Programmable Control Products

PACSystems* RSTi
CANopen Network Adapter

User's Manual, GFK-2807

January 2013

Warnings, Cautions and Notes as Used in this Publication

Warning

Warning notices are used in this publication to emphasize that hazardous voltages, currents, temperatures, or other conditions that could cause personal injury exist in this equipment or may be associated with its use.

In situations where inattention could cause either personal injury or damage to equipment, a Warning notice is used.

Caution

Caution notices are used where equipment might be damaged if care is not taken.

Note: Notes merely call attention to information that is especially significant to understanding and operating the equipment.

This document is based on information available at the time of its publication. While efforts have been made to be accurate, the information contained herein does not purport to cover all details or variations in hardware or software, nor to provide for every possible contingency in connection with installation, operation, or maintenance. Features may be described herein which are not present in all hardware and software systems. GE Intelligent Platforms assumes no obligation of notice to holders of this document with respect to changes subsequently made.

GE Intelligent Platforms makes no representation or warranty, expressed, implied, or statutory with respect to, and assumes no responsibility for the accuracy, completeness, sufficiency, or usefulness of the information contained herein. No warranties of merchantability or fitness for purpose shall apply.

If you purchased this product through an Authorized Channel Partner, please contact the seller directly.

General Contact Information

Online technical support and GlobalCare	http://www.ge-ip.com/support
Additional information	http://www.ge-ip.com/
Solution Provider	solutionprovider.ip@ge.com

Technical Support

If you have technical problems that cannot be resolved with the information in this guide, please contact us by telephone or email, or on the web at www.ge-ip.com/support

Americas

Online Technical Support	www.ge-ip.com/support
Phone	1-800-433-2682
International Americas Direct Dial	1-780-420-2010 (if toll free 800 option is unavailable)
Technical Support Email	support.ip@ge.com
Customer Care Email	customercare.ip@ge.com
Primary language of support	English

Europe, the Middle East, and Africa

Online Technical Support	www.ge-ip.com/support
Phone	+800-1-433-2682
EMEA Direct Dial	+352-26-722-780 (if toll free 800 option is unavailable or if dialing from a mobile telephone)
Technical Support Email	support.emea.ip@ge.com
Customer Care Email	customercare.emea.ip@ge.com
Primary languages of support	English, French, German, Italian, Czech, Spanish

Asia Pacific

|	
Online Technical Support	www.ge-ip.com/support
Phone	+86-400-820-8208
	+86-21-3217-4826 (India, Indonesia, and Pakistan)
Technical Support Email	support.cn.ip@ge.com (China)
	support.jp.ip@ge.com (Japan)
	support.in.ip@ge.com (remaining Asia customers)
Customer Care Email	customercare.apo.ip@ge.com
	customercare.cn.ip@ge.com (China)

Contents

Contents

1.   Introduction

The PACSystems RSTi Network Interface and I/O family provides a cost effective, modular distributed I/O system. The RSTi network is ideally suited for distributed applications such as water/wastewater, process control, packaging and assembly. You can easily add RSTi modules to the system to build functional remote I/O stations to meet your application requirements.
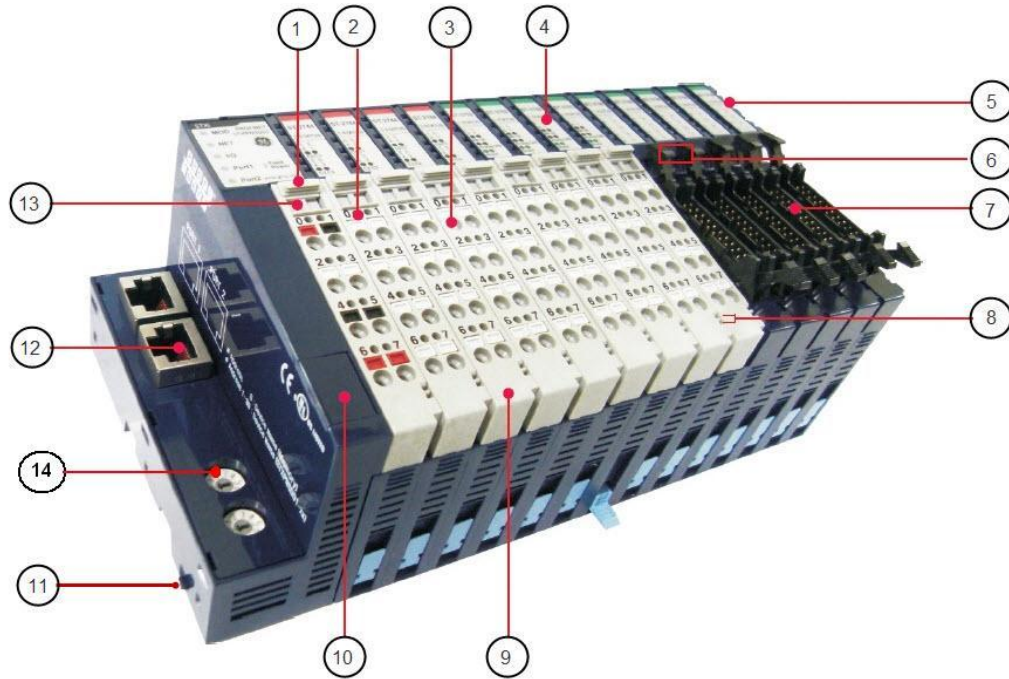
Sr. Number	Label
1	Removable Switch of Terminal Block
2	Tester Pin Hole
3	Screw less Connection System
4	I/O Status Display LED
5	System-Data Pin (6 Pins)
6	Module Number Marking (Header Type)
7	Header Type Module (16 points)
8	Field Power Pin (2 pins)

Sr. Number	Label
9	RTB (Removable Terminal Block)
10	Reserved communication Port (Useful to only manufacturer)
11	PUSH Lock for DIN rail
12	Fieldbus Connector (actual connector depends upon the type of network adapter refer Section 3.0)
13	Module Number Marking (on the Removable terminal Block)
14	Node ID or station address setting (either rotary or dip switches depends upon the type of network adapter)

A set of interconnected RSTi modules can be chosen to suit the application and connected as a slave on a CANopen network. An RSTi Network Adapter provides the interface between the network and the RSTi modules. The Network Adapter and I/O modules selected for an application constitute an I/O station.

1.2 I/O Station Capacity

■ Up to 32 IO devices can be connected to a Network Adapter .The power consumption of all the modules in the node should be calculated and ensured that it does not exceed the capacity of Network Adapter and power modules.

1.3 Installation

The Network Adapter must be connected to the left of the other RSTi modules in the I/O station.

Within the RSTi station the bus connection, power supply, and power distribution are completed by connecting modules together on the DIN rail. Sensors and actuators are wired to the RSTi modules using spring clamp terminals on the module's removable terminal strips. These terminal strips can be keyed so that they cannot be accidentally swapped. If a module need to be replaced, just remove the terminal strip from the module and there is no need for the wiring to be disturbed..

