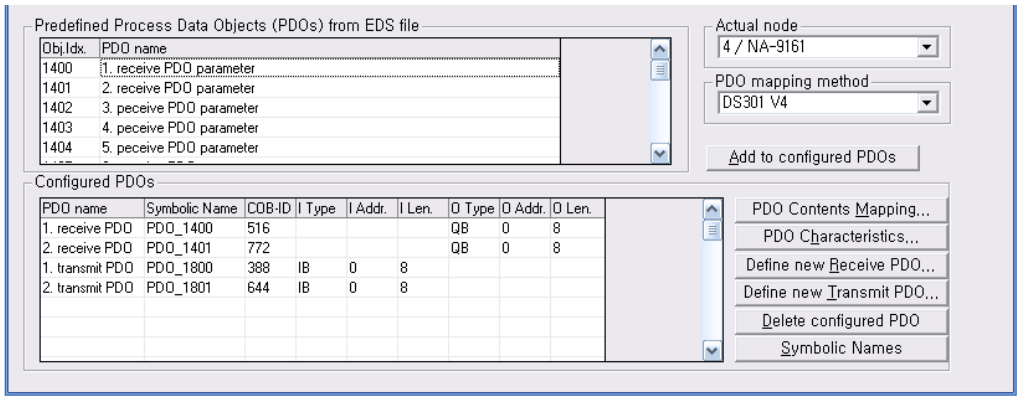


-Open Node Configuration setting
-'Predefined Process Data Objects (PDOs) from EDS file'

Obj.Idx	Description	PDOName
1400	Discrete Output	1.ReceivePDOparameter
1401	Analog Output	2.ReceivePDOparameter
1800	Discrete Input	1.TransmitPDOparameter
1801	Analog Input	2.TransmitPDOparameter





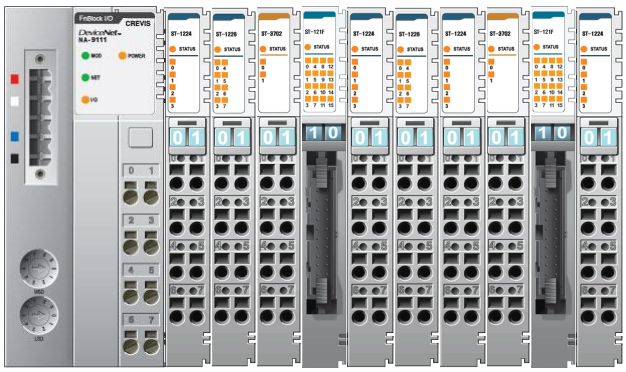


- Until now, one that will download to master.

9.6.2 Example of Input Process Image Map

Input image data depends on slot position and expansion slot data type. Input process image data is only ordered by expansion slot position when input image mode is uncompressed (mode 0). But, when input image mode is compressed (mode 1), input process image data is ordered by expansion slot position and slot data type.
Input process image mode can be set by Object Index 0x4500

For example slot configuration



Slot Address	Module Description
0	CANopen Adapter
1	4-discrete input
2	8-discrete input
3	2-analog input
4	16-discrete input
5	4-discrete input
6	8-discrete input
7	4-discrete input
8	2-analog input
9	16-discrete input
10	1ch, high speed counter

Compress mode data format

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Index	Sub-Index
0	Analog Input Ch0 low byte (Slot#3)								0x6401	0x01
1	Analog Input Ch0 high byte (Slot#3)								0x6401	0x01
2	Analog Input Ch1 low byte (Slot#3)								0x6401	0x02
3	Analog Input Ch1 high byte (Slot#3)								0x6401	0x02
4	Analog Input Ch0 low byte (Slot#8)								0x6401	0x03
5	Analog Input Ch0 high byte (Slot#8)								0x6401	0x03
6	Analog Input Ch1 low byte (Slot#8)								0x6401	0x04
7	Analog Input Ch1 high byte (Slot#8)								0x6401	0x04
8	Discrete Input 4 pts. (Slot#2)				Discrete Input 4 pts. (Slot#1)				0x6000	0x01
9	Discrete Input 4 pts. (Slot#4)				Discrete Input 4 pts. (Slot#2)				0x6000	0x02
10	Discrete Input 8 pts. (Slot#4)								0x6000	0x03
11	Discrete Input 4 pts. (Slot#5)				Discrete Input 4 pts. (Slot#4)				0x6000	0x04
12	Discrete Input 8 pts. (Slot#6)								0x6000	0x05
13	Discrete Input 4 pts. (Slot#9)				Discrete Input 4 pts. (Slot#7)				0x6000	0x06
14	Discrete Input 8 pts. (Slot#9)								0x6000	0x07
15					Discrete Input 4 pts. (Slot#9)				0x6000	0x08
16	HSC Input 0byte(Slot#10)								0x3000	0x01
17	HSC Input 1byte(Slot#10)								0x3000	0x02
18	HSC Input 2byte(Slot#10)								0x3000	0x03
19	HSC Input 3byte(Slot#10)								0x3000	0x04
20	HSC Input 4byte(Slot#10)								0x3000	0x05
21	HSC Input 5byte(Slot#10)								0x3000	0x06

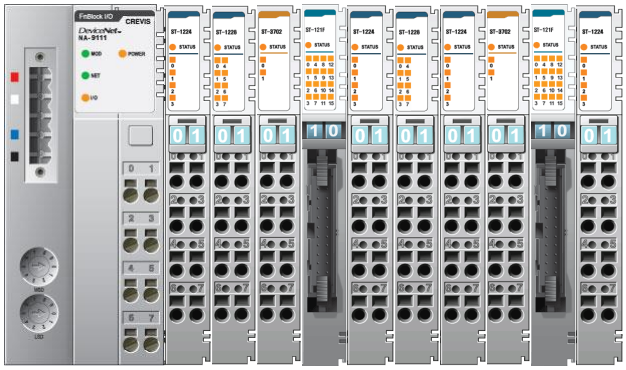
Non-compress mode data format

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Index	Sub-Index
0	Analog Input Ch0 low byte (Slot#3)								0x6401	0x01
1	Analog Input Ch0 high byte (Slot#3)								0x6401	0x01
2	Analog Input Ch1 low byte (Slot#3)								0x6401	0x02
3	Analog Input Ch1 high byte (Slot#3)								0x6401	0x02
4	Analog Input Ch0 low byte (Slot#8)								0x6401	0x03
5	Analog Input Ch0 high byte (Slot#8)								0x6401	0x03
6	Analog Input Ch1 low byte (Slot#8)								0x6401	0x04
7	Analog Input Ch1 high byte (Slot#8)								0x6401	0x04
8	Reserved				Discrete Input 4 pts. (Slot#1)				0x6000	0x01
9	Discrete Input 8 pts. (Slot#2)								0x6000	0x02
10	Discrete Input low 8 pts. (Slot#4)								0x6000	0x03
11	Discrete Input high 8 pts. (Slot#4)								0x6000	0x04
12	Reserved				Discrete Input 4 pts. (Slot#5)				0x6000	0x05
13	Discrete Input 8 pts. (Slot#6)								0x6000	0x06
14	Reserved				Discrete Input 4 pts. (Slot#7)				0x6000	0x07
15	Discrete Input low 8 pts. (Slot#9)								0x6000	0x08
16	Discrete Input high 8 pts. (Slot#9)								0x6000	0x09
17	Reserved				Discrete Input 4 pts. (Slot#10)				0x6000	0x0A
18	HSC Input 0byte(Slot#10)								0x3000	0x01
19	HSC Input 1byte(Slot#10)								0x3000	0x02
20	HSC Input 2byte(Slot#10)								0x3000	0x03
21	HSC Input 3byte(Slot#10)								0x3000	0x04
22	HSC Input 4byte(Slot#10)								0x3000	0x05
23	HSC Input 5byte(Slot#10)								0x3000	0x06

9.6.3 Example of Output Process Image Map

Output image data depends on slot position and expansion slot data type. Output process image data is only ordered by expansion slot position when output image mode is uncompressed (mode 0). But, when output image mode is compressed (mode 1), output process image data is ordered by expansion slot position and slot data type.
Output process image mode can be set by Object Index 0x4500.

