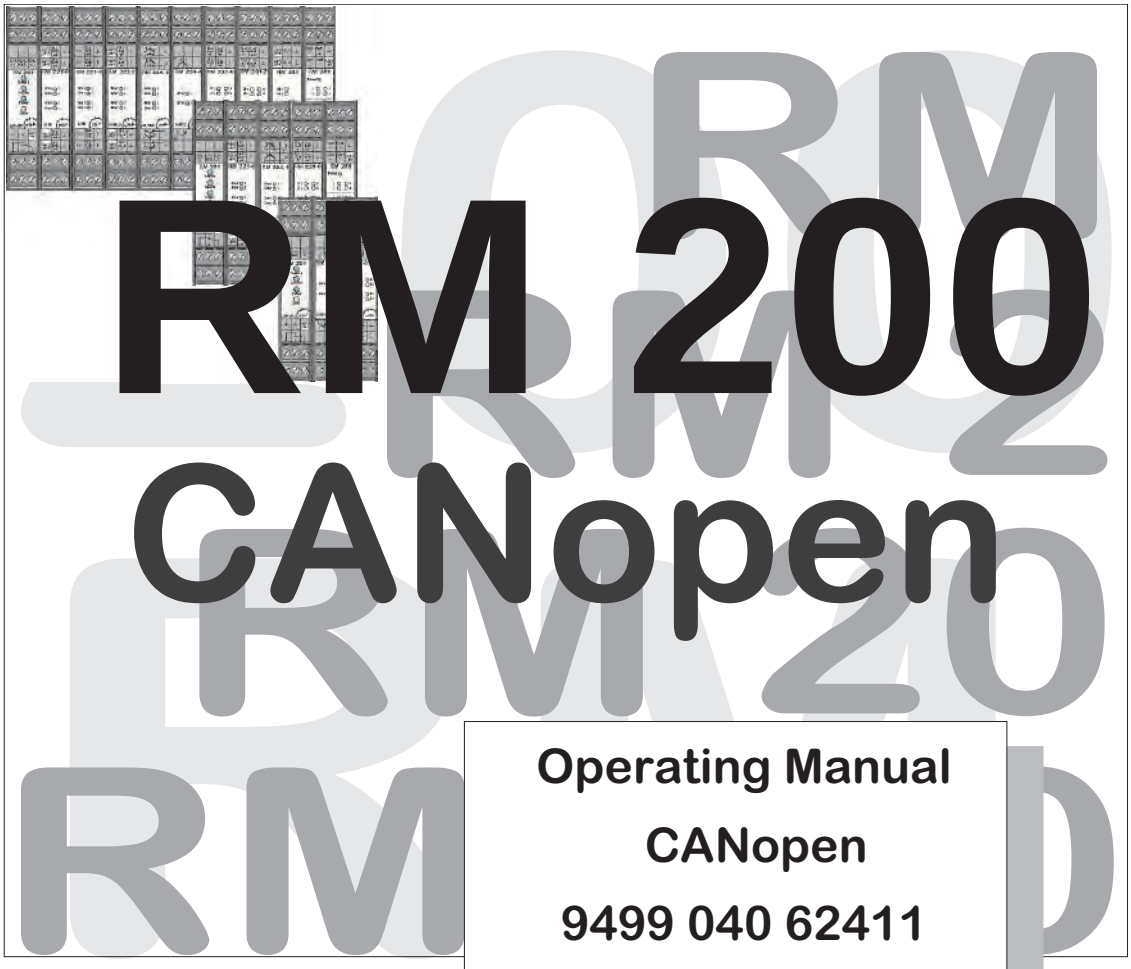
PMA Prozeß- und Maschinen-Automation GmbH

Modular I/O system



RM 200 CANopen

Operating Manual

CANopen

9499 040 62411

valid from: 8364

A company of the NEWAYS group

A publication of PMA Prozeß- und Maschinen Automation
P.O. Box 310229
D-34058 Kassel
Germany

Contents

1 Introduction

The input/output modules RM 200 with communication ports for CANopen or PROFIBUS-DP provide a high degree of flexibility when designing new plants. The compact, plug-in modules can be combined into cost-effective, de-centralized I/O islands. Due to the modular concept, type and number of the I/Os can be matched optimally to the requirements. Subsequent system extensions present no problems.

The fieldbus coupler module RM 201 (9407-738-20101) of the modular I/O system RM 200 is equipped with a CANopen interface for transmission of process data, parameters and configuration data. The connection is realized via screw-terminals. These serial communication interface permits connections to supervisory systems, visualization tools,etc.

Communication is according to the master/slave-principle. The coupler module RM 201 is always CANopen-slave.

Cable medium as well as physical and electrical interface properties:

- Network topology
  Linear bus with bus termination at both ends. Switchable termination resistance for RM 201.
- Transmission medium
  screened, twisted-pair cable
- Baudrates and cable length (without repeater)
  The maximum cable length depends on the used transmission rate.
  The baudrate of the RM 201 can be set via coding DIP-switches or can be recognized automatically.

