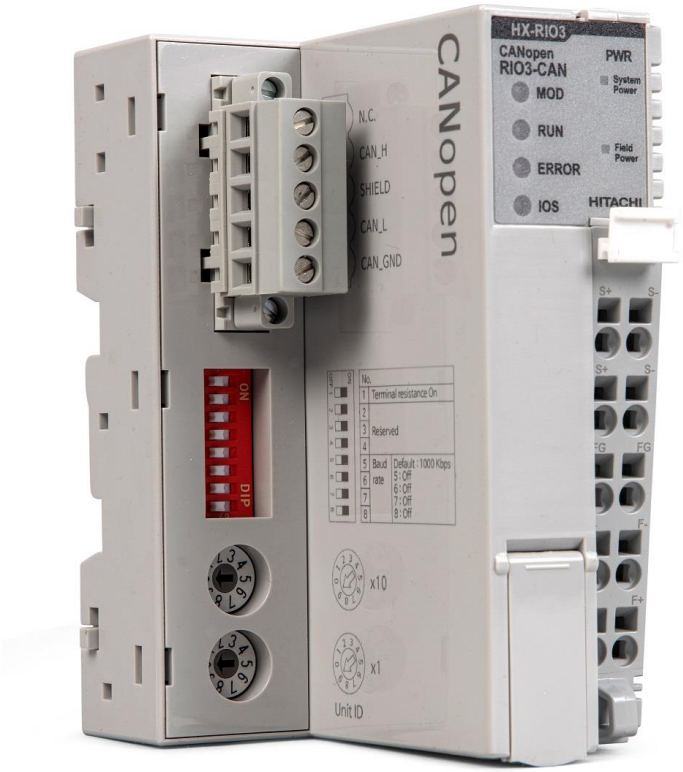
CANopen Network Adapter

RIO3-CAN
User Manual



HITACHI
Inspire the Next

DOCUMENT CHANGE SUMMARY				
REV.	**PAGES**	**REMARKS**	**DATE**	**Editor**
1.02	All		March 21	Faber

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 HITACHI 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, HITACHI 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. 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, G-BUS Pin.

1.1.3. Certification

2. Environment Specification

Environment Specification	
Operating Temperature	-40℃~70℃
UL Temperature	-20℃~60℃
Storage Temperature	-40℃~85℃
Relative Humidity	5% ~ 90% non-condensing
Operating Altitude	2000m
Mounting	DIN rail
General Specification	
Shock Operating	IEC 60068-2-6
Vibration resistance	Based on IEC 60068-2-6 DNVGL-CG-0039 : Vibration Class B, 4g
EMC resistance burst/ESD	EN 61000-6-2 : 2005
Installation Pos. / Protect. Class	Vertical and horizontal installation is available.
Product Certifications	CE, UL, EAC

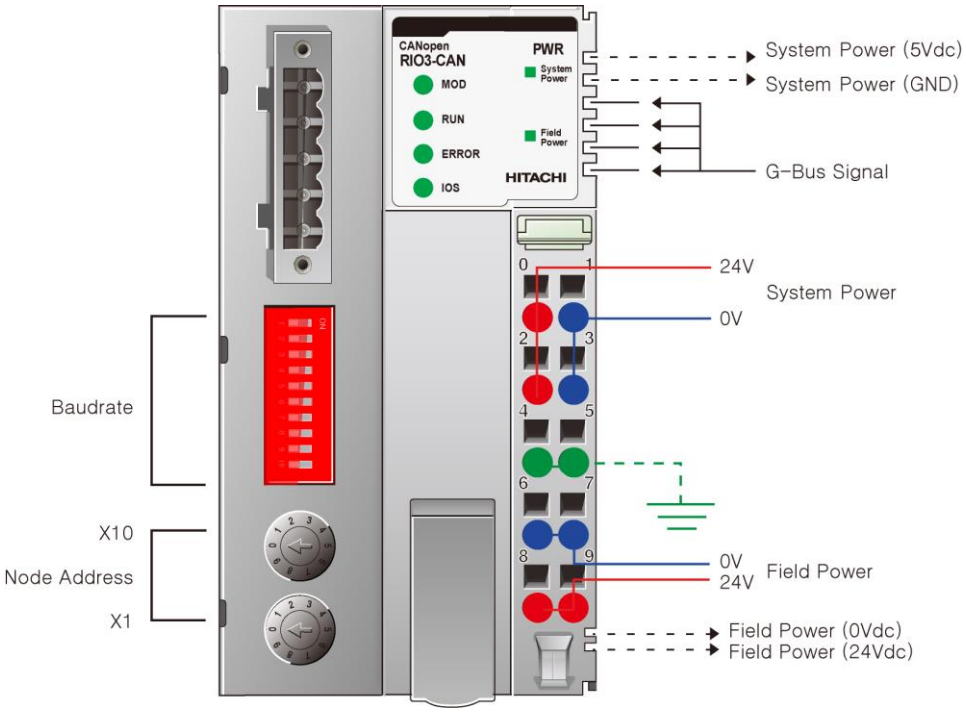
3. General Description

3.1. RIO3-CAN Specification

Interface Specifications, RIO3-CAN	
Adapter Type	Slave node (CANopen, CiA 301)
Max. PDO Number	32 TPDO / 32 RPDO
Max. SDO Number	1 Standard SDOs
PDO Mapping Method	Auto Mapping : CiA 301 (default), Sequential Manual Mapping (Reference to Index : 0x20F0, Sub Index : 6)
Max. Expansion Module	63 slots
Max. Input size	252 bytes
Max. Output size	252 bytes
Nodes	1~99
Baudrate	10, 20, 50, 100, 125, 250, 500, 800, 1000 Kbps(default 1000Kbps)
Interface Connector	5pin Open Connector
Other Serial Port	RS232 for MODBUS/RTU(Touch Panel, I/O Guide ...)
Serial Configuration (RS232)	Node : 1 (Fixed) Baud Rate : 115200 (Fixed) Data bit : 8 (Fixed) Parity bit : No parity (Fixed) Stop bit : 1 (Fixed)
Indicator	6 LEDs 1 Green/Red, Module Status (MOD) 1 Green, Network Status(RUN) 1 Red, Error Status (ERROR) 1 Green/Red, Expansion I/O Module Status (IOS) 1 Green, System Power Status 1 Green, Field Power Status
Module Location	Starter module left side of RIO3 Series system
Field Power Detection	About 14Vdc
General Specification	
UL System Power	Supply voltage : 24Vdc nominal, Class2
System Power	Supply voltage : 24Vdc nominal Supply voltage range : 15~30Vdc Protection : Output current limit (Min. 1.5A)

	Reverse polarity protection
Power dissipation	70mA typical @ 24Vdc
Current for I/O module	1.5A @ 5Vdc
Isolation	System power to internal logic : Non-isolation System power I/O driver : Isolation
UL Field Power	Supply voltage : 24Vdc nominal, Class2
Field Power	Supply voltage : 24Vdc typical(Max. 30Vdc) * Field Power Range is different depending on IO Module series. Refer to IO Module`s Specification.
Max. Current Field Power Contact	DC 10A Max
Wiring	I/O Cable Max. 2.0 ㎟ (AWG 14)
Torque	0.8Nm(7 lb-in)
Weight	162g
Module Size	54mm x 99mm x 70mm
Environment Condition	**Refer to '2. Environment Specification'**

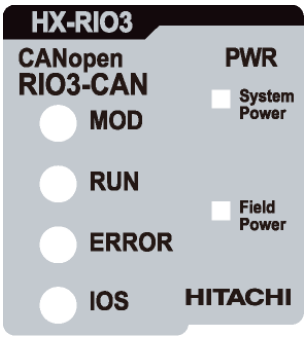
3.2. RIO3-CAN Wiring Diagram



Pin No.	Signal Description	Signal Description	Pin No.
0	System Power, 24V	System Power, Ground	1
2	System Power, 24V	System Power, Ground	3
4	F.G	F.G	5
6	Field Power, Ground	Field Power, Ground	7
8	Field Power, 24V	Field Power, 24V	9

3.3. RIO3-CAN LED Indicator

3.3.1. LED Indicator



LED No.	LED Function / Description	LED Color
MOD	Module Status	Green/Red
RUN	CANopen Status	Green
ERROR	CANopen Error Status	Red
IOS	Extension Module Status	Green/Red
System Power	System Power Enable	Green
Field Power	Field Power Enable	Green

3.3.2. MOD(Module Status LED)

Status	LED	To indicate
Not Powered	OFF	Not power is supplied to the unit.
Normal Operational	Green	The unit is operating in normal condition.
Unrecoverable Fault	Red	The device has an unrecoverable fault. - Memory error or CPU watchdog error.