For example slot configuration



Slot Address	Module Description
0	CANopen Adapter
1	4-discrete output
2	8-discrete output
3	2-analog output
4	16-discrete output
5	4-discrete output
6	8-discrete output
7	2-realy output
8	2-realy output
9	2-analog output
10	16-discrete output
11	1ch, high speed counter

Compress mode data format

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Index	Sub-Index
0	Analog Output Ch0 low byte (Slot#3)								0x6411	0x01
1	Analog Output Ch0 high byte (Slot#3)								0x6411	0x01
2	Analog Output Ch1 low byte (Slot#3)								0x6411	0x02
3	Analog Output Ch1 high byte (Slot#3)								0x6411	0x02
4	Analog Output Ch0 low byte (Slot#9)								0x6411	0x03
5	Analog Output Ch0 high byte (Slot#9)								0x6411	0x03
6	Analog Output Ch1 low byte (Slot#9)								0x6411	0x04
7	Analog Output Ch1 high byte (Slot#9)								0x6411	0x04
8	Discrete Output 4 pts. (Slot#2)				Discrete Output 4 pts. (Slot#1)				0x6200	0x01
9	Discrete Output 4 pts. (Slot#4)				Discrete Output 4 pts. (Slot#2)				0x6200	0x02
10	Discrete Output low 8 pts. (Slot#4)								0x6200	0x03
11	Discrete Output 4 pts. (Slot#5)				Discrete Output 4 pts. (Slot#4)				0x6200	0x04
12	Discrete Output 8 pts. (Slot#6)								0x6200	0x05
13	Discrete Output 4 pts. (Slot#10)				Discrete Output 2 pts. (Slot#8)		Discrete Output 2 pts. (Slot#7)		0x6200	0x06
14	Discrete Output high 8 pts. (Slot#10)								0x6200	0x07
15	Reserved				Discrete Output 4 pts. (Slot#10)				0x6200	0x08
16	HSC Output low byte(Slot#11)								0x3200	0x01
17	HSC Output high byte(Slot#11)								0x3200	0x02

Non-compress mode data format

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Index	Sub-Index
0	Analog Output Ch0 low byte (Slot#3)								0x6411	0x01
1	Analog Output Ch0 high byte (Slot#3)								0x6411	0x01
2	Analog Output Ch1 low byte (Slot#3)								0x6411	0x02
3	Analog Output Ch1 high byte (Slot#3)								0x6411	0x02
4	Analog Output Ch0 low byte (Slot#9)								0x6411	0x03
5	Analog Output Ch0 high byte (Slot#9)								0x6411	0x03
6	Analog Output Ch1 low byte (Slot#9)								0x6411	0x04
7	Analog Output Ch1 high byte (Slot#9)								0x6411	0x04
8	Reserved				Discrete Output 4 pts. (Slot#1)				0x6200	0x01
9	Discrete Output 8 pts. (Slot#2)								0x6200	0x02
10	Discrete Output low 8 pts. (Slot#4)								0x6200	0x03
11	Discrete Output high 8 pts. (Slot#4)								0x6200	0x04
12	Reserved				Discrete Output 4 pts. (Slot#5)				0x6200	0x05
13	Discrete Input 8 pts. (Slot#6)								0x6200	0x06
14	Reserved						Discrete Output 2 pts. (Slot#7)		0x6200	0x07
15	Reserved						Discrete Output 2 pts. (Slot#8)		0x6200	0x08
16	Discrete Output low 8 pts. (Slot#10)								0x6200	0x09
17	Discrete Output high 8 pts. (Slot#10)								0x6200	0x0A
18	Reserved				Discrete Output 4 pts. (Slot#11)				0x6200	0x0B
19	HSC Output low byte(Slot#11)								0x3200	0x01
20	HSC Output high byte(Slot#11)								0x3200	0x02

9.6.4 Default Identifier

CANopen provides default identifiers for the most important communication objects and these are derived from the 7-bit node address (the node ID) and a 4-bit function code in accordance with the following scheme:
11 Bit Identifier

10	9	8	7	6	5	4	3	2	1	0
Function				Code Node ID						

The COB ID is given according to DS301. This gives rise to the following default identifiers:

Object	Function	Function Code	COB ID (hex/Dec)	Object for Communication parameter/mapping
NMT	Boot-up	0000	0x00 / 0	-
SYNC	Synch.	0001	0x80 / 128	0x1500+0x1006
EMERGENCY	Status/Error	0001	0x81-0xFF / 129-255	-
PDO 1(Tx)	Digital Input	0011	0x181-0x1FF / 385-511	0x1800/0x1A00
PDO 1(Rx)	Digital Output	0100	0x201-0x27F / 513-639	0x1400/0x1600
PDO 2(Tx)	Analog Input	0101	0x281-0x2FF / 641-767	0x1801/0x1A01
PDO 2(Rx)	Analog Output	0110	0x301-0x37F / 769-895	0x1401/0x1601
SDO (Tx)	Parameter	1011	0x581-0x5FF / 1409-1535	-
SDO (Rx)	Parameter	1100	0x601-0x67F / 1537-1663	-
Node guard	Life/Node guard	1110	0x701-0x77F / 1793-1919	0x100C,0x100D,0x100E

The COB ID can be changed vis SDO.
The PDOs 3-8 do not have default values in Device Profile 402. The COB ID of these PDOs have to be set by the user with regard to the COB ID which are already use by the network.

9.7 Object Directory

All the CANopen objects relevant for the Network Adaptor are entered into the CANopen object directory. The object directory is divided into three different regions:

1) communication-specific profile region (index 0x1000 – 0x1FFF)
2) manufacturer-specific profile region (index 0x2000 – 0x5FFF)
3) standardized device profile region (0x6000 – 0x9FFF)

Region 1 thus contains the description of all the parameters particular to communication, the manufacturer-specific entries are described in region 2, and region 3 stores the objects for the device profile according to DS-401. Every entry in the object directory is identified by a 16 bit index.