1.4 Configuration

The CANopen Network Adapters can be configured using IOGuidePro tool, for more information refer to the help section of IOGuidePro tool.

1.5 Features

■ Modules can be easily installed and connected without tools.

■ Flexible and modular structure allows I/O stations to be easily expanded.

■ A comprehensive selection of I/O modules supports a wide range of applications.

■ Small removable terminal blocks conserve panel space and save time making system connections.

■ Module-based diagnostic functions

■ The amount of costly parallel wiring is reduced. Within a station, voltage and data routing can be carried out without additional wiring, reducing the cabinet space needed.

■ Different parts of the system can be operated independent of one another. This means that pretests can be carried out when the system is set up and that the whole system can be adapted and expanded.

1.6 List of RSTi Network adapters

RSTi CANopen Network Adapters

■ **STXCAN001   CANopen network adapter**

1.7 PACSystems Documentation

■ **PACSystems RSTi I/O Manual – GFK 2745**
■ **Information about CANopen**

For detailed information about CANopen, contact the CAN in Automation e.V. (CiA) organization: http://www.can-cia.org/

2.   Installation

<div style="text-align: center;">

Warning

</div>

Installing or removing modules or wiring with power applied to the system or field wiring can cause an electrical arc. This can result in unexpected and potentially dangerous action by field devices. Arcing is an explosion risk in hazardous locations. Be sure that the area is non-hazardous or remove power appropriately before installing or removing modules or wiring.

Potentially dangerous voltages are present on a module's terminals, even when system power is turned off. Field power must be turned off when installing or removing a terminal block assembly.

Personnel, who install, operate and maintain automation systems that contain these products must be trained and qualified to perform those functions.

Overloading power modules or Network adapter can result into electric arc and damage to modules.

<div style="text-align: center;">

Caution

</div>

Check the rated voltage and terminal array before wiring.

Ensure that specified environmental conditions are not exceeded. Avoid placing the module in direct sunlight.

Review module specifications carefully, and ensure that input and output connections are made in accordance with the specifications.

Use specified cables for wiring.

Field power isolators must be used according to the requirements of the 5VDC/24VDC/48VDC or AC voltage modules used in the system.

If system power consumption exceeds the power limits, use system power expansion modules.

Power supplies for system power and field power must be supplied from separate sources.

2.1   Module Mounting

2.1.1   How to Mount on DIN Rail

1.   Press down the module lightly on the DIN Rail until it clicks & locks.
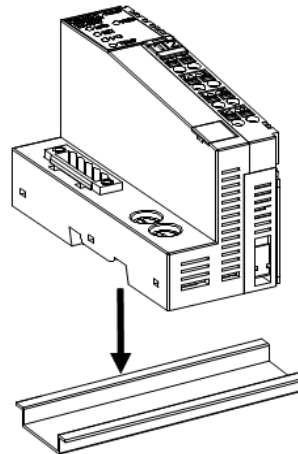
Figure 1: Pressing down the module lightly

2.   . You can use the PUSH lock on the side of module as a second locking mechanism. This lock will maintain module intact to DIN rail.
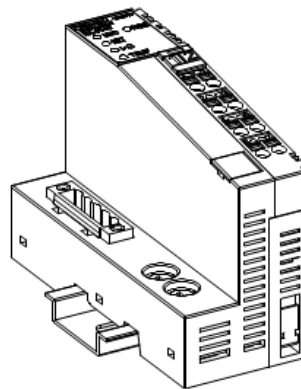
Figure 2: PUSH lock as a second locking mechanism

2.1.2   How to dismount from DIN Rail

1. Pull down the locking mechanism by using small flat screw driver as in the following figures.
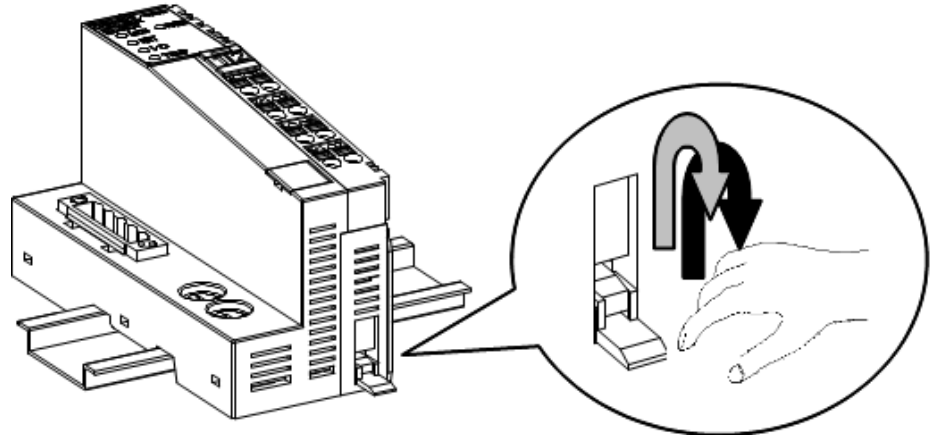
Figure 3: Pulling down the lock

2. Push the lock on the side of module, pull up the module to remove from the DIN rail.
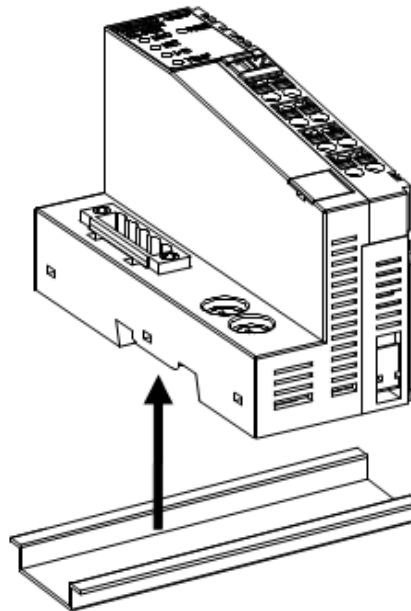
Figure 4: Pulling up the module

2.2   Installing and Removing Components

To plug in the module use a small-bladed screwdriver and push down the locking lever located at bottom of the module. Install the module on DIN rail firmly; push up the locking lever to lock. To pull out the RSTi module, push down the locking lever rail.
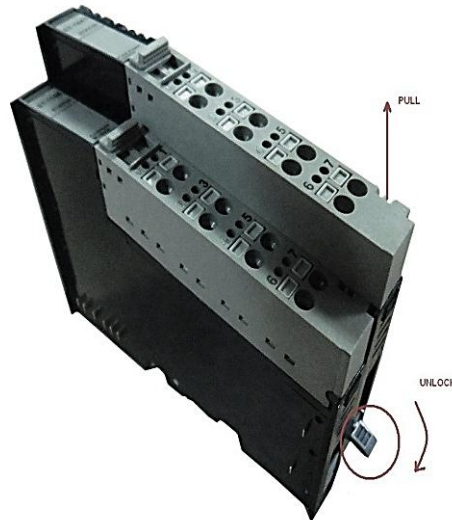
Figure 5: Removing RSTi module from DIN rail