3.3.3. RUN(Module Status LED)

Status	LED	To indicate
Not Powered	OFF	Module is either not getting power, or it is NOT_ACTIVE status. Module is initializing.
Stopped	Single flash Green	Module is stopped
Pre-Operational	Blinking Green	Module is in PRE-OPERATIONAL
Operational	Green	Module is in OPENRATIONAL

3.3.4. ERROR (CANopen Error LED)

Status	LED	To indicate
Not Powered	OFF	Module is not getting powered or No error.
Warning limit reached	Single flash	At least one of the error counters of the CAN controller has reached or exceeded the warning limit.
Error control Event	Double flash	A guard event (NMT-Slave or NMT-Master) or a Heartbeat event has occurred.
Sync. Error	Triple flash	The SYNC message has not been received within then configured communication cycle period time out (see index 0x1006)
Bus Off	ON	The CAN controller is bus off.

3.3.5. IOS LED (Extension Module Status LED)

Status	LED	To indicate
Not Powered	OFF	Device may not be powered.
No Expansion Module	Flashing Red	Adapter has no expansion module
Internal Bus Connection, Run Exchanging I/O	Green	Exchanging I/O data.
Expansion Configuration Failed	Red	One or more expansion module occurred in fault state. - Detected invalid expansion module ID. - Overflowed Input/Output Size - Too many expansion module - Initialization failure - Communication failure. - Changed expansion module configuration. - Mismatch vendor code between adapter and expansion module.

3.3.6. Field-, System Power LED(Field-, System Power Status LED)

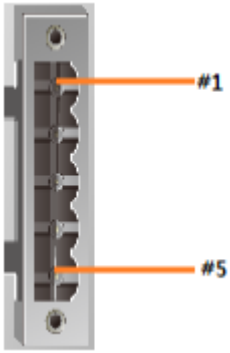
Status	LED	To indicate
Not supplied field, system power	OFF	Not supplied 24Vdc field power, 5Vdc system power
Supplied field, system power	Green	Supplied 24Vdc field power, 5Vdc system power

3.3.7. Indicator states and flash rates

LED ON	Constantly ON
LED OFF	Constantly OFF.
LED Blinking	Equal ON and OFF times with a frequency of approximately 10 Hz: ON for approximately 50ms and OFF for approximately 50ms.
LED Single flash	One short flash (approximately 200ms) followed by a long OFF phase (approximately 1000ms)
LED Double flash	A sequence of two short flashes (approximately 200ms), separated by an OFF phase (approximately 200ms). The sequence is finished by a long OFF phase (approximately 1000ms)
LED Triple flash	A sequence of three short flashes (approximately 200ms), separated by an OFF phase (approximately 200ms). The sequence is finished by a long OFF phase (approximately 1000ms)

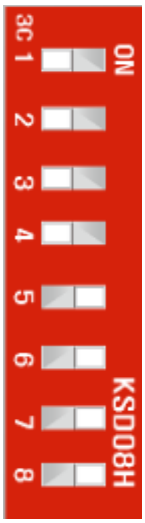
3.4. RIO3-CAN Electrical Interface

3.4.1. 5pin Open Connector



No.	Signal Name	Description
1	-	
2	CAN H	CAN High
3	Shield	F.G
4	CAN L	CAN Low
5	CAN G	CAN GND

3.4.2. Dip Switch

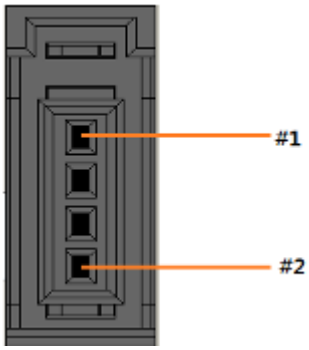


DIP Pole#	Description	
1	Terminal Resister	On : Terminal Resister On
2	Reserved	
3		
4		
5	Baudrate #4	Default : 0 (1000kbps) 5 6 7 8 (ex.) 0 0 0 1 : 10Kbps 0 0 1 0 : 20Kbps 0 0 1 1 : 50Kbps 0 1 0 0 : 100Kbps 0 1 0 1 : 125Kbps 0 1 1 0 : 250Kbps 0 1 1 1 : 500Kbps 1 0 0 0 : 800Kbps 1 0 0 1 : 1000Kbps
6	Baudrate #3	
7	Baudrate #2	
8	Baudrate #1	

3.4.3. Address setup

Each Adapter could have an unique address (from 1 to 99) so that it can be addressed independently from other nodes. The address 0 is reserved.

3.4.4. RS232 Port for MODBUS/RTU, Touch Panel or I/O Guide

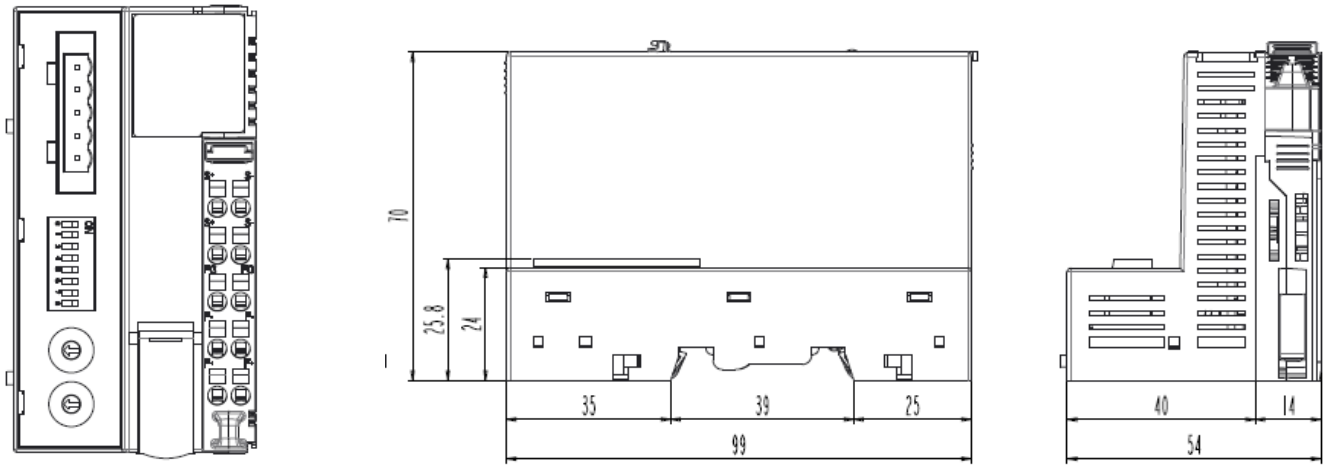


RS232 (37204-64A3-004PL/3M		
Pin#	**Signal Name**	**Description**
1	Reserved	----
2	TXD	RS232 TXD
3	RXD	RS232 RXD
4	GND	RS232 GND

4. Dimension

4.1. RIO3-CAN

(mm)



5. Communication

5.1. Device Model

5.1.1. 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 RIO3-CAN supports the following objects:

- 32 transmit PDOs
- 32 receive PDOs
- 1 standard SDO (server)
- 1 emergency object
- Node guarding
- NMT objects

Every CANopen device possesses a CANopen object directory in which

The parameters for all the CANopen objects are entered.

5.2. PDO (Process Data)

5.2.1. Introduction

The real-time data transfer is performed by means of "Process Data Objects (PDO)". The transfer of PDO is performed with no protocol overhead.

The PDO correspond to objects in the object dictionary and provide the interface to the application objects. Data type and mapping of application objects into a PDO is determined by a corresponding default PDO mapping structure within the object dictionary.

There are two kinds of use for PDO. The first is data transmission and the second data reception. It is distinguished in Transmit-PDO (TPDO) and Receive-PDO (RPDO). CANopen devices supporting TPDO are PDO producer and CANopen devices supporting RPDO are called PDO consumer.

5.2.2. 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 analog inputs or outputs are located in the second PDO. These PDOs are accordingly occupied by the Network Adapters - 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. Analog I/O

5.2.3. 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

When it comes to the communication partners involved, we have a similar arrangement as with the SDOs. The default is that the master is the only node that receives Transmit Process Data Objects (TPDO). And only the master may send Receive Process Data Objects (RPDO) to the slaves.