9.7.1 Communication Profile Area

The following table contains all objects of the communication profile supported by the Network adaptor

Index	Sub-Index	Name	Type	Attribute	Default	Meaning
0x1000	0	device Type	unsigned32	ro	Value	Statement of device type
0x1001	0	error register	unsgined8	ro		Error register
0x1003	0	predefine error field	unsgined8	ro	0x00	Number of error states stored
	1	standard error filed	unsigned32	ro	0x00	Error state are stored
0x1005	0	COB-ID sync message	unsigned32	rw	0x00000080	Identifier of the Sync message
0x1006	0	communication cycle period	unsigned32	rw	0x00000000	Communication cycle period in. '0' if not used
0x1008	0	manufacturer device name	visible string	ro	CANOPEN…	Device name of the Adapter
0x1009	0	manufacturer hardware version	visible string	ro	-	H/W version description
0x100A	0	manufacturer software version	visible string	ro	-	Software version number
0x100C	0	guard time	unsigned16	rw	0x00C8	Interval between two guard telegrams. Is set by the NMTmaster.(mS)
0x100D	0	life time factor	unsgined8	rw	0x02	Life time factor * guard time = life time(watchdog for life guarding)
0x1010	0	store parameters	unsgined8	ro	1	Number of store options
	1	save all parameters	unsigned32	rw	0	Store all parameters
0x1011	0	restore default parameters	unsgined8	ro	1	Number of restore options
	1	restore all default	unsigned32	rw	0	Restore all default parameters
0x1014	0	COB-ID emergency message	unsigned32	rw	0x80+node ID	COB-ID of the emergency object
0x1015	0	inhibit time EMCY	unsigned16	rw		
0x1016	0	Consumer Heartbeat time	unsgined8	ro	1	Number of entries
	1	Consumer Heartbeat time	unsigned32	rw	0	Heartbeat time value
0x1017	0	producer Heartbeat time	unsigned16	rw		
0x1018	0	identity object	unsgined8	ro	4	
	1	manufacturer ID	unsigned32	ro	0x029D	
	2	product code	unsigned32	ro		
	3	revision number	unsigned32	ro		
	4	serial number	unsigned32	ro		

0x1400	0	receive PDO parameter	unsgined8	rw	5	number of following parameters
	1	COB-ID used by PDO	unsigned32	rw	0x200+nodeID	COB-ID RxPDO1
	2	transfer type	unsgined8	rw	254	Transmission type of the PDO
	3	inhibit time	unsigned16	rw	0x8813	Inhibit Ime of the PDO
	5	event timer	unsigned16	rw	0	Event time of the PDO
…	…	…				
0x1407	0	receive PDO parameter	unsgined8	rw	5	number of following parameters
	1	COB-ID used by PDO	unsigned32	rw	0x80000000	COB-ID RxPDO8
	2	transfer type	unsgined8	rw	254	Transmission type of the PDO
	3	inhibit time	unsigned16	rw	50000	Inhibit Ime of the PDO
	5	event timer	unsigned16	rw	0	Event time of the PDO
0x1600	0	receive PDO mapping	unsgined8	rw	0	Number of mapped objects.
	1	1. Object	unsigned32	rw	0x62000108	
	…	…				
	8	8. Object	unsigned32	rw	0x62000808	
…	…	…				
0x1607	0	receive PDO mapping	unsgined8	rw	0	Number of mapped objects.
	1	1. Object	unsigned32		0	
	…	…				
	8	8. Object	unsigned32	rw	0	
0x1800	0	transfer PDO parameter	unsgined8	rw	5	number of following parameters
	1	COB-ID used by PDO	unsigned32	rw	0x180+nodeID	COB-ID TxPDO1
	2	transfer type	unsgined8	rw	254	Transmission type of the PDO
	3	inhibit time	unsigned16	rw	5000	Inhibit Ime of the PDO
	5	event timer	unsigned16	rw	0	Event time of the PDO
…	…	…				
0x1807	0	transfer PDO parameter	unsgined8	rw	5	number of following parameters
	1	COB-ID used by PDO	unsigned32	rw	0x80000000	COB-ID TxPDO1
	2	transfer type	unsgined8	rw	254	Transmission type of the PDO
	3	inhibit time	unsigned16	rw	5000	Inhibit Ime of the PDO
	5	event timer	unsigned16	rw	0	Event time of the PDO
0x1A00		transfer PDO mapping	unsgined8	rw	0	Number of mapped objects.
	1	1. Object	unsigned32	rw	0x60000108	
	…	…				
	8	8. Object	unsigned32	rw	0x60000808	
…	…	…				
0x1A07	0	transfer PDO mapping	unsgined8	rw	0	Number of mapped objects.
	1	1. Object	unsigned32	rw	0	
	…	…				
	8	8. Object	unsigned32	rw	0	

- **Object 0x1000, Device Type**

The object indicates the implemented device profile. The CANopen Network Adaptor has implemented the Device Profile for Generic I/O Modules" (device profile No. 401). Moreover, in the index 0x1000 the value informs about the type of modules connected.

Format:

MSB LSB

4 byte	3 byte	2 byte	1 byte	0 byte
0x00	0x00	0000.4321 (bit)	0x01	0x91
		Device connect Number	Device Profile Number	

Bit	**Meaning**
1	1 = 1, if at least one digital input is connected.
2	2 = 1, if at least one digital output is connected.
3	3 = 1, if at least one analog input is connected.
4	4 = 1, if at least one analog output is connected.

- **Object 0x1001, Error Register**

This register contains internal errors. This register is also part of the emergency message

Format:

Bit	**Meaning**
0	General Error
1	Reserved
2	Reserved
3	Reserved
4	Communication
5	Device profile specific
6	Reserved
7	Manufacturer specific

In the event of an error, bit 0 is always set. Additional bits used specify the error in more detail.

- **Object 0x1003, Pre-defined Error Field**

The sub-index 0 contains the errors currently stored in the field. If a new error occurs, it will be entered in sub-index 1, and all errors already existing moved down by one sub-index. A max. Of 20 error entries are supported. Should more than 20 errors occur, each time the error contained in sub-index 20 is written over?

Format:

Bit31 Bit16	Bit15 Bit0
Additional Information	Error code

The additional information corresponds to the first 2 bytes of the additional code of the Emergency telegram. The error code coincides with the error code in the Emergency telegram.
The complete error memory is deleted by writing a .0" in sub-index 0.

- **Object 0x1005, COB-ID SYNC message**

The object defines the COB ID for the synchronization message.

Bit31	Bit11	Bit10	Bit0
Reserved (always 0)		COB-ID	

- **Object 0x1006, Communication Cycle Period**

The object defines the max. Time in µs for two subsequent SYNC messages.
The internal resolution is 2ms. If the value is 0, no SYNC monitoring is performed.

- **Object 0x1008, Manufacturer Device Name**

The object indicates the device name of the Network Adaptor.

- **Object 0x1009, Manufacturer Hardware Version**

The object indicates the current hardware version of the Network Adaptor

- **Object 0x100A, Manufacturer Software Version**

The object indicates the current software version of the Network Adaptor

- **Object 0x100C, Guard Time**

The object indicates the *Guarding Time* in milli-seconds. An NMT master cyclically interrogates the NMT slave for its status. The time between two interrogations is termed *Guard Time*.

- **Object 0x100D, Life Time Factor**

The life *Time Factor* is part of the *Node Guarding Protocol*. The NMT slave checks if it was interrogated within the *Node Life Time* (Guard time multiplied with the life time factor). If not, the slave works on the basis that the NMT master is no longer in its normal operation. It then triggers a *Life Guarding Event*.
If the node life time is zero, no monitoring will take place.