2.3   Internal Bus/Field Power Contacts

Communication between the Network adapters and the IO 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 pins and 2 field power pins.
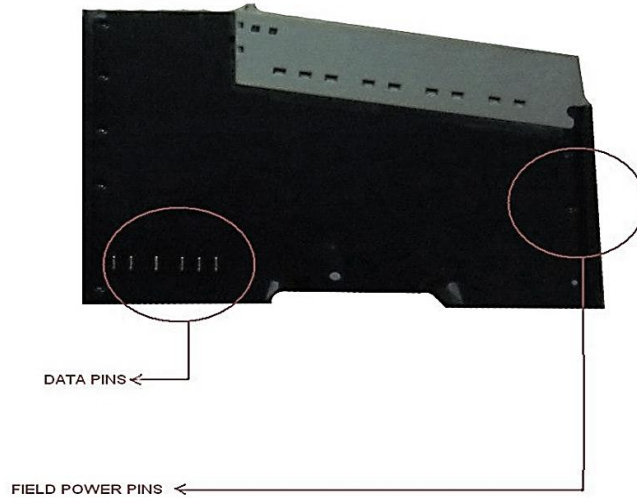
Figure 6: Internal bus with 6 data pins and 2 field power pins

Warning

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

2.3.1   RSTi Bus Data Pin & Field Power Pin Description

Table 1: RSTi Bus Pin Name and Description

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)

2.3.2 RSTi Data Bus System
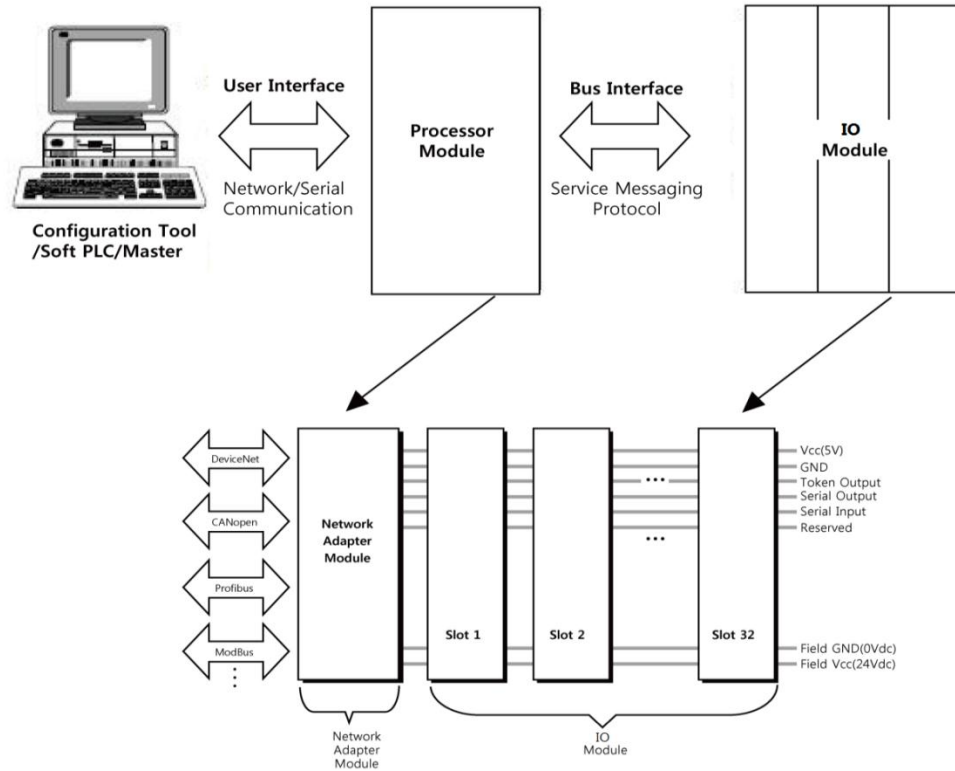
Figure 7: Interface between different modules

- **Network Adapter Module: The Network Adapter Module forms the link between the field bus and the field devices through IO Modules. The connection to different field bus systems can be established by each of the corresponding Network Adapter Modules: PROFIBUS, CANopen, DeviceNet, Ethernet/IP, CC-Link, MODBUS/Serial, MODBUS/TCP, PROFINET etc.**
- **IO Module: The IO Modules are supported by a variety of input and output field devices. There are digital and analog input/output modules and special function modules.**
- **Two types of Bus Message**

  Service Messaging
  I/O Messaging

3.   CANopen Network Adapter

3.1   STXCAN001 (CANopen)

3.1.1   Interface

The following figure shows the interface diagram for STXCAN001
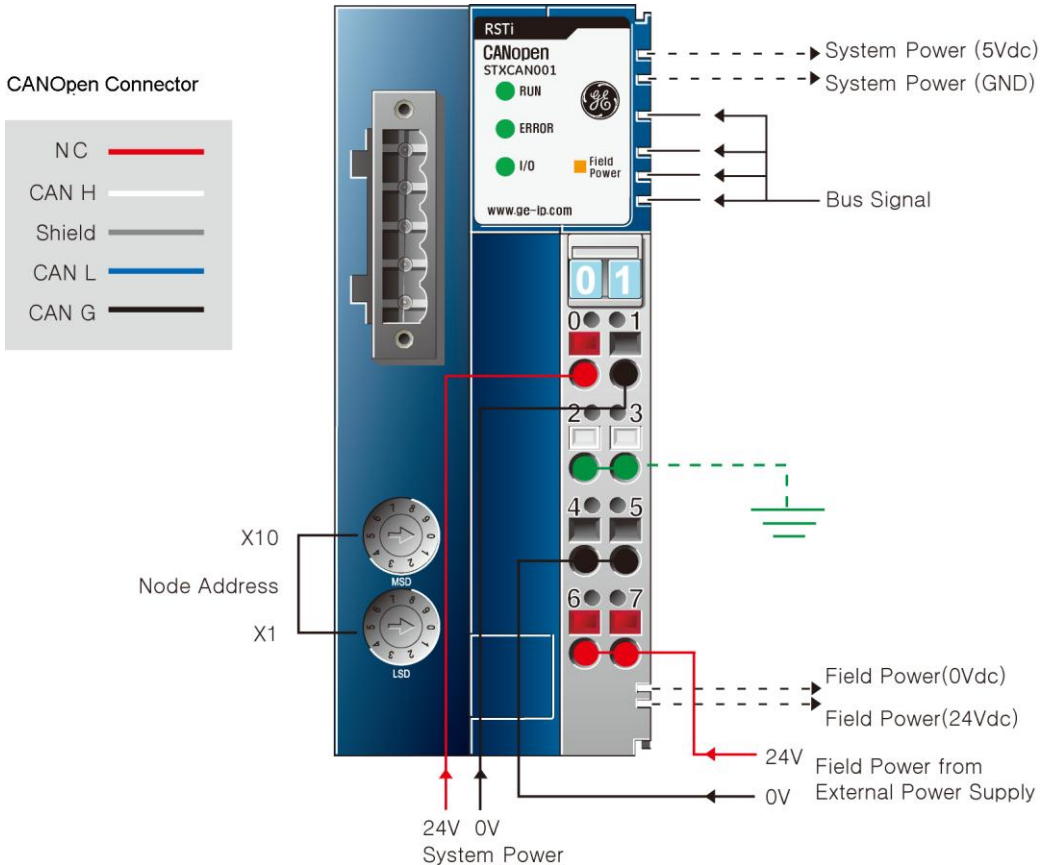
Figure 8: CANopen Network Adapter module: STXCAN001