Once these new linking settings are made and the network goes into the operational mode, the master would not need to get involved into the process data communication and could focus on other things like network management.

5.2.4. PDO Communication Type

Event driven

Message transmission is triggered by the occurrence of an object specific event. For synchronous PDOs this is the expiration of the specified transmission period, synchronised by the reception of the SYNC object.

For acyclically transmitted synchronous PDOs and asynchronous PDOs the triggering of a message transmission is a device-specific event specified in the device profile.

Polling

The PDOs can be polled by remote transmission request. The HITACHI CANopen bus Adapter supports the interrogation of PDOs by means of remote frames.

Synchronized

In order to synchronize CANopen devices a synchronization object (SYNC object) is transmitted periodically by a synchronization application. The SYNC object is represented by a pre-defined communication object.

PDO transmission type

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

Synchronous (transmission types 0-240 and 252) means that the transmission of the PDO shall be related to the SYNC object. Preferably the devices use the SYNC as a trigger to output or actuate based on the previous synchronous Receive PDO respectively to update the data transmitted at the following synchronous Transmit PDO.

Asynchronous

Event-driven means that the PDO may be transmitted at any time based on the occurrence of a CANopen device internal event. The definition of the event does not fall into the scope of this specification and may be specified in device profiles and application profiles.

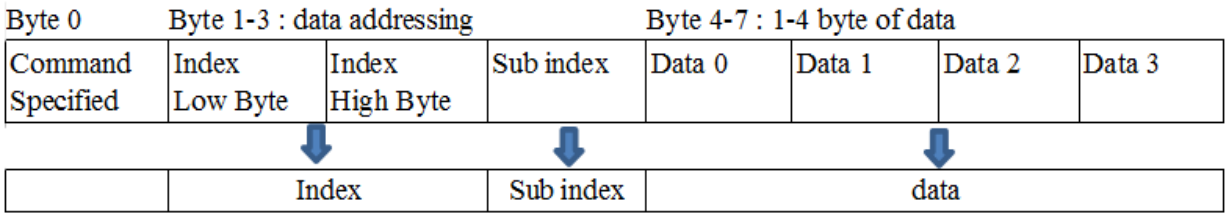
Inhibit Time

The inhibit time represents the minimum time that must elapse between transmission of two TPDOs. This enables a reduction of the bus load and an increase in data bandwidth. The value is defined as multiple of 100ms. It is not allowed to change the value while the PDO exists.

5.3. SDO (Service Data)

5.3.1. Introduction

With Service Data Objects (SDOs) the access to entries of a device Object Dictionary is provided. As these entries may contain data of arbitrary size and data type SDOs can be used to transfer multiple data sets (each containing an arbitrary large block of data) from a client to a server and vice versa. The multiplexer consists of a 16-bit index and an 8-bit sub-index that address the corresponding entries in the object directory.



Basically an SDO is transferred as a sequence of segments. Prior to transferring the segments there is an initialization phase where client and server prepare themselves for transferring the segments. For SDOs, it is also possible to transfer a data set of up to four bytes during the initialization phase. This mechanism is called SDO expedited transfer.
Optionally an SDO may be transferred as a sequence of blocks where each block may consist of a sequence of up to 127 segments containing a sequence number and the data. Prior to transferring the blocks there shall be an initialization phase where client and server may prepare themselves for transferring the blocks and negotiating the number of segments in one block. After transferring the blocks there shall be a finalization phase where client and server may verify the correctness of the previous data transfer by comparing checksums derived from the data set. The transfer type mentioned above is called SDO block transfer, which is faster than the segmented transfer for a large set of data.

5.4. 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:

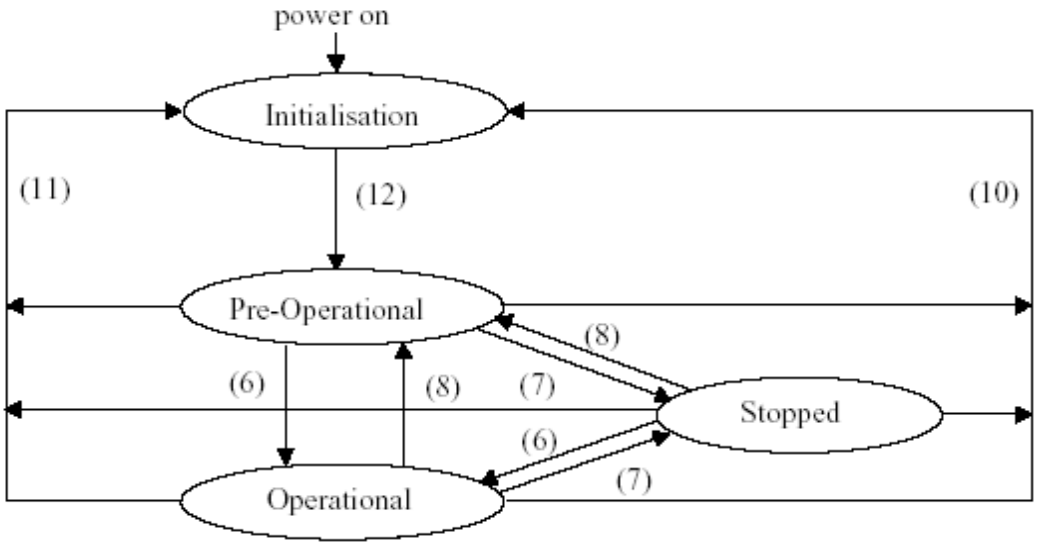
• No settings have yet been saved (Index 0x1010).

• The saved setting was discarded by the Network Adapter, because modules were connected or disconnected.

5.5. NMT (Network Management)

5.5.1. 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.

5.5.2. 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.

5.5.3. 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.

5.5.4. 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.

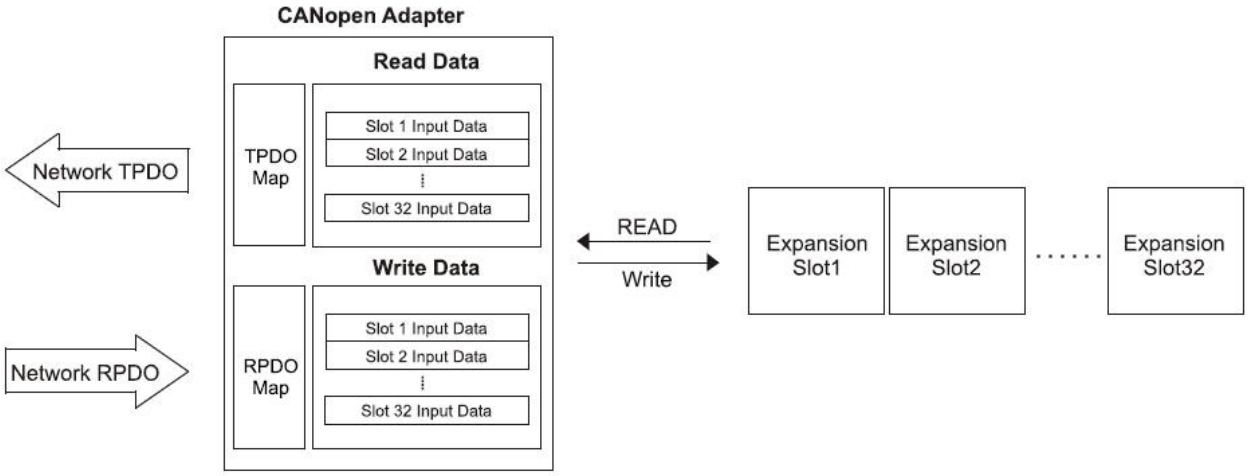
Response from the NMT Slave to a Life Guard Remote Frame

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.

5.6. I/O Process Image Map

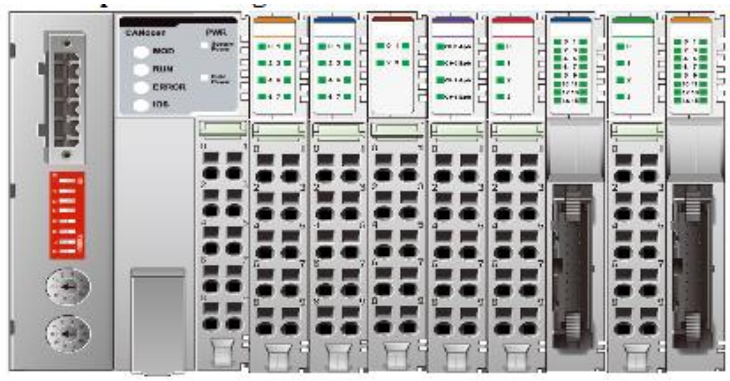
An expansion module may have 3 types of data as I/O data, configuration parameter and memory register. The data exchange between network adapter and expansion modules is done via an I/O process image data by RIO3 Bus protocol. The following figure shows the data flow of process image between network adapter and expansion modules.