- **Object 0x1010, Store Parameters**

This object allows to permanently storing the settings made by the user. For this purpose, the signature .save" (lower case letters ASCII - MSB. 0x65 76 61 73 - LSB) must be written into the index 0x1010 sub index 1. The storing process runs in the background and takes approx. 2-3 seconds. When the storing process is finished, the SDO reply telegram is sent. Communication remains possible during storage by means of SDOs. An error message as a result of a new storage attempt only occurs, when the previous one was not yet finished.
It is also not possible to trigger the storage function for as long as .Restore" is active.
As soon as a setting is stored, the Emergency .Changed HW configuration. Is not sent any longer if the Network Adaptor is started up again without changing the module configuration.

Attention :

If following the storage of a configuration only the module ID is changed via the DIP switch, the saved configuration is continued to be used. In other words, all module ID specific entries in the object directory (objects that are module ID dependent and have the .rw" attribute) signal with the old values.

- **Object 0x1011, Restore default Parameters**

This object allows resetting the user stored parameters to the original default values.
Sub-indexes 2 and 3 are not supported.
The load command is processed in the background and takes approx. 2-3 seconds. When the performance is finished, the SDO reply message is sent. Communication can be continued during performance using SDOs. An error message is only tripped with another attempt to send a load command, if the previous one is not yet completed. It is also not possible to trigger a load command for as long as .Save" is active.
Sub-index 1 - Permanent entry of default parameters:
Writing the signature .load" (lower case letters ASCII - MSB 0x64 0x61 0x6F 0x6C LSB) into the index 0x1011 sub-index 1 entails loading of the standard factory settings after the following Power ON and each further Power On (until the next SAVE command is given).

- **Object 0x1014, COB-ID Emergency Object**

The object defines the COB ID for the EMCY message.

Bit31	Bit30 ... Bit11	Bit10 ... Bit0
0/1 valid/invalid	reserved (always 0)	COB-ID

If a new COB ID is to be entered, set bit 31 to 1 first, because standard DS301 does not allow to change a valid COB ID (Bit31=0).

- **Object 0x1015, Inhibit Time Emergency Object**

This object indicates the time in minutes which must be allowed to elapse prior to another Emergency to be sent. An entry of zero deactivates the delayed transmission.
Due to the fact that with delayed transmission the entries are entered in a queue, the max. number of Emergencies in quick succession is limited to the queue size (20 entries). If this number is exceeded, an Emergency is sent immediately indicating the overflow.
One time unit is 100µs.

- **Object 0x1016, Consumer Heartbeat Time**

This entry allows the monitoring of a maximum of 1modules. The system checks whether each module defined in this object has created a *Heartbeat* within the set time. If the set time was exceeded, a *Heartbeat-Event* is triggered. The *Heartbeat-Time* is entered in milli-seconds. The monitoring is deactivated, if the time value is 0.
Format:

	MSB		LSB
Bit	31-24	23-16	15-0
Value	Reserved	Node-ID	Heartbeat Time
Data Type	-	Unsigned8	Unsigned16

- **Object 0x1017, Producer Heartbeat Time**

The object defines the time between two Heartbeat messages sent in milliseconds. If the time is 0, no Heartbeat is sent. The Heartbeat transmission starts as soon as a value other than 0 is entered.

- **Object 0x1018, Identity Object**

The object specifies the device used.

- **Object 0x1400 ~ 0x1407, Receive PDO Communication Parameter**

This object is used to set the communication parameters of the RxPDOs. 8 RxPDOs are supported. The default COB IDs of the first four PDOs is reassigned according to the DS301 standard. All further PDOs are deactivated. If not all default PDOs are used (i.e. a smaller number of modules is connected), also the default PDOs not used are deactivated.

Format COB-ID:

Bit31	Bit30	Bit29 ... Bit11	Bit10 ... Bit0
0/1 valid/invalid	0/1 RTR allowed / not allowed	reserved (always 0)	COB-ID

If a new COB ID is to be entered, bit 31 must be set to 1 first, because the DS301 standard does not permit to change a valid COB ID (Bit31=0).

- **Object 0x1600 ~ 0x1607, Receive PDO Mapping Parameter**

This object is used to define the data, which is to be transmitted by means of the PDO.
Sub-index 0 contains the number of objects valid for the PDO.

Design 1. to 8. Object:

Bit31 Bit16	Bit15 Bit8	Bit7 Bit0
Index (Unsigned16)	Sub-Index (Unsigned8)	Size (Unsigned8)

Index: Index of the object to be transmitted
Sub-Index: Sub-index of the object to be transmitted
Size: Object size in bits Due to the fact that max. 8 bytes can be transmitted in a PDO, the sum of the valid object lengths must not exceed 64 (8Byte*8Bit)

- **Object 0x1800 ~ 0x1807, Transmit PDO Communication Parameter**

This object is used to set the communication parameters of the TxPDOs. 8 TxPDOs are supported. The default COB IDs of the first four PDOs is reassigned according to the DS301 standard. All other PDOs are de-activated. If not all default PDOs are used (i.e. a smaller number of modules is connected), also the default PDOs not used are de-activated.
Inhibit Time shows the min. time between two consecutive PDOs having the same COB ID. One time unit is 100us. The transmitted value is internally rounded to the next smaller millisecond.
If a new value is to be entered, the COB ID has to be set invalid (Bit 31 = 1), because the DS301 standard does not permit to enter a new time when the COB ID (Bit31=0) is valid.
The Event Timer defines the time after the elapse of which a PDO is sent, even if no change of the PDO data has occurred. Enter the time in milliseconds. The timer is re-started whenever an event occurs (change to the PDO data).
If the time is shorter than the inhibit time, a new event is generated once the inhibit time has elapsed! The event timer can only be used for the transmission types 254/255.

Attention :

An object entry can only be mapped in a **max. of 3 different** PDOs.

- **Object 0x1A00 ~ 0x1A07, Transmit PDO Mapping Parameter**

This object is used to define the data, which is transmitted using the PDO. Sub-index 0 contains the number of objects valid for the PDO.

Design 1. to 8. Object:

Bit31 Bit16	Bit15 Bit8	Bit7 Bit0
Index (Unsigned16)	Sub-Index (Unsigned8)	Size (Unsigned8)

Index: Index of the object to be transmitted
Sub-Index: Sub-index of the object to be transmitted
Size: Object size in bits Due to the fact that max. 8 bytes can be transmitted in a PDO, the sum of the valid object lengths must not exceed 64 (8Byte*8Bit)

9.7.2 Manufacturer Specific Profile Area

The following table shows all objects of the manufacturer profile supported by the Network Adaptor.