The following table lists the pin numbers and their description for STXCAN001

Table 2: STXCAN001: Pin Description

Pin Number	Description	Pin Number	Description
0	System power 24v	1	System power 0 v
2	Ground	3	Ground
4	Field power 0v	5	Field power 0v
6	Field power 24v	7	Field power 24v

3.1.2 Specifications

Table 3: Interface specifications

Items	Specification
Number of Nodes	99
Max. IO Module	32
Max. Input Size	32words (64bytes)
Max. Output Size	32words (64bytes)
Max. Length Bus Line	Depending on Baud rate
Communication Rate	10 Kbit/s ---1 Mbit/s
Number of PDOs available	8 Transmit PDOs / 8 Receive PDOs
Number of SDOs available	1 Standard SDOs
Indicators	4 LEDs 1 Green, CAN-RUN Status 1 Red, CAN-ERR Status 1 Green/Red Bus Status 1 Green, Field Power Status

Table 4: General specifications

Items	Specification
System Power	Supply voltage : 24Vdc nominal Supply voltage range : 11~28.8Vdc
Power Dissipation	100mA typical @24Vdc
Current for I/O Module	1.5A @5Vdc
Isolation	Network to Logic L Isolation Logic to Field Power : Isolation Logic to System power : Non Isolation
Field Power	Supply voltage: 24Vdc nominal Supply voltage range: 11~28.8Vdc
Max. Current Field Power Contact	DC 10A Max.
Weight	155g
Module Size	45mm x 99mm x 70mm
Environment Condition	Refer: Environment Specification

3.1.3   LED Indicators

3.1.3.1   CAN-RUN LED

Table 5: CAN-RUN Status LED

Color	Status	Function
OFF	Not On -line	The Device is not on-line or No power is supplied to the unit.
Solid (Green)	Device Operational	The unit is operating in normal condition.
Single Flash (Green)	Device Stopped	The device is in STOPPED state
Continuous Flash (Green)	Device Pre-Operational	The device is in PRE-OPERATIONAL state

3.1.3.2   CAN-ERR Status LED

Table 6: CAN-ERR Status LED

Color	Status	Function
OFF	No Error	
Single Flash Red	Warning limit reached	At least one of the error counters of the CAN controller has reached or exceeded the warning level (too many error frames).
Double Flash Red	Error Control Event	The guarding monitor has asserted, guarding telegrams are no longer being received. The adapter is pre-operational state.
Triple Flash Red	Sync error	A sync error has occurred. - The adapter is in pre-operational (PDOs switch off).
Solid Red	BUS off	The CAN controlled BUS is off

3.1.3.3 IO Module Status LED (I/O)

Table 7: IO Module Status LED

Color	Status	Function
Off	Not powered	Device has no IO module or may not be powered
Flashing Green	Bus is On-line, Not Exchanging I/O data	Bus is on-line but does not exchanging I/O data, Initializing IO modules.
Solid Green	Bus is online, Exchanging IO data	IO modules initialized & connection established, exchanging I/O data
Solid Red	Bus connection fault during exchanging IO data	One or more IO modules in fault Status. - Changed IO module configuration. - Bus communication failure.
Flashing Red	IO Configuration failed	Failed to initialize IO module - Detected invalid IO module ID. - Overflowed Input / Output Size - Too many IO module - Initial protocol failure

3.1.3.4 Field Power Status LED

Table 8: Field Power Status LED

Color	Status	Function
Off	Field Power not connected	24V dc field power not connected
Solid Green	Field Power connected	24V dc field power connected

3.1.4   STXCAN001 Dimensions
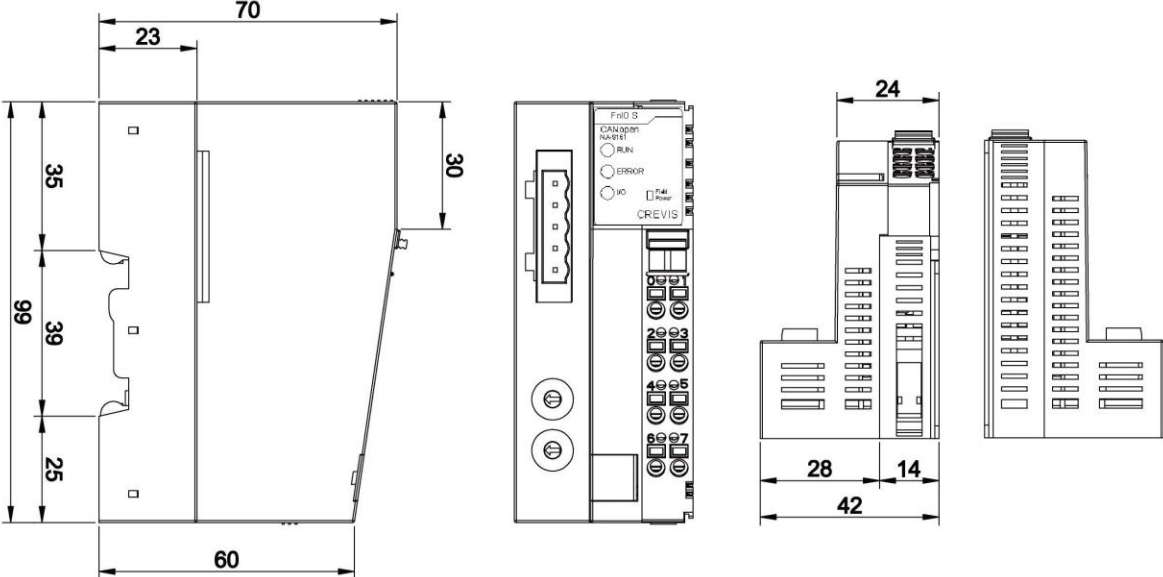
Figure 9: Dimension for STXCAN001

3.1.5   Total IO

The maximum number of IO module assemblies that can be connected is 32. So the maximum length is 426mm.