5.6.1. Example of Input and Output 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

Eexample slot configuration



Slot Address	Module Description
#0	CANopen Adapter
#1	8-discrete output
#2	8-discrete input
#3	4-analog output
#4	2ch, high speed counte
#5	4-relay output
#6	16-discrete input
#7	4-analog input
#8	16-discrete output

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 Ch2 low byte (Slot#3)								0x6411	0x03
5	Analog Output Ch2 high byte (Slot#3)								0x6411	0x03
6	Analog Output Ch3 low byte (Slot#3)								0x6411	0x04
7	Analog Output Ch3 high byte (Slot#3)								0x6411	0x04
8	Discrete Output 8 pts. (Slot#1)								0x6200	0x01
9	Reserved				Discrete Output 4 pts. (Slot#5)				0x6200	0x02
10	Discrete Output low 8 pts. (Slot#8)								0x6200	0x03
11	Discrete Output high 8 pts. (Slot#8)								0x6200	0x04
12	Discrete Input 8 pts. (Slot#2)								0x6000	0x01
13	Discrete Input low 8 pts. (Slot#6)								0x6000	0x02
14	Discrete Input high 8 pts. (Slot#6)								0x6000	0x03
15	Discrete Input low 8 pts. (Slot#9)								0x6000	0x08
16	Discrete Input high 8 pts. (Slot#9)								0x6000	0x09
17	Analog Output Ch0 low byte (Slot#7)								0x6400	0x01

18	Analog Output Ch0 high byte (Slot#7)	0x6400	0x01
19	Analog Output Ch1 low byte (Slot#7)	0x6400	0x02
20	Analog Output Ch1 high byte (Slot#7)	0x6400	0x02
21	Analog Output Ch2 low byte (Slot#7)	0x6400	0x03
22	Analog Output Ch2 high byte (Slot#7)	0x6400	0x03
23	Analog Output Ch3 low byte (Slot#7)	0x6400	0x04
24	Analog Output Ch3 high byte (Slot#7)	0x6400	0x04
25	Special Input LL byte Ch#0 (Slot#4)	0x3000	0x01
26	Special Input LH byte Ch#0 (Slot#4)	0x3000	0x02
27	Special Input HL byte Ch#0 (Slot#4)	0x3000	0x03
28	Special Input HH byte Ch#0 (Slot#4)	0x3000	0x04
29	Special Input LL byte Ch#1 (Slot#4)	0x3000	0x05
30	Special Input LH byte Ch#1 (Slot#4)	0x3000	0x06
31	Special Input HL byte Ch#1 (Slot#4)	0x3000	0x07
32	Special Input HH byte Ch#1 (Slot#4)	0x3000	0x08
33	Special Output byte Ch#0 (Slot#4)	0x3200	0x01
34	Special Output byte Ch#1 (Slot#4)	0x3200	0x02

6. 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.

6.1. 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	0x00	Device type	unsigned32	ro		Statement of device type
0x1001	0x00	Error register	unsgined8	ro		Error register
0x1003	0x00	Predefine error field	unsgined8	rw	0	Number of error states stored
	0x01	Standard error filed	unsigned32	ro	0	Error state are stored
0x1005	0x00	COB-ID sync message	unsigned32	rw	0x80000080	Identifier of the Sync message
0x1006	0x00	Communication cycle period	unsigned32	rw	0	Communication cycle period in. '0' if not used
0x1007	0x00	Synchronous Window Length	unsigned32	rw	0	Synchronous Window Length
0x1008	0x00	Manufacturer device name	visible string	ro	RIO3-CAN	Device name of the Adapter
0x1009	0x00	Manufacturer hardware version	visible string	ro	100	H/W version description
0x100A	0x00	Manufacturer software version	visible string	ro	100	Software version number
0x100C	0x00	Guard time	unsigned16	rw	0x00C8	Interval between two guard telegrams. Is set by the NMTmaster.(ms)
0x100D	0x00	Life time factor	unsgined8	rw	2	Life time factor * guard time = life time(watchdog for life guarding)
0x1010	0x00	Largest sub-index supported	unsigned32	ro	1	Number of store options
	0x01	Save all parameters	unsigned32	rw	1	Store all parameters 0x65766173(ASCII : save)
0x1011	0x00	Restore default parameters	unsgined8	ro	1	Number of restore options
	0x01	Restore all default	unsigned32	rw	1	Restore all default parameters 0x64616F6C(ASCII : load)
0x1012	0x00	COB-ID Time Stamp	unsigned32	rw	100	COB-ID Time Stamp
0x1014	0x00	COB-ID emergency message	unsigned32	ro	0x80+node ID	COB-ID of the emergency object
0x1015	0x00	Inhibit time EMCY	unsigned16	rw	0	COB-ID SYNC
0x1016	0x00	Number of entries	unsgined8	ro	5	Heartbeat time value
	0x01	Consumer Heartbeat time1	unsigned32	rw	0	
	0x02	Consumer Heartbeat time2	unsigned32	rw	0	
	...	...	...	...	...	
	0x05	Consumer Heartbeat time5	unsigned32	rw	0	
0x1017	0x00	Producer Heartbeat time	unsigned16	rw		Producer Heartbeat time (0~127)
0x1018	0x00	Identity object	unsgined8	ro	4	Identity Object
	0x01	Manufacturer ID	unsigned32	ro	0x0000029D	
	0x02	Product code	unsigned32	ro	0x00009060	
	0x03	Revision number	unsigned32	ro	0x00000100	
	0x04	Serial number	unsigned32	ro		
0x1019	0x00	Synchronous counter overflow value	unsigned16	rw	0	Synchronous counter overflow value
0x1029	0x00	Number of error	unsgined8	ro	1	Error Behavior
	0x01	Communication Error	unsgined8	rw	0	
0x1400	0x00	Number of entries	unsgined8	ro	5	Receive PDO Communication
	0x01	COB-ID used by PDO	unsigned32	rw	0x200+nodeID	Parameter 1-32
	0x02	Transfer type	unsgined8	rw		
…	…	…				
0x141F	0x00	Number of entries	unsgined8	ro	5	
	0x01	COB-ID used by PDO	unsigned32	rw		
	0x02	Transfer type	unsgined8	rw		
0x1600	0x00	Number of entries	unsgined8	rw	8	Receive PDO Mapping

	0x01	1. Mapping Entry 1	unsigned32	rw		Parameter 1-32
	…	…				
	0x08	8. Mapping Entry 8	unsigned32	rw		
…	…	…				
0x161F	0x00	Number of entries	unsgined8	rw	8	
	0x01	1. Mapping Entry 1	unsigned32	rw		
	…	…				
	0x08	8. Mapping Entry 8	unsigned32	rw		
0x1800	0x00	Number of entries	unsgined8	ro	5	Transmit PDO Communication
	0x01	COB-ID used by PDO	unsigned32	rw	0x180+nodeID	Parameter 1-32
	0x02	Transfer type	unsgined8	rw		
…	…	…				
0x181F	0x00	Number of entries	unsgined8	ro	5	
	0x01	COB-ID used by PDO	unsigned32	rw		
	0x02	Transfer type	unsgined8	rw		
0x1A00	0x00	Number of entries	unsgined8	rw	8	Transmit PDO Mapping
	0x01	1. Mapping Entry 1	unsigned32	rw		Parameter 1-32
	…	…				
	0x08	8. Mapping Entry 8	unsigned32	rw		
…	…	…				
0x1A1F	0x00	Number of entries	unsgined8	rw	8	
	0x01	1. Mapping Entry 1	unsigned32	rw		
	…	…				
	0x08	8. Mapping Entry 8	unsigned32	rw		

- **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

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 0x1008, Manufacturer Device Name**

The object contains the length of the time window for synchronous PDOs in us. It is 0 if not used.

- **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 milliseconds. 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 Adapter 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 milliseconds. 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 0x1029, Error behavior**

The object specifies to which state an I/O module shall be set, when a communication error, output error or input error is detected.