Index	Sub-Index	Name	Type	Attribute	Default	Meaning
0x2000	0	read memory data	unsigned8	ro	None	number of entries(slot number)
	1	read memory slot #01	visible data	ro	None	expansion slot#1 memory block read
	…	…				
	32	read memory slot #32	visible data	ro	None	expansion slot#32 memory block read
0x2020	0	input information	unsigned8	ro	1	
	1	digital input bit size	unsigned16	ro		expansion digital input all bit size
0x2200	0	write memory data	unsigned8	ro	None	number of entries(slot number)
	1	write memory slot #01	visible data	wo	None	expansion slot#1 memory block write
	…	…				
	32	write memory slot #32	visible data	wo	None	expansion slot#32 memory block write
0x2220	0	output information	unsigned8	ro	1	
	1	digital output bit size	unsigned16	ro		Expansion digital output all bit size
0x3000	0	read special input data	unsigned8	ro	None	number of entries
	1	special Input8 0H~7H	unsigned8	ro	None	1st special input block
	…	…				
	64	special Input8 1F8H~1FFH	unsigned8	ro	None	64st special input block
0x3200	0	write special output data	unsigned8	ro	None	number of entries
	1	special output8 0H~7H	unsigned8	ro	None	1st special output block
	…	…				
	64	special output8 1F8H~1FFH	unsigned8	ro	None	64st special output block
0x4500	0	FnBus communication register	unsgined8	ro	0x03	number of communication register block
	1	FnBus status	unsigend32	ro	0x00000000	FnBus Error monitoring ,field power check
	2	FnBus Data mode	unsigend8	rw	0x01	0: non compress mode 1: compress mode
	3	expansion IO active flag	unsgined32	rw	0x00000000	expansion module active flag(bit active)
0x4501	0	Range of Read Memory map	unsigend8	ro	None	number of expansion slot
	1	vision configure of slot #01	unsigned32	ro	None	number of read memory window slot #1
	…					
	32	vision configure of slot #01	unsigned32	ro	None	number of read memory window slot #32
0x4502	0	Expansion slot configuration	unsigend8	ro	None	number of expansion slot
	1	configuration of slot #01	visible data	rw	None	configuration parameter of slot #01
	…					
	32	configuration of slot #32	visible data	rw	None	configuration parameter of slot #32
0x4503	0	expansion module product code	unsigend8	ro	None	number of expansion module product code
	1	slot product code #01	unsigned32	ro	None	expansion module product code slot #1
	…					
	32	slot product code #32	unsigned32	ro	None	expansion module product code slot #32
0x4504	0	expansion module catalog code	unsigend8	ro	None	number of expansion module catalog code
	1	slot catalog code #01	unsigned32	ro	None	expansion module catalog code slot #1
	…					
	32	slot catalog code #32	unsigned32	ro	None	expansion module catalog code slot #32

- **Object 0x2020, Digital Inputs Bits Size Information.**

Index	Sub	Name	Type	Attribute	Default	Meaning
0x2020	0	input information	Unsigned8	ro	1	
	1	digital input bit size	Unsigned16	ro		expansion digital input all bit size

- **Object 0x2220, Digital Inputs Bits Size Information.**

Index	Sub	Name	Type	Attribute	Default	Meaning
0x2220	0	output information	Unsigned8	ro	1	
	1	digital output bit size	Unsigned16	ro		Expansion digital output all bit size

- **Object 0x3000, Special Modules, Inputs.**

  This object contains the process data of the special input modules. Sub-index 1 contains the first 8 special input channels from the left to the right, counted from starting with the Network Adaptor. Sub-index 2 the next etc.

- **Object 0x3200, Special Modules, Outputs.**

  This object contains the process data of the special output modules. Sub-index 1 contains the first 8 special output channels from the left to the right, counted from starting with the Network Adaptor. Sub-index 2 the next etc.

- **Object 0x4500, FnBus Communication Register**

Index	Sub	Name	Type	Attribute	Default	Meaning
0x4500	0	FnBus communication register	unsgined8	ro	0x03	number of communication register block
	1	FnBus status	unsigend32	ro	0x00000000	FnBus Error monitoring ,field power check
	2	FnBus Data mode	unsigend8	rw	0x01	0: non compress mode 1: compress mode
	3	expansion IO active flag	unsgined32	rw	0x00000000	expansion module active flag (bit active)

- **Object 0x4502, I/O Modules Parameter Configuration.**

Index	Sub	Name	Type	Attribute	Default	Meaning
0x4502	0	Expansion slot configuration	unsigend8	ro	None	number of expansion slot
	1	configuration of slot #01	visible data	ro	None	configuration parameter of slot #01
	…					
	32	configuration of slot #32	visible data	ro	None	configuration parameter of slot #32

- **Object 0x4503, I/O Modules Product code Register.**

Index	Sub	Name	Type	Attribute	Default	Meaning
0x4503	0	expansion module product code	unsigend8	ro	None	number of expansion module product code
	1	slot product code #01	unsigned32	ro	None	expansion module product code slot #1
	…					
	32	slot product code #32	unsigned32	ro	None	expansion module product code slot #32

- **Object 0x4504, I/O Modules Catalog code Resister.**

Index	Sub	Name	Type	Attribute	Default	Meaning
0x4504	0	expansion module catalog code	unsigend8	ro	None	number of expansion module catalog code
	1	slot catalog code #01	unsigned32	ro	None	expansion module catalog code slot #1
	…					
	32	slot catalog code #32	unsigned32	ro	None	expansion module catalog code slot #32

9.7.3 Standard Device Profile Area – DS401

The following table shows all objects of the standard profile DS401 supported by the Network Adaptor.

Index	Sub-Index	Name	Type	Attribute	Default	Meaning
0x6000	0	digital 1byte inputs	unsgined8	ro	None	Number of available 8bit digital input blocks
	1	input8 1~8h	unsgined8	rw	None	1st input block
	…	…				
	64	input8 1F9~200h	unsgined8	rw	None	64 input block
0x6200	0	digital 1byte outputs	unsgined8	ro	None	Number of available 8bit digital output blocks
	1	output8 1~8h	unsgined8	rw	None	1st output block
	…	…				
	64	output8 1F9~200h	unsgined8	rw	None	64 output block
0x6206	0	Error Mode Output 8-Bit	unsgined8	ro	None	Release of pre-defined error values of the 8 bit digital output data
	1	Error mode output 01h to 08h	unsgined8	rw	0xFF	
	…					
	64	Error mode output 1F9h to 200h	unsgined8	rw	0xFF	
0x6207	0	Error value output 8-bit	unsgined8	ro	None	Pre-defined error values of the 8 bit digital output data
	1	Error value output 01h to 08h	unsgined8	rw	0x00	
	…					
	64	Error value output 1F9h to 200h	unsgined8	rw	0x00	
0x6401	0	analog inputs	unsgined8	rw	None	Number of available analog input blocks
	1	analog input16 01h	unsigned16	rw	None	1st input block
	…	…				
	32	analog input16 20h	unsigned16	rw	None	32 input block
0x6411	0	analog outputs	unsgined8	rw	None	Number of available analog output blocks
	1	analog output16 01h	unsigned16	rw	None	1st output block
	…	…				
	32	analog output8 20h	unsigned16	rw	None	32 output block
0x6443	0	Analogue output error mode	unsgined8	ro	None	Release of pre-defined error values of the 16 bit output data
	1	Error mode analogue output 01h	unsgined8	rw	0x01	
	…					
	32	Error mode analogue output 20h	unsgined8	rw	0x01	
0x6444	0	Analogue output error value integer	unsgined8	ro	None	Value in the event of an error of the 16 bit output data
	1	Analogue output error value 01h	unsigned16	rw	0x00	
	…					
	32	Analogue output error value 20h	unsigned16	rw	0x00	

- **Object 0x6000, Digital Inputs**

This object contains the process data of the digital input modules. Sub-index 1 contains the first 8 digital input channels from the left to the right, counted from starting with the Network Adaptor. Sub-index 2 the next etc.