3.2 Configuration of Field Bus Node

3.2.1 CANopen Connector
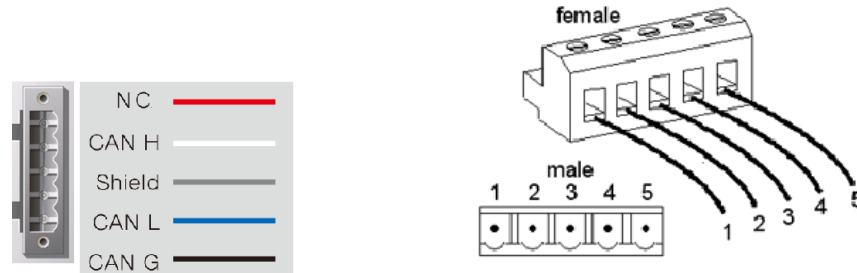
Figure 10: STXCAN001 Cable Description

Pin Number	Signal	Description
1	CAN_GND	Ground / 0V/ V-
2	CAN_L	CAN_L Bus line (dominant low)
3	Shield	Optional CAN shield
4	CAN_H	CAN_L Bus line (dominant High)
5	NC	No connection

Table 9: STXCAN001 connector signal description

3.2.2 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 [Kbps]
	Length-related Resistance [m/m]	Cross-section [mm^2]		
0 ... 40	70	0.25 to 0.34	124	1000kbps at 40m
40 … 300	< 60	0.34 to 0.6	150 to 300	> 500kbps at 100m
300 ... 600	< 40	0.5 to 0.6	150 to 300	> 100kbps at 500 m
600 ... 1000	< 26	0.75 to 0.8	150 to 300	> 50kbps at 1 km

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

Wire cross-Section [mm²]	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

3.2.3   Network Address and Baud Rate

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

3.2.3.1 Node ID

STXCAN001 Node address is determined by the position of rotary switches x1 & X10. For example: x10 = 2 & x1 = 7, address = 10 * 2 + 1*7 = 27.

Select address within the range of 01 to 99 (Station no. 01~99)..

3.2.3.2 Baud Rate

When both switches are set to '0' & powered up, the network adapter enters into baud rate setting mode. The RUN LED flashing green indicates the configuration mode.

- 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
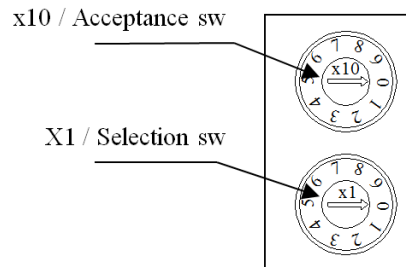
Figure 11: Rotate Switch for STXCAN001

To set the desired baud rate rotate the selection switch, clock wise to the position mentioned in below table. To confirm the baud rate rotate the acceptance switch, clock wise from 0 to 1.

The RUN LED Solid green indicates the baud rate confirmed. Power down the network adapter & set desired node ID as mentioned in above section.

Table 10: Baud rate setting for STXCAN001

Select S/W	0	1	2	3	4	5	6	7	8
Acceptance S/W	0→1	0→1	0→1	0→1	0→1	0→1	0→1	0→1	0→1
Baud rate	1Mbit/s	800Kbits/s	500Kbit/s	250Kbit/s	125Kbit/s	100Kbit/s	50Kbit/s	20Kbit/s	10Kbit/s

Note: **NODE ID addresses have to be unique throughout the entire interconnected networks.** Switches rotation direction is clockwise.

4.   *Communication*

4.1   *Device Model*

4.1.1   *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 STXCAN001 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.

4.2   *PDO (Process Data)*

4.2.1   *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 Adapter: 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.

4.2.2   *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 Adapters - for example, if 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. Analog I/O

4.2.3   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:
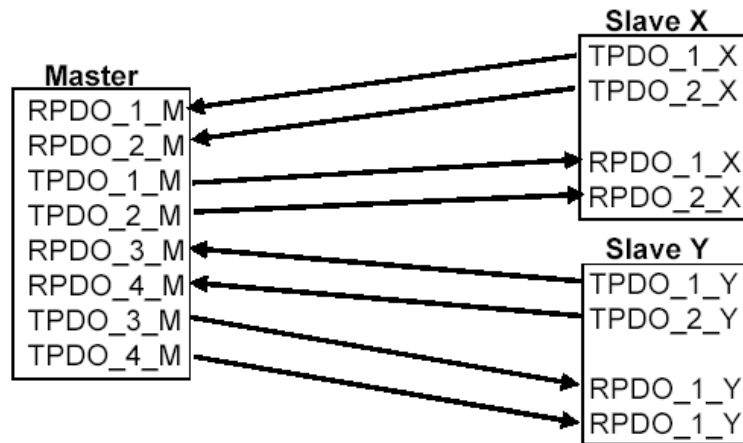
PDO Linking

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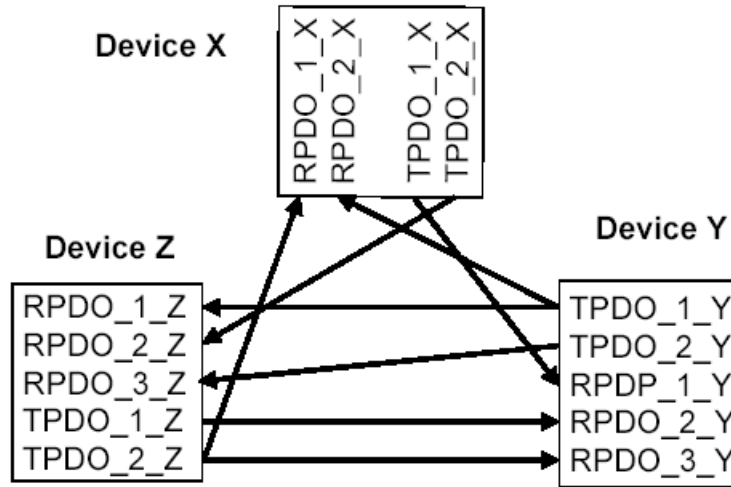
This procedure is known as PDO linking. It permits, for example, easy construction of electronic drives in which several slave axes simultaneously listen to the actual value in the master axis TxPDO.

4.2.4 PDO Communication Type

Event driven

In the event of change in input value, data is transmitted instantaneously over the network. 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 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 setting of 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 TxPDO or RxPDO 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).

Transmission types can be varied for various data, discrete input data rate can be faster and analogue data rate can be set to low. This provides an option to ensure time critical data is updated faster. 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.

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.