Value	Description
00h	Change to NMT state Pre-operational (only if currently in NMT state Operational)
01h	No change of the NMT state
02h	Change to NMT state Stopped

- **Object 0x1400 ~ 0x141F 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 resigned 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 ~ 0x161F, 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. 32 bytes can be transmitted in a PDO, the sum of the valid object lengths must not exceed 252 (32Byte*8Bit = 256 but, Fn-bus specification is Max.252bytes)

- **Object 0x1800 ~ 0x181F, 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.

Attention :

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

- **Object 0x1A00 ~ 0x1A1F, 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 252

6.2. Manufacturer Specific Profile Area

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

*0x2100(Slot#0=NA), 0x2101(Slot#1), 0x2102(Slot#2)...

Index	Sub-Index	Name	Type	Attribute	Default	Meaning
0x20F0	0x00	Number of entries	unsigned8	ro	0x4F	
	0x01	Node id current switch value	unsigned8	ro		Node ID value
	0x02	Number of module	unsigned16	ro		Number of Module
	0x03	Run command 0 : IO_Available 1 : IO_Ready 2 : IO_Idle	unsigned16	ro	0	

		3 : IO_Run 4 : IO_Timeout 5 : IO_Minorfault 6 : IO_Unrecoveryfault 7 : Reset				
	0x04	All module id	unsigned16*n	ro		All module name
	0x05	Baudrate	unsigned16	ro		Baudrate Setting value 0x03E8 : 1000(default)
	0x06	PDO Mapping Method 0 : DS301(Auto) (default) 2 : Manual	unsigned8	rw	0	
	0x0C	Status, error information	unsigned32	ro	0x00000003	
	0x0E	All size of output bit	unsigned16	ro		
	0x0F	All size of input bit	unsigned16	ro		
	0x10	All size of output byte	unsigned16	ro		
	0x11	All size of input byte	unsigned16	ro		
0x21xx	0x04	Module ID	unsigned16	ro		
	0x08	Address of output byte	unsigned16	ro		
	0x09	Address of input byte	unsigned32	ro		
	0x0C	Fnbus Status, error information	unsigned32	ro		
	0x0E	Size of output bit	unsigned16	ro		
	0x0F	Size of input bit	unsigned16	ro		
	0x12	IO Data output#0	unsigned8*n	rw		
	0x13	IO Data input#0	unsigned8*n	ro		
	0x16	Parameter length	unsigned16	ro		
	0x17	Parameter data	unsigned8*n	rw		
	0x18	Hardware Rev	unsigned32	ro		
	0x19	Firmware Rev	unsigned32	ro		
	0x1A	Firmware release date	unsigned32	ro		
	0x1B	Inspection date	unsigned16	ro		
	0x20	String, Module description	unsigned8*n	ro		
	0x50	Parameter Data 0	Unsigned32	rw		
	0x5F	Parameter Data F	Unsigned32	rw		
0x3000	0x00	Read special input data	unsigned8	ro	0xFE	Read Special Input 8 bit
	0x1	8-bit special input 1	unsigned8	ro		
	…	…	...	...	...	
	0xFE	8-bit special input 254	unsigned8	ro		
0x3005	0x00	Enable special input interrupts	unsigned8	rw	1	
0x3006	0x00	Number of entries	unsigned8	ro	0xFE	Special input Interrupt Mask any change for Special Inputs
	0x01	Special input interrupt mask any change for special input 1	unsigned8	rw	0xFF	
	...	...	...	...	...	
	0xFE	Special input interrupt mask any change for special input 254	unsigned8	rw	0xFF	
0x3200	0x00	Number of entries	unsigned8	ro	0xFE	number of entries
	0x01	8-bit special output 1	unsigned8	rw	0	1st special output block
	…	…	...	...	...	
	0xFE	8-bit special output 254	unsigned8	rw	0	64st special output block

0x3206	0x00	Number of entries	unsigned8	ro	0xFE	Error Mode 8 Special output
	0x01	Faultmode for special output 1	unsigned8	rw	0xFF	
	...	...	...	...	...	
	0xFE	Faultmode for special output 254	unsigned8	rw	0xFF	
0x3207	0x00	Number of entries	unsigned8	ro	0xFE	Fault State 8 output lines
	0x01	Fault state for special output 1	unsigned8	rw	0	
	...	...	...	...	...	
	0xFE	Fault state for special output 254	unsigned8	rw	0	

- **Object 0x2100(Slot#0=NA), Manufacture Information.**

Index	Sub	Name	Type	Attribute	Default	Meaning
0x2100	0x00	number of entries	unsigned8	ro	0x4F	number of entries(slot number)
	0x04	module id	unsigned16	ro	0	

- **Object 0x2101(Slot#1=IO), ... , 0x213F(Slot#63=IO), IO Module Information.**

Index	Sub	Name	Type	Attribute	Default	Meaning
0x213F	0x00	number of entries	unsigned8	ro	0x4F	number of entries(slot number)
	0x04	module id	unsigned16	ro	0	

- **Object 0x20F0, Network Adapter Status.**

This object contains the Network Adapter status.
- Sub-index 2 contains rotary switch value set Node ID from NA.
- Sub-index 5 contains Dip switch value set baudrate from NA. If you write 5(125), you can read 7D$_{h.}$
- Sub-index 6 set Master's PDO mapping method. By default, DS301 Auto mapping. PDO1 is digital IO and PDO2~3 analog IO fixed. From PDO4 additional IO slot is used in sequence. Manual mapping set all digital IO, second analog IO and Special IO lastly. So you need separate master.

- **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 0x3005, Special Inputs Interrupt.**

This object shall enable and disable special module the interrupt behavior without changing the interrupt mask. By default, special input activates an interrupt.

- **Object 0x3005, Special Inputs Interrupt.**

This object determines, which input port lines shall activate an interrupt by positive or/and negative edge detection.

- **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 0x3206, Special Output Error Mode**

This object is used to define whether the outputs change to a pre-defined error status (see object 0x3207) 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 0x3207 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 0x3207, Special 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 0x3206

6.3. Standard Device Profile Area – DS401

- **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 0x6005, Global Interrupt Enable Digital 8-bit**

This object shall enable and disable globally the interrupt behavior without changing the interrupt masks. In event-driven mode the device transmits the input values depending on the interrupt masks in objects 6006_h, 6007_h, and 6008_h. If the object is not supported, the device shall behave accordingly to the default value.

- **Object 0x6006, Interrupt Mask Any Change 8-bit**

This object determines, which input port lines shall activate an interrupt by positive or/and negative edge detection.
If the object is not supported the device shall behave accordingly to the default value.

- **Object 0x6007, Interrupt Mask Low-to-High 8-bit**

This object determines, which input port lines shall activate an interrupt by positive edge detection (logical 0 to 1). Done for groups of 8 lines. The values shall be in an ”OR” connection to the values of 6006_h object (Interrupt mask any change 8-bit). If inputs are inverted by 6002_h object (polarity input 8-bit), the positive logical edge shall correspond to negative physical edge.
0 = Interrupt Disabled
1 = Interrupt Enabled

- **Object 0x6008, Interrupt Mask High-to-Low 8-bit**

This object determines, which input port lines shall activate an interrupt by negative edge detection (logical 1 to 0). Done for groups of 8 lines. The values shall be in an ”OR” connection to the values of 6006_h object (Interrupt mask any change 8-bit). If inputs are inverted by 6002h object (polarity input 8-bit), the negative

logical edge shall correspond to positive physical edge.
0 = Interrupt Disabled
1 = Interrupt Enabled

- **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 0x6421, Analog Input Interrupt Trigger Selection**

This object determines, which events shall cause an interrupt for a specific channel. All bits set to 1b shall trigger the corresponding analog input. If the object is not supported, the device shall behave accordingly to the default value.

Format:

Bit	Value	Meaning
0	0b	Upper limit not exceeded
	1b	Upper limit exceeded
1	0b	Input not below lower limit
	1b	Input below lower limit
2	0b	Input not changed by more than delta
	1b	Input changed by more than delta
3	0b	Input not reduced by more than negative delta
	1b	Input reduced by more than negative delta
4	0b	Input not increased by more than positive delta

	1b	Input increased by more than positive delta
reversed	0b	Reserved for future use

- **Object 0x6423, Analog Input Global Interrupt Enable**

This object shall enable and disable globally the interrupt behavior without changing the interrupt mask. By default, no analog input activates an interrupt.