- **Object 0x6200, Digital Outputs**

This object contains the process data of the digital output modules. Sub-index 1 contains the first 8 digital output channels from left to right, counting starting from the Network Adaptor. Sub-index 2 the next etc.

- **Object 0x6206, Error Mode Output 8-Bit**

This object defines whether the outputs change to a pre-defined error status in the event of an error (i.e. adaptor changes to the *Stopped* status, Node guarding has failed,) (see object 0x6207). If the error is remedied, the outputs remain in their momentary status, i.e. the set error status of the output channels remains unchanged.

0 = Outputs remain unchanged (per channel)
1 = Outputs change to a pre-defined error status (per channel)

- **Object 0x6207, Error Value Output 8-Bit**

This object is used to define the values, which the outputs should assume in the event of an error. Prerequisite being that the corresponding bit in object 0x6206 is set.

0 = Output to 0 (per channel)
1 = Output to 1 (per channel)

Example: Index 0x6206 sub-index 0 = 1, sub-index 1 = 65 = 0x41
Index 0x6207 sub-index 0 = 1 sub-index 1 = 33 = 0x21
Channel 1 is set to 1, channel 7 is set to 0, and all other output channels remain unchanged in the event of an error

- **Object 0x6401, Analog Inputs 16 Bit**

This object contains the process data of the analog input modules. Sub-index 1 contains the first analog input channel from left to right, counting starting with the Network Adaptor. Sub-index 2 the second, etc.

- **Object 0x6411, Analog Outputs 16 Bit**

This object contains the process data of the analog output modules. Sub-index 1 contains the first analog output channel from left to right, counting starting with the Network Adaptor. Sub-index 2 the second, etc.

- **Object 0x6443, Analog Output Error Mode**

  This object is used to define whether the outputs change to a pre-defined error status (see object 0x6444) in the event of an error (i.e. adaptor changes to the *Stopped* status, Node guarding has failed,). Once the error is remedied, the outputs retain their momentary status, i, e. the set error status of the output channels remains unchanged.
  All analog outputs that are not covered by the object 0x6444 are always set to 0 in the event of an error.
  0 = The output remains unchanged
  1 = The output changes to a pre-defined error status

- **Object 0x6444, Analog Output Error Value Integer**

  This object is used to define values that they are to assume in the event of an error. Prerequisite being that the corresponding bit is set in object 0x6443

10. TROUBLE SHOOTING

How to diagnose by LED indicator

LED Status	Cause	Action
All LED turns off	-No power	-Check main power Cable
	-System power is not supplied.	-Contact Sales team and send module for repair.
MOD LED flashes green	-Failure of initialization EEPROM parameter.	-Contact Sales team and send module for repair.
MOD LED flashes red	-Excess of expansion slot - Excess of IO size - Wrong IO composition -Occurrence of EEPROM checksum error	-Use expansion slot up to 32. -Compose that IO total size is not excess. -Check composition I/O Module
MOD LED is red	-Wrong address ID -Occurrence critical error in firmware	-Contact Sales team and send module for repair.
I/O LED turns off	-Failure of realization expansion Module -None expansion Module	-Check connector status both NA series and expansion module.
I/O LED flashes red	Failure of configuration baud rate	-Check communication cable with Master -Check power for master.
	Failure of initialization I/O	-Use expansion slot up to 32. -Compose that IO total size is not excess.
		NA series notice unidentified expansion module ID. Check status of expansion module.
I/O LED is red	Failure of exchanging I/O data	Check status of expansion IO connection.
NET LED turns off	Failure of communication with Master	Check main power for master and communication cable.
NET LED flashed green	Failure of exchanging data with master	Check status in software for Master configuration.
NET LED is red	Communication connecting losts	Check BUS line cable for connection with master.
		Check duplication address.

How to diagnose when device couldn’t communicate network

Inspection of wrong or omission cable connection

- Check status of cable connection for each node.
- Check that all color matches between connector and cable.
- Check wire omission.

Terminator resistor

- If terminator resistor is not installed, install terminator resistor
- Check location of terminator resistor

Configuration of Node address

- Check duplication node address.

Configuration of Master

- Check configuration of master
- Check whether to do download or don’t
- Check composition is right as below Configuration of communication baud rate I/O size Configuration of each node

Ground and environment

- Check ground is contacted
- Check environment factor (temperature, humidity, etc.) is in less than regular limit

Appendix A

A.1 Product List

No.	ST-Number	Description	Module Id (hex)	Catalog Number	Product Code
1	ST-1214	4-sinking input, 24Vdc	03	00 03 00 41	83 C0 40 01
2	ST-1224	4-sourcing input, 24Vdc	04	00 04 00 41	83 C0 40 01
3	ST-1218	8-sinking input, 24Vdc	07	00 07 00 41	00 C0 40 01
4	ST-1228	8-sourcing input, 24Vdc	08	00 08 00 41	00 C0 40 01
5	ST-121F	16-sinking input, 24Vdc	13	00 13 01 41	01 C0 40 01
6	ST-122F	16-sourcing input, 24Vdc	14	00 14 01 41	01 C0 40 01
7	ST-1314	4-sinking input, 48Vdc	05	00 05 00 41	83 C0 40 01
8	ST-1324	4-sourcing input, 48Vdc	06	00 06 00 41	83 C0 40 01
9	ST-1804	4-ac input, 110Vac	09	00 09 00 41	83 C0 40 01
10	ST-1904	4-ac input, 220Vac	0A	00 0A 00 41	83 C0 40 01
11	ST-2314	4-sinking output, 24Vdc 0.5A	0E	00 0E 00 81	C0 83 80 01
12	ST-2324	4-sourcing output, 24Vdc 0.5A	10	00 10 00 81	C0 83 80 01
13	ST-2318	8-sinking output, 24Vdc 0.5A	11	00 11 00 81	C0 00 80 01
14	ST-2328	8-sourcing output, 24Vdc 0.5A	12	00 12 00 81	C0 00 80 01
15	ST-221F	16-sinking output, 24Vdc 0.3A	15	00 15 01 81	C0 01 80 01
16	ST-222F	16-sourcing output, 24Vdc 0.3A	16	00 16 01 81	C0 01 80 01
17	ST-2414	4-sinking output, diag, 24Vdc 0.5A	37	37 00 00 C1	83 83 C0 01
18	ST-2424	4-sourcing output, diag, 24Vdc 0.5A	38	38 00 00 C1	83 83 C0 01
19	ST-2514	4-sinking output, diag, 24Vdc 2A	35	35 00 00 C1	83 83 C0 01
20	ST-2524	4-sourcing output, diag, 24Vdc 2A	36	36 00 00 C1	83 83 C0 01
21	ST-2742	2-relay output, 230Vac 2A	0B	00 0B 00 81	C0 81 80 01
22	ST-2852	2-triac output, 120Vac 0.5A	0C	00 0C 00 81	C0 81 80 01
23	ST-3114	4-current analog input, 0~20mA, 12bit	1C	00 1C 43 41	43 C0 60 03
24	ST-3134	4-current analog input, 0~20mA, 14bit	1E	00 1E 43 41	43 C0 60 03
25	ST-3214	4-current analog input, 4~20mA, 12bit	1D	00 1D 43 41	43 C0 68 03
26	ST-3234	4-current analog input, 4~20mA, 14bit	1F	00 1F 43 41	43 C0 68 03
27	ST-3424	4-voltage analog input, 0~10V, 12bit	20	00 20 43 41	43 C0 60 03
28	ST-3444	4-voltage analog input, 0~10V, 14bit	22	00 22 43 41	43 C0 60 03
29	ST-3524	4-voltage analog input, -10~10V, 12bit	21	00 21 43 41	43 C0 60 03
30	ST-3544	4-voltage analog input, -10~10V, 14bit	23	00 23 43 41	43 C0 60 03
31	ST-3624	4-voltage analog input, 0~5V, 12bit	24	00 24 43 41	43 C0 60 03
32	ST-3644	4-voltage analog input, 0~5V, 14bit	25	00 25 43 41	43 C0 60 03
33	ST-3702	2-RTD/Resistance input	28	00 28 41 41	41 C0 68 03
34	ST-3802	2-Thermocouple/mV input	2A	00 2A 41 41	41 C0 68 03
35	ST-4112	2-current analog output, 0~20mA, 12bit	2C	00 2C 41 81	C0 41 A0 03
36	ST-4212	2-current analog output, 4~20mA, 12bit	2D	00 2D 41 81	C0 41 A0 03
37	ST-4422	2-voltage analog output, 0~10Vdc, 12bit	2E	00 2E 41 81	C0 41 A0 03
38	ST-4522	2-voltage analog output, -10~10Vdc, 12bit	2F	00 2F 41 81	C0 41 A0 03
39	ST-4622	2-voltage analog output, 0~5Vdc, 12bit	30	00 30 41 81	C0 41 A0 03
40	ST-5101	1 Channel, High Speed Counter, 5Vdc	34	34 05 01 C1	05 01 D0 03
41	ST-5111	1 Channel, High Speed Counter, 24Vdc	39	39 5 01 C1	05 01 D0 03