4.3   SDO (Service Data)

4.3.1   Introduction

Byte 0	Byte 1-3 : data addressing			Byte 4-7 : 1-4 byte of data			
Command Specified	Index Low Byte	Index High Byte	Sub index	Data 0	Data 1	Data 2	Data 3

-Upload

-Download

-Number of data byte

-Request

-Response

Object directory

Index	Sub-index	Data
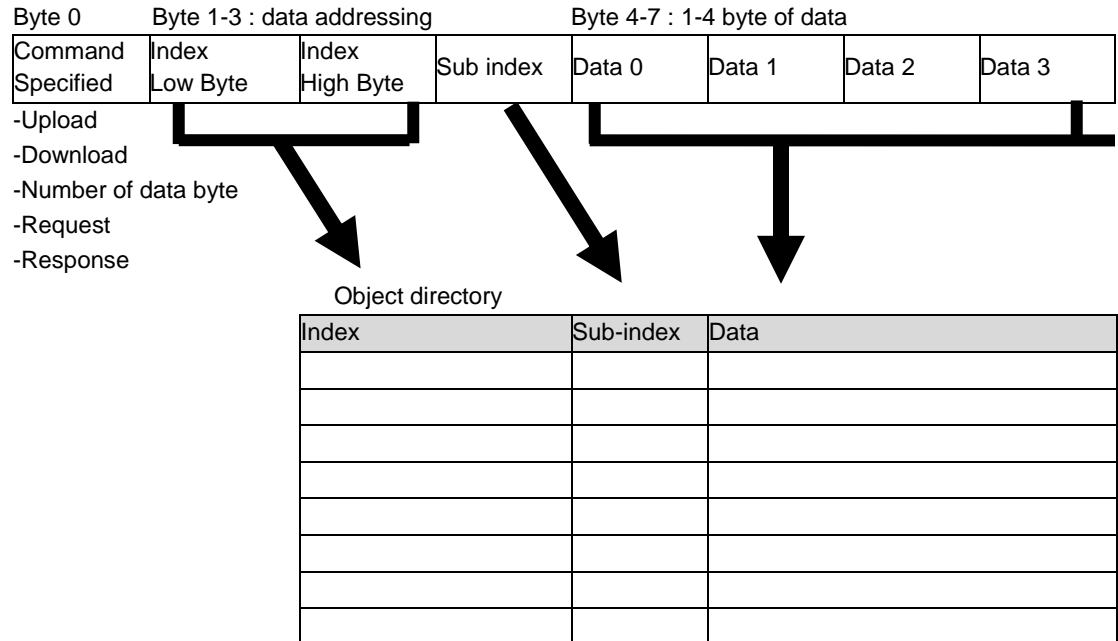
The CANopen Network Adapters are servers for the SDO, which means that at the request of a client they make data available (upload), or they receive data from the client (download). This involves a handshake between the client and the server. When the size of the parameter to be transferred is not more 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.

4.4   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 Adapter, because modules were connected or disconnected.**

4.5 NMT (Network Management)

4.5.1 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.
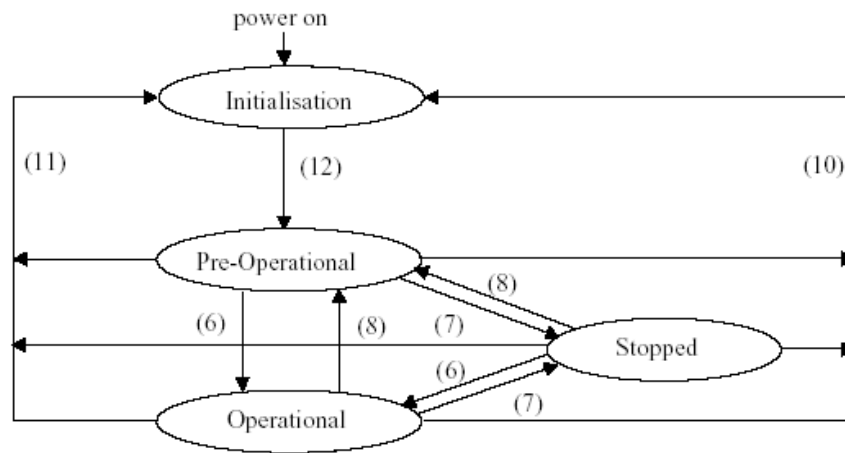
In **Initialization** state, the CANopen data structures of a node are initialized by the application. The CiA DS-301 standard defines various mandatory OD entries for this task as well as specific communication objects required for that. In the minimum device configuration, the identifier for these 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.

4.5.2   Boot up Message

After the initialization phase and the self-test, the Network Adapter 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.

4.5.3   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.

4.5.4   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.

4.5.5   IO Process Image Map

An IO module may have 3 types of data as I/O data, configuration parameter and memory register.

The data exchange between network adapter and IO modules is done via an I/O process image data by Bus protocol. The following figure shows the data flow of process image between network adapter and IO modules.

Figure 12: Data exchange between network adapter and IO modules
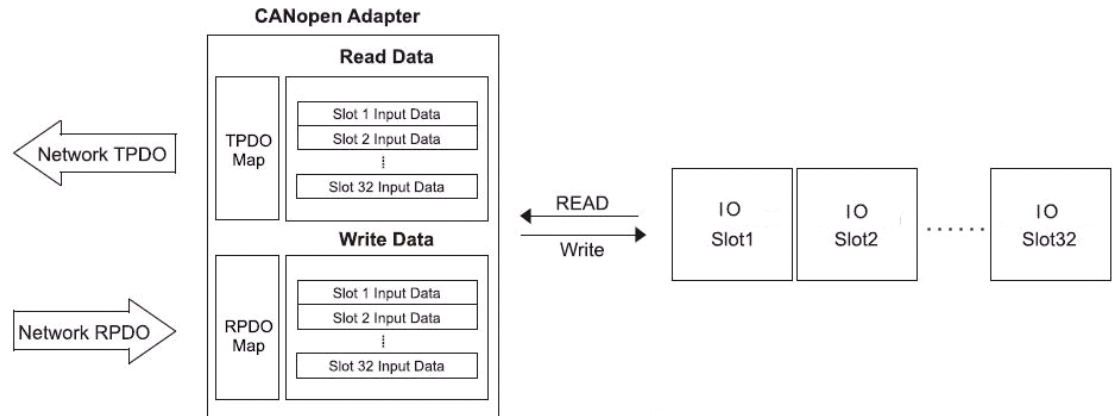
4.5.6   Input Process Image Map

Input image data depends on slot position and IO slot data type. Input process image data is only ordered by IO 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 IO slot position and slot data type.

Input process image mode can be set by Object Index 0x4500

Figure 13: Slot Configuration

Table 11: Slot number and Module Description

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

4.5.7   Output Process Image Map

Output image data depends on slot position and IO slot data type. Output process image data is only ordered by IO 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 IO slot position and slot data type.

Output process image mode can be set by Object Index 0x4500

Figure 14: Slot Configuration

Table 12: Slot number and Module Description

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

Byte								Index	Sub-Index
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

4.5.8   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 via 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.

4.5.9   Object Directory

All the CANopen objects relevant for the Network Adapter 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.