0 = global interrupt disabled
1 = global interrupt enabled

- **Object 0x6424, Analog Input Interrupt Upper Limit Integer**

If enabled (see 6423h object), an interrupt is triggered when the analog input is equal or rises above the given value. The value shall be always left adjusted. As long as the trigger condition is met, every change of the analog input data generates a new interrupt, if there is no additional trigger condition, e.g. an input interrupt delta (6426h).

- **Object 0x6425, Analog Input Interrupt Lower Limit Integer**

If enabled (see 6423h object), an interrupt is triggered when the analog input falls below the given value. The value shall be always left adjusted. As long as the trigger condition is met, every change of the analog input data generates a new interrupt, if there is no additional trigger condition, e.g. an input interrupt delta (6426h).

- **Object 0x6426, Analog Input Interrupt Delta Limit Integer**

This object shall set the delta value (rising or falling above or below the last communicated value) for interrupt-enabled analog inputs (see 6423h object).

- **Object 0x6427, Analog Input Interrupt Negative Delta Unsigned**

This object shall set the negative delta value (falling below the last communicated value) for interrupt-enabled analog inputs (see 6423h object).

- **Object 0x6428, Analog Input Interrupt Positive Delta Unsigned**

This object shall set the negative delta value (rising below the last communicated value) for interrupt-enabled analog inputs (see 6423h object).

- **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

7. MODBUS Interface

7.1. MODBUS Interface Register/Bit Map

- **Register Map**

Start Address	Read/Write	Description	Func. Code
0x0000 ~ 0x007E	Read	Process input image registers (Real Input Register)	3,4,23
0x0800 ~ 0x087E	Read/Write	Process output image registers (Real Output Register)	3,16,23
0x1000 *	Read	Adapter Identification special registers.	3,4,23
0x1020 *	Read/Write	Adapter Watchdog, other time special register.	3,4,6,16,23
0x1100 *	Read/Write	Adapter Information special registers.	3,4,6,16,23
0x2000 *	Read/Write	Expansion Slot Information special registers.	3,4,6,16,23

* The special register map must be accessed by read/write of every each address (one address).

- **Register Bit Map**

Start Address	Read/Write	Description	Func. Code
0x0000~	Read	Process input image bits All input registers area are addressable by bit address. Size of input image bit is size of input image register * 16.	2
0x1000~	Read/Write	Process output image bits All output registers area are addressable by bit address. Size of output image bit is size of output image register * 16.	1,5,15

7.2. Supported MODBUS Function Codes

Function Code	Function	Description
1(0x01)	Read Coils (Read output bit)	This function code is used to read from 1 to 2000 contiguous status of coils in a remote device. The Request PDU specifies the starting address, i.e. the address of the first coil specified, and the number of coils. In the PDU Coils are addressed starting at zero. Therefore coils numbered 1-16 are addressed as 0-15. The coils in the response message are packed as one coil per bit of the data field. Status is indicated as 1= ON and 0= OFF.

2(0x02)	Read Discrete Inputs (Read input bit)	This function code is used to read from 1 to 2000 contiguous status of discrete inputs in a remote device. The Request PDU specifies the starting address, i.e. the address of the first input specified, and the number of inputs. In the PDU Discrete Inputs are addressed starting at zero. Therefore Discrete inputs numbered 1-16 are addressed as 0-15. The discrete inputs in the response message are packed as one input per bit of the data field. Status is indicated as 1= ON; 0= OFF.
3(0x03)	Read Holding Registers (Read output word)	This function code is used to read the contents of a contiguous block of holding registers in a remote device. The Request PDU specifies the starting register address and the number of registers. The register data in the response message are packed as two bytes per register, with the binary contents right justified within each byte. For each register, the first byte contains the high order bits and the second contains the low order bits.

7.2.1. 8(0x08)Diagnostics

Sub-function 0x0000(0) Return Query Data

The data passed in the request data field is to be returned (looped back) in the response.
The entire response message should be identical to the request.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x0000(0)	Any	Echo Request Data	

Sub-function 0x0001(1) Restart Communications Option

The remote device could be initialized and restarted, and all of its communications event counters are cleared.
Especially, data field 0x55AA make the remote device to restart with factory default setup of EEPROM.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x0001(1)	0x0000, 0xFF00	Echo Request Data	Reset Only

Sub-function 0x000B(11) Return Bus Message Count

The response data field returns the quantity of messages that the remote device has detected on the communications system since its last restart, clear counters operation, or power-up.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x000B(11)	0x0000	Total Message Count	

Sub-function 0x000C(12) Return Bus Communication Error Count

The response data field returns the quantity of CRC errors encountered by the remote device since its last restart, clear counters operation, or power-up.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x000C(12)	0x0000	CRC Error Count	

Sub-function 0x000D(13) Return Bus Exception Error Count

The response data field returns the quantity of MODBUS exception responses returned by the remote device since its last restart, clear counters operation, or power-up.
Exception responses are described and listed in section 3.2.11.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x000D(13)	0x0000	Exception Error Count	

Sub-function 0x000E(14) Return Slave Message Count

The response data field returns the quantity of messages addressed to the remote device, or broadcast, that the remote device has processed since its last restart, clear counters operation, or power–up.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x000E(14)	0x0000	Slave Message Count	

Sub-function 0x000F(15) Return Slave No Response Count
The response data field returns the quantity of messages addressed to the remote device for which it has returned no response (neither a normal response nor an exception response), since its last restart, clear counters operation, or power–up.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x000F(15)	0x0000	Slave No Response Count	

Sub-function 0x0064(100) Return Slave MODBUS, Internal Status
The response data field returns the status of MODBUS and Internal addressed to the remote device.
This status values are identical with status 1word of input process image.

Sub-function	Data Field (Request)	Data Field (Response)	Description
0x0064(100)	0x0000	MODBUS, Internal Status	Same as status 1word

7.2.2. Error Response

In an exception response, the server sets the MSB of the function code to 1. This makes the function code value in an exception response exactly 80 hexadecimal higher than the value would be for a normal response.

- **Exception Codes**

Exception Code	Name	Description
01	Illegal Function	The function code received in the query is not an allowable action for the server (or slave).
02	Illegal Data Address	The data address received in the query is not an allowable address for the server (or slave).
03	Illegal Data Value	A value contained in the query data field is not an allowable value for server (or slave).
04	Slave Device Failure	An unrecoverable error occurred while the server (or slave) was attempting to perform the requested action.
05	Acknowledge	The server (or slave) has accepted the request and is processing it, but a long duration of time will be required to do so.
06	Slave Device Busy	Specialized use in conjunction with programming commands. The server (or slave) is engaged in processing a long–duration program command. The client (or master) should retransmit the message later when the server (or slave) is free.
08	Memory Parity Error	The server (or slave) attempted to read record file, but detected a parity error in the memory. The client (or master) can retry the request, but service may be required on the server (or slave) device.
0A	Gateway Path Unavailable	Specialized use in conjunction with gateways, indicates that the gateway was unable to allocate an internal communication path from the input port to the output port for processing the request.

7.3. MODBUS Special Register Map

The special register map can be accessed by function code 3, 4, 6 and 16. Also the special register map must be accessed by read/write of every each address (one address).

7.3.1. Adapter Identification Special Register (0x1000, 4096)

Address	Access	Type, Size	Description
0x1000(4096)	Read	1word	Vendor ID = 0x29D(669), HITACHI
0x1001(4097)	Read	1word	Device type = 0x000C, Network Adapter
0x1002(4098)	Read	1word	Product code = 0x9060 (RIO3-CAN)
0x1003(4099)	Read	1word	Firmware revision, if 0x0101, revision 1.01
0x1004(4100)	Read	2word	Product unique serial number
0x1005(4101)	Read	String Up to 34byte	Product name string (ASCII) "RIO3-CAN,CANopen Adapter, G-Bus"
0x1010(4112)	Read	2word	Firmware release date
0x1013(4115)	Read	1word	Firmware Code = 0x9261
0x101E(4126)	Read	7word - 1word - 1word - 1word - 1word - 1word - 2word	Composite Id of following address 0x1100(4352), Modbus RS232 Node. (Fixed 0x0001) 0x1000(4096), Vendor ID 0x1001(4097), Device type 0x1002(4098), Product code 0x1003(4099), Firmware revision 0x1004(4100), Product serial number

- String Type consist of valid string length (first 1word) and array of characters