Baudrate	Maximum cable length
10 / 20 /50 kbit/s	1000 m
100 kbit/s	800 m
125 kbit/s	500 m
250 kbit/s	250 m
500 kbit/s	100 m
800 kbit/s	50 m
1000 kbit/s	25 m

- Interface
  connectable with screw-/plug-in-terminals.
- Adressing
  Address settings via coding switches, range 01 ... 127, default 32

The modular I/O system RM 200 with CANopen interface offers many advantages with respect to handling and integration into a CAN network.

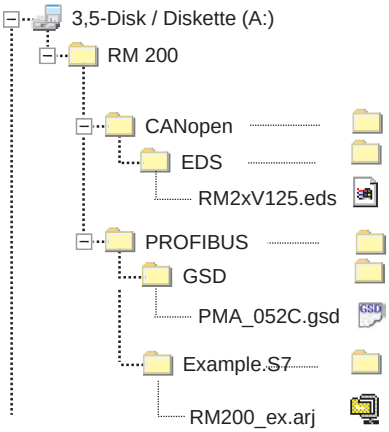
- Modules are pluggable in any order
  - up to 16 analog inputs per node
  - up to 16 analog outputs per node
  - up to 9 digital I/O modules per node
- Configuration of modules simply via CAN -configurator
- Broad range of available sensor and signal modules
- Plug and Play for the KS98+ - I/O-extension

This document describes the coupler module RM 201 in the **Software-Version 1.25** or later.

1.1 *Scope of delivery*

The engineering set consists of:

- disk



Only for RM 201:

RM2xV125.eds	Device description for CANopen, needed for CANopen-network configuration

Only for RM 202:

PMA_052C.gsd	Device description file, used for PROFIBUS-DP configuration tools
RM200_ex.arj	Project example in STEP® 7

- operating manual for PROFIBUS-DP
- operating manual for CANopen

2 Safety Instructions general

INSTRUMENT SAFETY

This instrument was built and tested according to VDE 0411 / EN61010-1 and was shipped in safe condition. The unit was tested before delivery and has passed the tests required in the test plan.

In order to maintain this condition and to ensure safe operation, the user must follow the hints and warnings given in these safety notes and operating instructions.

The unit is intended exclusively for use as a measuring and control instrument in technical installations.

The insulation meets standard EN 61010-1 with the values for overvoltage category, degree of contamination, operating voltage range and protection class specified in the operating instructions / data sheet.

The instrument must be operated only by trained persons. Maintenance and repair should be carried out only by trained, qualified personnel familiar with the relevant hazards.

The instrument may be operated within the specified environmental conditions (see data sheet) without impairing its safety.

The instrument is intended for mounting in an enclosure. Its contact safety is ensured by installation in a housing or switch cabinet.

UNPACKING THE INSTRUMENT

Remove instrument and accessories from the packing. Enclosed standard accessories:
Operating notes or operating instructions for the instrument (if necessary, fixing elements).

Check, if the shipment is correct and complete and if the instrument was damaged by improper handling during transport and storage.

WARNING!
If the instrument is so heavily damaged that safe operation seems impossible, the instrument must not be taken into operation.

We recommend to keep the original packing for shipment in case of maintenance or repair.

Caution! The instrument contains electrostatically sensitive components.
The special packing protects the instrument against damage by electrostatic discharge (ESD). Therefor, the instrument may be transported only in this packing. During mounting, the rules for protection against ESD must be followed.

MOUNTING

Mounting is done in dustfree and dry rooms, either in a panel or in the relevant socket of a 19- inch instrument carrier.

The ambient temperature at the place of installation must not exceed the permissible nominal temperature specified for operation in the data sheet.
When mounting several instruments at high packing density, sufficient ventilation must be provided to ensure correct function.

The sealing devices (e.g. sealing ring) required for the relevant protection type must also be fitted.

Two captive screws are provided at the instrument front for fixing the 19- inch module in the instrument carrier. With other instruments, the fixing elements delivered with the instrument must be used.
The instruments may be mounted only outside the explosion-hazarded area!

ELECTRICAL CONNECTIONS
All electrical wiring must conform to local standards (e.g. VDE 0100 in Germany).
The input leads must be kept separate from signal and mains leads.
The protective earth must be connected to the relevant terminal (in the instrument carrier).

The cable screening must be connected to the terminal for grounded measurement. In order to prevent stray electric interference, we recommend using twisted and screened input leads.
The electrical connections must be made according to the relevant connecting diagrams.

COMMISSIONING
Before instrument switch- on, ensure that the rules given below were followed:

- Ensure that the supply voltage corresponds to the specification on the type label.
- All covers required for contact safety must be fitted.
- Before instrument switch- on, check if other equipment and / or facilities connected in the same signal loop
- is / are not affected. If necessary, appropriate measures must be taken.
- On instruments with protection class I, the protective earth must be connected conductingly with the relevant terminal in the instrument carrier.
- The instrument must be operated only when mounted in its enclosure.

OPERATION
Switch on the supply voltage.
The instrument is now ready for operation. If necessary, a warm- up time of approx.
15 min. should be taken into account.