A.2 Digital Data Bit size Information.

: The index is can be access via SDO.

A.2.1 Input bit size information.

: All digital input data are counted.

Index	Sub-Index	Decimal Byte	Data Type	Description
0x2020	0x01	Byte 00	unsigned8	All Digital input bit size

Ex) Data Read: Id=RxSDO DLC=8; Data=40 20 20 00 xx xx xx xx

A.2.2 Output bit size information.

A.2.2.1 all digital output data are counted.

Index	Sub-Index	Decimal Byte	Data Type	Description
0x2220	0x01	Byte 00	unsigned8	All Digital output bit size

Ex) Data Read: Id=RxSDO DLC=8; Data=40 20 22 00 xx xx xx xx

A.3 Special IO Data Block.

A.3.1 Special Input Block

Index	Sub-Index	Decimal Byte	Data Type	Description
0x3000	0x01	Byte 00	unsigned8	0h~7h Special input data
	0x02	Byte 01	unsigned8	8h~15h Special input data
	. . .	. . .	. . .	. . .
	0x64	Byte64	unsigned8	1F8h~1FFh Special input data

*user set PDO mapping.

A.3.2 Special Output Block.

Index	Sub-Index	Decimal Byte	Data Type	Description
0x3200	0x01	Byte 00	unsigned8	0h~7h Special output data
	0x02	Byte 01	unsigned8	8h~15h Special output data
	. . .	. . .	. . .	. . .
	0x64	Byte64	unsigned8	1F8h~1FFh Special output data

*user set PDO mapping.

A.4 FnBus communication register Format

: The Index 0x4500 is can be access via SDO.

A.4.1 FnBus Error monitor data format

: This object is FnBus state.

Index	Sub-Index	Decimal Byte	Data Type	Description
0x4500	0x01	Byte 00	unsgined8	FnBus Error Code
		Byte 01	unsigend32	Error Slot number
		Byte 02	unsigend8	Reserve
		Byte 03	unsgined32	Field Power state 0x80 : not supply, 0x00 : supply

Ex) Data Read: Id=RxSDO DLC=8; Data=40 00 45 01 xx xx xx xx

A.4.2 FnBus Data Mode

: The Process Image is can be changed via this object.

Index	Sub-Index	Decimal Byte	Data Type	Description
0x4500	0x02	Byte 00	unsigned8	0 : non-compress mode
				1 : compress mode

Ex) Data Read: Id=RxSDO DLC=8; Data=40 00 45 02 xx xx xx xx
Data write: Id=RxSDO DLC=8; Data=2F 00 45 02 01 xx xx xx (compress mode set)

A.4.3 Expansion module active flag data format

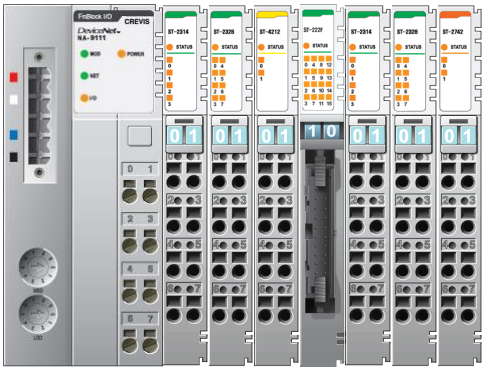
: The IO Slot is deactivated via the bit flag.

Index	Sub-Index	Data type	Decimal Bit	Description
0x4500	0x03	Unsigned32	Bit 00	Activate/Deactivate flag for slot position #1 (0:Active, 1:Deactivate)
			Bit 01	Activate/Deactivate flag for slot position #2 (0:Active, 1:Deactivate)
			Bit 03	Activate/Deactivate flag for slot position #3 (0:Active, 1:Deactivate)
			. . .	. . .
			Bit 30	Activate/Deactivate flag for slot position #31 (0:Active, 1:Deactivate)
			Bit 31	Activate/Deactivate flag for slot position #32 (0:Active, 1:Deactivate)

* bit position – IO slot poison.
Ex) Data Read: Id=RxSDO DLC=8; Data=40 00 45 03 xx xx xx xx
Data write: Id=RxSDO DLC=8; Data=2B 00 45 03 01 00 xx xx (Slot 1 Deactivated)

A.5 Module product code Read example

For Example



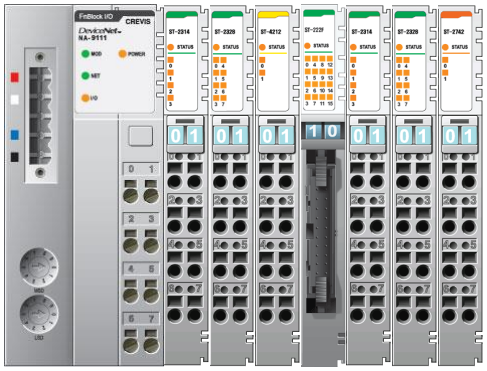
Slot Address	Module Description
0	CANopen Adapter
1	4-discrete output(ST-2424)
2	8-discrete output(ST-2318)
3	2-analog output(ST-4112)
4	16-discrete output(ST-222F)
5	4-discrete output(ST-2314)
6	8-discrete output(ST-2328)
7	2-realy output(ST-2742)