7.3.2. Adapter Identification Special Register (0x1100, 4352)

Address	Access	Type, Size	Description
0x1102(4354)	Read	1word	Start address of input image word register. =0x0000
0x1103(4355)	Read	1word	Start address of output image word register. =0x0800
0x1104(4356)	Read	1word	Size of input image word register.
0x1105(4357)	Read	1word	Size of output image word register.
0x1106(4358)	Read	1word	Start address of input image bit. = 0x0000
0x1107(4359)	Read	1word	Start address of output image bit. =0x1000
0x1108(4360)	Read	1word	Size of input image bit.
0x1109(4361)	Read	1word	Size of output image bit.
0x110A(4362)	Read	1word	Update time for cyclic data change (same as 0x1028)
0x110C(4364)	Read	1word	Field power status
0x110D(4365)	Read	1word	Current Rotary Switch State and Dip Switch Status (MSB) ex) Rotary SW(0x02) , Dip SW(0x01) = 0x8201 (0x8000 field power on)
0x110E(4366)	Read	up to	Expansion slot's GT-number including GN

		33word	First 1word is adapter's number, if GN-9289, then 0x9289
0x1110(4368)	Read	1word	Number of expansion slot
0x1113(4371)	Read	up to 33word	Expansion slot Module Id. Refer to Appendix A.1 Product List.
			First 1word is adapter's module id.
0x1119(4377)	Read	1word	Hi byte is Modbus status, low byte is internal status.
			Zero value means 'no error'.
0x111D(4381)	Read	1word	Adapter RIO3 Series Revision. If 0x013C, RIO3 Series Revision is 1.60

*After the system is reset, the new "Set Value" action is applied.
** If the slot location is changed, set default value automatically (all expansion slot are live).

7.3.3. Expansion Slot Information Special Register (0x2000, 8192)

Each expansion slot has 0x20(32) address offset and same information structure.

Slot#1 0x2000(8192)~0x201F(8223) Slot#2 0x2020(8224)~0x203F(8255)
Slot#3 0x2040(8256)~0x205F(8287) Slot#4 0x2060(8288)~0x207F(8319)
Slot#5 0x2080(8320)~0x209F(8351) Slot#6 0x20A0(8352)~0x20BF(8383)
Slot#7 0x20C0(8384)~0x20DF(8415) Slot#8 0x20E0(8416)~0x20FF(8447)
Slot#9 0x2100(8448)~0x211F(8479) Slot#10 0x2120(8480)~0x213F(8511)
Slot#11 0x2140(8512)~0x215F(8543) Slot#12 0x2160(8544)~0x217F(8575)
Slot#13 0x2180(8576)~0x219F(8607) Slot#14 0x21A0(8608)~0x21BF(8639)
Slot#15 0x21C0(8640)~0x21DF(8671) Slot#16 0x21E0(8672)~0x21FF(8703)
Slot#17 0x2200(8704)~0x221F(8735) Slot#18 0x2220(8736)~0x223F(8767)
Slot#19 0x2240(8768)~0x225F(8799) Slot#20 0x2260(8800)~0x227F(8831)
Slot#21 0x2280(8832)~0x229F(8863) Slot#22 0x22A0(8864)~0x22BF(8895)
Slot#23 0x22C0(8896)~0x22DF(8927) Slot#24 0x22E0(8928)~0x22FF(8959)
…..
Slot#63 0x27C0(10176)~0x27DF(10207)

Address Offset	Expansion Slot#1	Expansion Slot#2	Expansion Slot#3	Expansion Slot#4	……..	Expansion Slot#63
+ 0x00(+0)	0x2000(8192)	0x2020(8224)	0x2040(8256)	0x2060(8288)	…….	0x27C0(10176)
+ 0x01(+1)	0x2001(8193)	0x2021(8225)	0x2041(8257)	0x2061(8289)	…….	0x27C1(10177)
+ 0x02(+2)	0x2002(8194)	0x2022(8226)	0x2042(8258)	0x2062(8290)	…….	0x27C2(10178)
+ 0x03(+3)	0x2003(8195)	0x2023(8227)	0x2043(8259)	0x2063(8291)	…….	0x27C3(10179)
+ 0x04(+4)	0x2004(8196)	0x2024(8228)	0x2044(8260)	0x2064(8292)	…….	0x27C4(10180)
+ 0x05(+5)	0x2005(8197)	0x2025(8229)	0x2045(8261)	0x2065(8293)	…….	0x27C5(10181)
+ 0x06(+6)	0x2006(8198)	0x2026(8230)	0x2046(8262)	0x2066(8294)	…….	0x27C6(10182)
+ 0x07(+7)	0x2007(8199)	0x2027(8231)	0x2047(8263)	0x2067(8295)	…….	0x27C7(10183)
+ 0x08(+8)	0x2008(8200)	0x2028(8232)	0x2048(8264)	0x2068(8296)	…….	0x27C8(10184)
+ 0x09(+9)	0x2009(8201)	0x2029(8233)	0x2049(8265)	0x2069(8297)	…….	0x27C9(10185)
+ 0x0A(+10)	0x200A(8202)	0x202A(8234)	0x204A(8266)	0x206A(8298)	…….	0x27CA(10186)
+ 0x0B(+11)	0x200B(8203)	0x202B(8235)	0x204B(8267)	0x206B(8299)	…….	0x27CB(10187)
+	0x200C(8204)	0x202C(8236)	0x204C(8268)	0x206C(8300)	…….	0x27CC(10188)

0x0C(+12)						
+ 0x0D(+13)	0x200D(8205)	0x202D(8237)	0x204D(8269)	0x206D(8301)	…….	0x27CD(10189)
+ 0x0E(+14)	0x200E(8206)	0x202E(8238)	0x204E(8270)	0x206E(8302)	…….	0x27CE(10190)
+ 0x0F(+15)	0x200F(8207)	0x202F(8239)	0x204F(8271)	0x206F(8303)	…….	0x27CF(10191)
+ 0x10(+16)	0x2010(8208)	0x2030(8240)	0x2050(8272)	0x2070(8304)	…….	0x27D0(10192)
+ 0x11(+17)	0x2011(8209)	0x2031(8241)	0x2051(8273)	0x2071(8305)	…….	0x27D1(10193)
+ 0x12(+18)	0x2012(8210)	0x2032(8242)	0x2052(8274)	0x2072(8306)	…….	0x27D2(10194)
+ 0x13(+19)	0x2013(8211)	0x2033(8243)	0x2053(8275)	0x2073(8307)	…….	0x27D3(10195)
+ 0x14(+20)	0x2014(8212)	0x2034(8244)	0x2054(8276)	0x2074(8308)	…….	0x27D4(10196)
+ 0x15(+21)	0x2015(8213)	0x2035(8245)	0x2055(8277)	0x2075(8309)	…….	0x27D5(10197)
+ 0x16(+22)	0x2016(8214)	0x2036(8246)	0x2056(8278)	0x2076(8310)	…….	0x27D6(10198)
+ 0x17(+23)	0x2017(8215)	0x2037(8247)	0x2057(8279)	0x2077(8311)	…….	0x27D7(10199)
+ 0x18(+24)	0x2018(8216)	0x2038(8248)	0x2058(8280)	0x2078(8312)	…….	0x27D8(10200)
+ 0x19(+25)	0x2018(8217)	0x2038(8249)	0x2058(8281)	0x2078(8313)	…….	0x27D9(10201)
+ 0x1A(+26)	0x201A(8218)	0x203A(8250)	0x205A(8282)	0x207A(8314)	…….	0x27DA(10202)
+ 0x1B(+27)	0x201B(8219)	0x203B(8251)	0x205B(8283)	0x207B(8315)	…….	0x27DB(10203)
+ 0x1C(+28)	0x201C(8220)	0x203C(8252)	0x205C(8284)	0x207C(8316)	…….	0x27DC(10204)
+ 0x1D(+29)	0x201D(8221)	0x203D(8253)	0x205D(8285)	0x207D(8317)	…….	0x27DD(10205)
+ 0x1E(+30)	0x201E(8222)	0x203E(8254)	0x205E(8286)	0x207E(8318)	…….	0x27DE(10206)
+ 0x1F(+31)	0x201F(8223)	0x203F(8255)	0x205F(8287)	0x207F(8319)	…….	0x27DF(10207)