4.5.10   Communication Profile Area

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

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	Reserved				
0x100A	0	Reserved				
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	Unsigned8	RO	1	Number of store options
	1	Save all parameters	Unsigned32	RW	0	Store all parameters
0x1011	0	Restore default parameters	Unsigned8	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	Unsgined32	RW	0	Heartbeat time value
0x1017	0	Producer Heartbeat time	Unsgined16	RW		
0x1018	0	Identity object	Unsgined8	RO	4	
	1	Manufacturer ID	Unsigned32	RO	0x01000180	
	2	Product code	Unsgined32	RO		
	3	Revision number	Unsgined32	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+nodelD	COB-ID RxPDO1
	2	Transfer type	Unsgined8	RW	254	Transmission type of the PDO
	3	Inhibit time	Unsigned16	RW	0x8813	Inhibit time of the PDO
	5	Event timer	Unsigned16	RW	0	Event time of the PDO
…	…	…				
	0	Receive PDO parameter	Unsgined8	RW	5	number of following parameters
	1	COB-ID used by PDO	Unsigned32	RW	0x80000000	COB-ID RxPDO8
0x1407	2	Transfer type	Unsgined8	RW	254	Transmission type of the PDO
	3	Inhibit time	Unsigned16	RW	50000	Inhibit time of the PDO
	5	Event timer	Unsigned16	RW	0	Event time of the PDO
	0	Receive PDO mapping	Unsgined8	RW	0	Number of mapped objects.
0x1600	1	1. Object	Unsigned32	RW	0x62000108	
	…	…				
	8	8. Object	Unsigned32	RW	0x62000808	
…	…	…				
	0	receive PDO mapping	Unsgined8	RW	0	Number of mapped objects.
0x1607	1	1. Object	Unsigned32		0	
	…	…				
	8	8. Object	Unsigned32	RW	0	
	0	Transfer PDO parameter	Unsgined8	RW	5	number of following parameters
	1	COB-ID used by PDO	Unsigned32	RW	0x180+nodelD	COB-ID TxPDO1
0x1800	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
…	…	…				
	0	Transfer PDO parameter	Unsgined8	RW	5	number of following parameters
	1	COB-ID used by PDO	Unsigned32	RW	0x80000000	COB-ID TxPDO1
0x1807	2	Transfer type	Unsgined8	RW	254	Transmission type of the PDO
	3	Inhibit time	Unsigned16	RW	5000	Inhibit time of the PDO
	5	Event timer	Unsigned16	RW	0	Event time of the PDO

		Transfer PDO mapping	Unsgined8	RW	0	Number of mapped objects.
0x1A00	1	1. Object	Unsigned32	RW	0x60000108	
	…	…				
	8	8. Object	Unsigned32	RW	0x60000808	
…	…	…				
	0	Transfer PDO mapping	Unsgined8	RW	0	Number of mapped objects.
0x1A07	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 Adapter 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 Adapter.

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

The object indicates the current hardware version of the Network Adapter

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

The object indicates the current software version of the Network Adapter

■ **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.

■   **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		Sub-Index		Size	
(Unsigned16)		(Unsigned8)		(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.

Warning

  An object entry can only be mapped in maximum. of three 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)

4.5.11 Manufacturer Specific Profile Area

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

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	IO slot#1 memory block read
	…	…				
	32	Read memory slot #32	Visible data	RO	None	IO slot#32 memory block read
0x2020	0	Input information	Unsigned8	RO	1	
	1	Digital input bit size	Unsigned16	RO		IO 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	IO slot#1 memory block write
	…	…				
	32	Write memory slot #32	Visible data	WO	None	IO slot#32 memory block write
0x2220	0	Output information	Unsigned8	RO	1	
	1	Digital output bit size	Unsigned16	RO		IO 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	Bus communication register	Unsgined8	RO	0x03	Number of communication register block
	1	Bus status	Unsigend32	RO	0x00000 000	Bus Error monitoring ,field power check
	2	Bus Data mode	Unsigend8	RW	0x01	0: non compress mode 1: compress mode
	3	IO active flag	Unsgined32	RW	0x00000 000	IO module active flag(bit active)
0x4501	0	Range of Read Memory map	Unsigend8	RO	None	Number of IO 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	IO slot configuration	Unsigend8	RO	None	Number of IO 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	IO module product code	Unsigend8	RO	None	Number of IO module product code
	1	Slot product code #01	Unsigned32	RO	None	IO module product code slot #1
	…					
	32	Slot product code #32	Unsigned32	RO	None	IO module product code slot #32
0x4504	0	IO module catalog code	Unsigend8	RO	None	Number of IO module catalog code
	1	Slot catalog code #01	Unsigned32	RO	None	IO module catalog code slot #1
	…					
	32	Slot catalog code #32	Unsigned32	RO	None	IO 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		IO 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		IO 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 Adapter. 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 Adapter. Sub-index 2 the next etc.

■ **Object 0x4500, Bus Communication Register**

Index	Sub	Name	Type	Attribute	Default	Meaning
0x4500	0	Bus communication register	Unsgined8	RO	0x03	Number of communication register block
	1	Bus status	Unsigend32	RO	0x00000000	Bus Error monitoring ,field power check
	2	Bus Data mode	Unsigend8	RW	0x01	0: non compress mode 1: compress mode
	3	IO active flag	Unsgined32	RW	0x00000000	IO module active flag (bit active)

Bus communication register Format: The Index 0x4500 can be accessed via SDO.

Bus Error monitor data format: This object is Bus state.

Index	Sub-Index	Decimal Byte	Data Type	Description
0x4500	0x01	Byte 00	unsgined8	Bus 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

Bus Data Mode: The Process Image 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)

IO 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)

■   **Object 0x4502, I/O Modules Parameter Configuration**

Index	Sub	Name	Type	Attribute	Default	Meaning
0x4502	0	IO slot configuration	Unsigend8	RO	None	Number of IO 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	IO module product code	Unsigend8	RO	None	Number of IO module product code
	1	Slot product code #01	Unsigned32	RO	None	IO module product code slot #1
	…					
	32	Slot product code #32	Unsigned32	RO	None	IO module product code slot #32

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

Index	Sub	Name	Type	Attribute	Default	Meaning
0x4504	0	IO module catalog code	Unsigend8	RO	None	Number of IO module catalog code
	1	Slot catalog code #01	Unsigned32	RO	None	IO module catalog code slot #1
	…					
	32	Slot catalog code #32	Unsigned32	RO	None	IO module catalog code slot #32

4.5.12 Standard Device Profile Area – DS401

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

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 Adapter. 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 Adapter. 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. adapter 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 Adapter. 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 Adapter. 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. adapter 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

A.   *Diagnostics*

A.1   *How to Diagnose when Device Cannot Communicate with the Network*

- **Verify that cable connections are correct.**
- **If terminator resistor is not installed, install terminator resistor. Check location of terminator resistor.**
- **Ensure there are no duplicate node addresses.**
- **Check configuration of master for port settings, time out, data size etc.**
- **Power cycle the network adapter to ensure applicability of correct baud rate, parity, byte formats, IO configuration parameters, Input and Output process image modes.**
- **Check system power & field power connections. Ensure separate power sources are used for system power and Field power. Ensure Network adapter or system power modules are not over loaded.**
- **Ensure ground cable connections.**
- **Ensure environment factors are within the limit.**