Object

Index	Sub-Index	Data	Description	SDO protocol
0x4503	0x00	07	Total expansion slot number	Id=RxSDO DLC=8; Data=40 03 45 00 xx xx xx xx
0x4503	0x01	83 83 C0 01	Slot#1 Product code	Id=RxSDO DLC=8; Data=40 03 45 01 xx xx xx xx
0x4503	0x02	C0 00 80 01	Slot#2 Product code	Id=RxSDO DLC=8; Data=40 03 45 02 xx xx xx xx
0x4503	0x03	C0 41 A0 03	Slot#3 Product code	Id=RxSDO DLC=8; Data=40 03 45 03 xx xx xx xx
0x4503	0x04	C0 01 80 01	Slot#4 Product code	Id=RxSDO DLC=8; Data=40 03 45 04 xx xx xx xx
0x4503	0x05	C0 83 80 01	Slot#5 Product code	Id=RxSDO DLC=8; Data=40 03 45 05 xx xx xx xx
0x4503	0x06	C0 00 80 01	Slot#6 Product code	Id=RxSDO DLC=8; Data=40 03 45 06 xx xx xx xx
0x4503	0x07	C0 81 80 01	Slot#7 Product code	Id=RxSDO DLC=8; Data=40 03 45 07 xx xx xx xx

A.6 Module Catalog code Read example

For Example



Slot Address	Module Description
0	CANopen Adapter
1	4-discrete output(ST-2424)
2	8-discrete output(ST-2318)
3	2-analog output(ST-4112)
4	16-discrete output(ST-222F)
5	4-discrete output(ST-2314)
6	8-discrete output(ST-2328)
7	2-realy output(ST-2742)

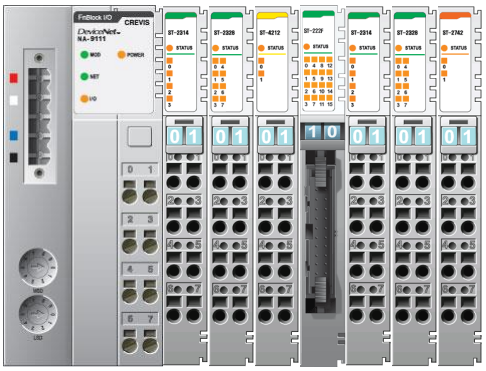
object

Index	Sub-Index	Data	Description	SDO protocol
0x4504	0x00	07	Total expansion slot number	Id=RxSDO DLC=8; Data=40 04 45 00 xx xx xx xx
0x4504	0x01	38 00 00 C1	Slot#1 Product code	Id=RxSDO DLC=8; Data=40 04 45 01 xx xx xx xx
0x4504	0x02	00 11 00 81	Slot#2 Product code	Id=RxSDO DLC=8; Data=40 04 45 02 xx xx xx xx
0x4504	0x03	00 2C 41 81	Slot#3 Product code	Id=RxSDO DLC=8; Data=40 04 45 03 xx xx xx xx
0x4504	0x04	00 16 01 81	Slot#4 Product code	Id=RxSDO DLC=8; Data=40 04 45 04 xx xx xx xx
0x4504	0x05	00 0E 00 81	Slot#5 Product code	Id=RxSDO DLC=8; Data=40 04 45 05 xx xx xx xx
0x4504	0x06	00 12 00 81	Slot#6 Product code	Id=RxSDO DLC=8; Data=40 04 45 06 xx xx xx xx
0x4504	0x07	00 0B 00 81	Slot#7 Product code	Id=RxSDO DLC=8; Data=40 04 45 07 xx xx xx xx

A.7 Configuration Parameter

A.7.1 Configuration parameter format

For Example



Slot Address	Module Description
0	CANopen Adapter
1	4-discrete output(ST-2424)
2	8-discrete output(ST-2318)
3	2-analog output(ST-4112)
4	16-discrete output(ST-222F)
5	4-discrete output(ST-2314)
6	8-discrete output(ST-2328)
7	2-realy output(ST-2742)

Object

Index	Sub-Index	Data	Description
0x4502	0x00	7	Number of entries
0x4502	0x01	Unsigned 16	Slot#1 Parameter 2 byte
0x4502	0x02	Unsigned 16	Slot#2 Parameter 2 byte
0x4502	0x03	Unsigned 48	Slot#3 Parameter 6 byte
0x4502	0x04	Unsigned 32	Slot#4 Parameter 4 byte
0x4502	0x05	Unsigned 16	Slot#5 Parameter 2 byte
0x4502	0x06	Unsigned 16	Slot#6 Parameter 2 byte
0x4502	0x07	Unsigned 16	Slot#7 Parameter 2 byte

Ex) output slot #2(ST2318) all channel setting of the hold last state.
SDO protocol: Id=RxSDO DLC=8; Data= 2b 02 45 02 FF 00 00 00

Appendix B

Bus cable and termination resistors

The cables, connectors, and termination resistors used in CANopen networks shall meet the requirements defined in ISO 11898. In addition, here are given some guidelines for selecting cables and connectors.

The table below shows some standard values for DC parameters for CANopen networks with less than 64 nodes:

Bus length [m]	Bus cable (1)		Termination resistance [Ω]	Baud rate [Kbit/s]
	Length-related Resistance [m/m]	Cross-section [mm^2]		
0 ... 40	70	0.25 … 0.34	124	1000 at 40m
40 … 300	< 60	0.34 … 0.6	150 … 300	> 500 at 100m
300 ... 600	< 40	0.5 ... 0.6	150 ... 300	> 100 at 500 m
600 ... 1000	< 26	0.75 ... 0.8	150 ... 300	> 50 at 1 km

(1) Recommended cable AC parameters: 120- impedance and 5-ns/m specific line delay

For drop cables a wire cross-section of 0.25 to 0.34 mm2 would be an appropriate choice in many cases.

Besides the cable resistance, there should also be considered the real resistance of the connectors, if calculating the voltage drop. The transmission resistance of one connector should be in the range of 2.5 to 10 m .

With the assumed values for

Minimum dominant value Vdiff.out.min = 1.5 V

Minimum differential input resistance Rdiff.min = 20 k

Requested differential input voltage Vth.max = 1.0 V

Minimum termination resistance RT.min = 118

The maximum wiring length is given for different bus cables and different number of connected bus nodes in the following table.

Wire cross-Section [mm^2]	Maximum length [m] (1)			Maximum length [m] (2)		
	n = 32	n = 64	n = 100	n = 32	n = 64	n = 100
0.25	200	170	150	230	200	170
0.5	360	310	270	420	360	320
0.75	550	470	410	640	550	480

(1) Safety margin of 0.2 (2) safety margin of 0.1

Note: If driving more than 64 nodes and/or more than 250 m bus length the accuracy of the VCC supply voltage for the ISO 11898 transceiver is recommended to be 5% or better. You also have to consider the minimum supply voltage of at least 4.75V when driving 50 load, i.e. 64 bus nodes, and at least 4.9V when driving 45 loads, i.e. 100 bus nodes.