Address Offset	Access	Type, Size	Description
+ 0x02(+2) **	Read	1word	Input start register address of input image word this slot.
+ 0x03(+3) **	Read	1word	Input word's bit offset of input image word this slot.
+ 0x04(+4) **	Read	1word	Output start register address of output image word this slot.
+ 0x05(+5) **	Read	1word	Output word's bit offset of output image word this slot.
+ 0x06(+6) **	Read	1word	Input bit start address of input image bit this slot.
+ 0x07(+7) **	Read	1word	Output bit start address of output image bit this slot.
+ 0x08(+8) **	Read	1word	Size of input bit this slot
+ 0x09(+9) **	Read	1word	Size of output bit this slot
+ 0x0A(+10)**	Read	n word	Read input data this slot
+ 0x0B(+11)**	Read/Write	n word	Read/write output data this slot
+ 0x0E(+14)	Read	1word	RIO3-number, if RIO3- XDP8, returns 0x1238 (hexID)
+ 0x0F(+15)	Read	String	First 1word is length of valid character string.
		upto 72byte	If RIO3- XDP8, returns

			“00 1E 52 54 2D 31 32 33 38 2C 20 38 44 49 2C 20 32 34 56 64 63 2C 20 55 6E 69 76 65 72 73 61 6C 00 00”
			Valid character size = 0x001E =30 characters,
			“RIO3- XDP8, 8DI, 24Vdc, Universal”
+ 0x10(+16)	Read	1word	Size of configuration parameter byte
+ 0x11(+17)**	Read/Write	n word	Read/write Configuration parameter data, up to 8byte. Refer to A.2 ***
+ 0x17(+23)	Read	1word	Firmware Revision
			ex) 0x0001 (Major revision 0 /Minor revision 1)

* After the system is reset, the new “Set Value” action is applied.
** Nothing of output, input, memory or configuration parameter corresponding slot returns Exception 02.

8. Troubleshooting

8.1. How to diagnose by LED indicator

LED Status	Cause	Action
All LED turns off	- No power	- Check main power Cable
MOD LED is red	- Occurrence critical error in firmware	- Contact Sales team and send module for repair.
ERR LED Single flash	- At least one of the error counters of the CAN controller has reached or exceeded the warning limit.	- Check the communication cable or communication setting.
ERR LED Double flash	- A guard event (NMT-Slave or NMT-Master) or a Heartbeat event has occurred.	- Adjust guard time or life time factor. If Heartbeat event has occurred, module power reset.
ERR LED Triple flash	- The SYNC message has not been received within then configured communication cycle period time out (see index 0x1006)	- Adjust transfer type setting.
ERR LED ON	- The CAN controller is bus off.	- The CAN controller is bus off.
IOS LED turns off	- Device may not be powered.	- Check main power Cable
IOS LED flashes red	- Adapter has no expansion module	- Add one or more expansion modules.
IOS LED is red	One or more expansion module occurred in fault state. - Detected invalid expansion module ID. - Overflowed Input/Output Size - Too many expansion module - Initialization failure - Communication failure. - Changed expansion module configuration. - Mismatch vendor code between adapter and expansion module.	- Use expansion slot up to 63. - Compose that IO total size is not excess. - Check status of expansion IO connection. - Check the vendor code of module.
Field Power LED turns off	- Field power is not supplied.	- Check main power Cable

		- Contact Sales team and send module for repair.
System Power LED turns off	- System power is not supplied.	- Check main power Cable - Contact Sales team and send module for repair.

8.2. 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.	RIO3-Number	Description	ID (hex)
Digital Input Module			
01	RIO3-XDP8	8 Points, Universal, 24Vdc, 10RTB	1238
02	RIO3-XDP16C	16 Points, Universal, 24Vdc, 20P connector	123F
03	RIO3-XDP16T	16 Points, Universal, 24Vdc, 18RTB	12DF
04	RIO3-XDP32C	32 Points, Universal, 24Vdc, 40P connector	12FA
05	RIO3-XY16T	8 Sink Input / 8 Source Output with Diagnostic, 24Vdc	1428
06	RIO3-XAH4	4 Points, 240Vac, 10RTB	1904
Digital Output Module			
07	RIO3-YTP8	8 Points, Source, 24Vdc/0.5A, 10RTB	2328
08	RIO3-YTP16C	16 Points, Source, 24Vdc/0.3A, 20P connector	222F
09	RIO3-YTP16T	16 Points, Source, 24Vdc/0.3A, 18RTB	226F
10	RIO3-YTP32C	32 Points, Source, 24Vdc/0.3A, 40P connector	22CA
11	RIO3-YS4	4 Points, MOS Relay, 240Vdc/ac, 0.5A, 10RTB	2734
12	RIO3-YS8	8 Points, MOS Relay Output Terminal, 240Vdc, 0.5A	2738
13	RIO3-YR4	4 Points, Relay, 24Vdc/2A, 240Vac/2A, 10RTB	2744
Analog Input Module			
14	RIO3-LDC2	2ch load cell input unit, strain gauge	3002
15	RIO3-AX4I	4 Channels, 0~20, 4~20mA, 12bits, 10RTB	3114
16	RIO3-AXH4I	4 Channels, 0~20, 4~20mA, 16bits, 10RTB	3154
17	RIO3-AX8I	8 Channels, 0~20, 4~20mA, 12bits, 10RTB	3118
18	RIO3-AXH8I	8 Channels, 0~20, 4~20mA, 16bits, 10RTB	3158
19	RIO3-AX16IC	16 Channels, 0~20, 4~20mA, 12bits, 20P connector	311F
20	RIO3-AX16IT	16 Channels, 0~20, 4~20mA, 12bits, 18RTB	317F
21	RIO3-AX4V	4 Channels, 0~10, 0~5, 1~5Vdc, 12bits, 10RTB	3424
22	RIO3-AXH4V	4 Channels, 0~10, 0~5, 1~5Vdc, 16bits, 10RTB	3464
23	RIO3-AX8V	8 Channels, 0~10, 0~5, 1~5Vdc, 12bits, 10RTB	3428
24	RIO3-AXH8V	8 Channels, 0~10, 0~5, 1~5Vdc, 16bits, 10RTB	3468
25	RIO3-AX16VC	16 Channels, 0~10, 0~5, 1~5Vdc, 12bits, 20P connector	342F
26	RIO3-AX16VT	16 Channels, 0~10, 0~5, 1~5Vdc, 12bits, 18RTB	347F
27	RIO3-RTD4T	4 Channels, RTD, 10RTB	3704
28	RIO3-RTD8C	8 Channels, RTD, 20P connector	3708
29	RIO3-TC4T	4 Channels, Thermocouple, 10RTB	3804
30	RIO3-E3AC	AC Measurement	3901
Analog Output Module			
31	RIO3-AY4I	4 Channels, Current Output, 4~20mA, 12bits	4214
32	RIO3-AYH4I	4 Channels, Current Output, 4~20mA, 16bits	4254
33	RIO3-AY8I	8 CHANNELS CURRENT OUTPUT, 4~20mA, 12BIT	4218
34	RIO3-AY4V	4CH, 0~10Vdc, 12Bits, 10RTB	4424

35	RIO3-AYH4V	4CH, 0~10Vdc, 16Bits, 10RTB	4464
36	RIO3-AY8V	8CH, 0~10Vdc, 12Bits, 10RTB	4428
37	RIO3-AY16VC	16CH, 0~10Vdc, 12Bits, 20P Connector	442F
38	RIO3-AY16VT	16CH, 0~10Vdc, 12Bits, 18RTB	447F
Special Module			
39	RIO3-CU24L	High Speed Counter, 2CHs, 24Vdc, Encoder Input, 10RTB	
40	RIO3-RS232	1CH, RS 232, RTS/CTS, Full Duplex Type, 10RTB	5211
41	RIO3-RS485	1CH, RS 485, Half Full Duplex Type, 10RTB	5231
42	RIO3-PWM2	PWM Output, 2CHs, 0.5A/24Vdc, Source, 18RTB	5442
43	RIO3-PO2	Pulse Output, 2CHs, 0.5A/24Vdc, Source, 18RTB	5642
Power Module			
44	RIO3-SHD	Shield Module	7408
45	RIO3-0VDC	Common for 0Vdc	7508
46	RIO3-PSD	Power Expansion, In 24Vdc, Out 1A/5Vdc	7511
47	RIO3-24VDC	Common for 24Vdc	7518
48	RIO3-VDC	Common for 0Vdc, 24Vdc	7588
49	RIO3-PS	Field Power, 5/24/48 Vdc, 110/220 Vac	7641

A.2. Glossary

- System Power : The power for starting up CPU.
- Field Power : The power for input and output line.
- Terminator Resistor : Resistor for prevention reflected wave.
- EDS : Electronic Data Sheet.
- Sink : The method of in/output power supply if a device has no power source.
- Source : The method of in/output power supply if a device has the power source.

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 [mm²]		
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 [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

(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.