B. Product List

Table 13: Product List

ST-Number	Description	ID(hex)	Production Status
	Digital Input Modules		
ST-1214	4 Points, Sink(Positive), 12V/24Vdc	41 00 03	Active
ST-1218	8 Points, Sink(Positive), 12V/24Vdc	41 00 07	Active
ST-121F	16 Points, Sink(Positive), 12V/24Vdc	41 01 13	Active
ST-1224	4 Points, Source(Negative), 12V/24Vdc	41 00 04	Active
ST-1228	8 Points, Source(Negative), 12V/24Vdc	41 00 08	Active
ST-122F	16 Points, Source(Negative), 12V/24Vdc	41 01 14	Active
ST-1314	4 Points, Sink(Positive), 48Vdc	41 00 05	Active
ST-1324	4 Points, Source(Negative), 48Vdc	41 00 06	Active
ST-1804	4 Points, 110Vac,	41 00 09	Active
ST-1904	4 Points, 220Vac,	41 00 0A	Active
	Digital Output Modules		
ST-221F	16 Points Sink(Negative Logic), 24Vdc/0.5A,	81 01 15	Active
ST-222F	16 Points Source(Positive Logic), 24Vdc/0.5A,	81 01 16	Active
ST-2314	4 Points Sink(Negative Logic), 24Vdc/0.5A,	81 00 0E	Active
ST-2318	8 Points Sink(Negative Logic), 24Vdc/0.5A,	81 00 11	Active
ST-2324	4 Points Source(Positive Logic), 24Vdc/0.5A,	81 00 10	Active
ST-2328	8 Points Source(Positive Logic), 24Vdc/0.5A,	81 00 12	Active
ST-2414	4 Points Sink(Negative Logic), 24Vdc/0.5A, Diagnostics	81 00 08	Active
ST-2424	4 Points Source(Positive Logic),24Vdc/0.5A, Diagnostics	C1 00 00 38	Active
ST-2514	4 Points Sink(Negative Logic), 24Vdc/2A, Diagnostics	C1 00 00 35	Active
ST-2524	4 Points Source(Positive Logic), 24Vdc/2A, Diagnostics	C1 00 00 36	Active
ST-2742	2 Points, 230Vac/2A, 24Vdc/2A, Relay	81 00 0B	Active
ST-2852	2 Points, 12~125Vac/0.5A, Triac	81 00 0C	Active

ST-Number	Description	ID(hex)	Production Status
Analog Input Modules			
ST-3114	4 Channels, Current, 0~20mA, 12bit	41 43 1C	Active
ST-3134	4 Channels, Current, 0~20mA, 14bit	41 43 1E	Active
ST-3214	4 Channels, Current, 4~20mA, 12bit	41 43 1D	Active
ST-3234	4 Channels, Current, 4~20mA, 14bit	41 43 1F	Active
ST-3424	4 Channels, Voltage, 0~10Vdc, 12bit	41 43 20	Active
ST-3444	4 Channels, Voltage, 0~10Vdc, 14bit	41 43 22	Active
ST-3524	4 Channels, Voltage, -10Vdc~10Vdc, 12bit	41 43 21	Active
ST-3544	4 Channels, Voltage, -10Vdc~10Vdc, 14bit	41 43 23	Active
ST-3624	4 Channels, Voltage, 0~5Vdc, 12bit	41 43 24	Active
ST-3644	4 Channels, Voltage, 0~5Vdc, 14bit	41 43 25	Active
ST-3702	2 Channels, RTD, Status	41 41 28	Active
ST-3802	2 Channels, TC	41 41 2A	Active
Analog Output Modules			
ST-4112	2 Channels, Current, 0~20mA, 12bit	81 41 2C	Active
ST-4212	2 Channels, Current, 4~20mA, 12bit	81 41 2D	Active
ST-4422	2 Channels, Voltage, 0~10Vdc, 12bit	81 41 2E	Active
ST-4522	2 Channels, Voltage, -10~10Vdc, 12bit	81 41 2F	Active
ST-4622	2 Channels, Voltage, 0~5Vdc, 12bit	81 41 30	Active

ST-Number	Description	ID(hex)	Production Status
Special Modules			
ST-5101	1 Channel, High Speed Counter, 5V Input	C1 01 05 34	Active
ST-5111	1 Channel, High Speed Counter, 24V Input	C1 01 05 39	Active

C. *Product Certifications and Installation Guidelines for Conformance*

This appendix describes the compliance markings and standards to which the RSTi products have been certified.

C.1 *Safety Notes*

Warning

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, for example, RSTi Bus Pin.

C.2 *Certifications*

$_c$UL$_{US}$ Listed Industrial Control Equipment, certified for U.S. and Canada

See UL File E105285

CE Mark

EN 61000-6-2:2005; Industrial Immunity

EN 61000-6-4:2007; Industrial Emissions

C.3   Government Regulations

U.S., Canadian, Australian, and European regulations are intended to prevent equipment from interfering with approved transmissions or with the operation of other equipment through the AC power source.

The PACSystems RSTi family of products has been tested and found to meet or exceed the requirements of U.S. (47 CFR 15), Canadian (ICES-003), Australian (AS/NZS 3548), and European (EN 61000-6-4:2007) regulations for Class A digital devices when installed in accordance with the guidelines noted in this manual. These various regulations share commonality in content and test levels with that of CISPR 22 and based on this commonality testing to the each individual standard was deemed inappropriate.

The FCC requires the following note to be published according to FCC guidelines:

Note: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case user will be required to correct the interference at his own expense.

Industry Canada requires the following note to be published:

Note: This Class A digital apparatus complies with Canadian ICES-003.

C.4   Environmental Specifications

C.4.1   CANopen STXCAN001

Table 14: Environmental specification for STXCAN001

Items	*Specification*
Operating Temperature	-20℃ to 55℃ for UL applications ; -20℃ to 60℃ for non-UL applications
Non-Operating Temperature	Storage -40℃ to 85℃
Relative Humidity	5% ~ 90% Non-condensing
Operating Altitude	2000m
Mounting	DIN Rail

C.5   Abbreviations

Table 15: Abbreviations

Items	Description
ASCII	American Standard Code for Information Interchange
AI	Analog Input
AO	Analog Output
NA	Network adapter
CAL	CAN Application Layer
CAN	Controller Area Network
COB ID	Communication Object Identifier
DI	Digital Input
DO	Digital Output
EMCY	Emergency Object
I/O	Input / Output
ID	Identifier, Identification
Idx	Index
M	Master
NMT	Network Management
PDO	Process Data Object
RO	Read Only
RTR	Remote Transmit Request
RxPDO	Receive PDO
RW	Read/Write
SDO	Service Data Object
S-Idx	Sub-Index
TxPDO	Transmit PDO

**GE Intelligent Platforms
Information Centers**

Headquarters:
1-800-433-2682 or 1-434-978-5100

Global regional phone numbers
are available on our web site
www.ge-ip.com

Additional Resources

For more information, please visit the GE
Intelligent Platforms web site:

www.ge-ip.com

GFK-2807