WARNING !
Any interruption of the protective earth in the instrument carrier can impair the instrument safety. Purposeful interruption is not permissible.
If the instrument is damaged to an extent that safe operation seems impossible, shut it down and protect it against accidental operation.

TROUBLE SHOOTING
Before checking the instrument, all possibilities of error in other equipment and connections (input leads, wiring, equipment connected in the output circuit) should be checked. If the trouble cannot be located by checking these points, we recommend returning the instrument to the manufacturer.

HINT
Note that primary elements (especially thermocouples) connected to the energized transmitter are grounded in many cases, i.e. that the insulation resistance during operation can be reduced considerably. In these cases, additional connection to earth is not permissible.

SHUT- DOWN
For permanent shut- down, disconnect the instrument from all voltage sources and protect it against accidental operation.

Before instrument switch- off, check that other equipment and / or facilities connected in the same signal loop is / are not affected. If necessary, appropriate measures must be taken.

MAINTENANCE, REPAIR AND MODIFICATION
The instrument needs no particular maintenance.

WARNING!
When opening the instruments, or when removing covers or components, live parts or terminals can be exposed.

Before carrying out such work, the instrument must be disconnected from all voltage sources.

After completing such work, re- shut the instrument and re-fit all covers and components. Check, if the specifications on the type label are still correct, and change them, if necessary.

When opening the instruments, electrostatically sensitive components can be exposed.
The following work may be carried
out only at workstations which are protected against ESD.

Modifications, maintenance and repair may be carried out only by trained, authorized persons. For this, the user is invited to contact the PMA service.

If a trouble was found to be due to a blown fuse, the cause must be determined and removed. For replacement, only fuses of the same type and current rating as the original fuse must be used.

Using repaired fuses, or short- circuiting the fuse socket is inadmissible.

EXPLOSION PROTECTION
Non-intrinsically safe instruments must not be operated in explosion-hazarded areas. Moreover, the output and input circuits of the instrument / instrument carrier must not lead into explosion-hazarded areas. Exceptions refer only to instruments for which a certificate of conformity exists. For these EX- instruments, the specifications in the relevant certificate of conformity and the local regulations for installation of electrical apparatus in explosion-hazarded areas must be taken into account additionally.

3 Hints on operation

3.1 *Mouting*

An RM 200 system comprises a basic module (housing) for mounting on a snap-on rail with 3, 5 or 10 sockets.
The left socket is generally reserved for bus coupler module CANopen **RM 201**. Dependent of requirements, I/O modules or dummies are fitted in the other sockets. The modules click into the basic module and can be released for replacement by means of simple tools.

The connecting terminals can be withdrawn easily from the the modules.

The plug-in cards must not be plugged in or withdrawn with the supply voltage switched on.

The basic modules are intended for DIN-rail mounting according to EN 50022. The mouting is carried out by locking the metal ledge on the back side below. For dismantling a basic module the metal ledge must be released.

Module installation into a basic module: Slide in the module at the respective place. Listen to the ‘click’ for proper engaging. The installation of the fieldbus coupler always must be placed at the absolutely left position. All other modules can be installed at any position (but see below). For removing: Release the two ledges and pull out the module.

Temperature modules like RM 224-x should be placed far away from modules with higher power demand, e.g. RM 252, RM 231-x, RM 201 etc..

The relay module RM 252 should not be mounted right of the RM 201.

Using a mixture of modules with four channels and two channels please place the ones with two channels right from the four channels ones.

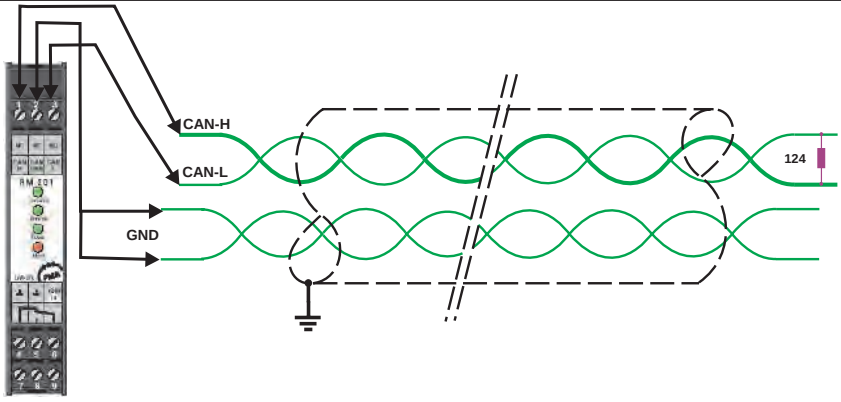
To keep the specified protection degree (IP20) epmty slots must be protected by slot covers RM 214.

The screw-/plug-in-terminals can be plugged in from above or below into the module housing (audible locking). Removing the scre-/plug-in-terminals takes place by levering out, e.g. With a screwdriver.
Due to contact-voltage proof not connected terminals should remain in the respective place.

3.2 *Interface connection*

The CANopen bus is physically connected via screw-/plug-in terminals.

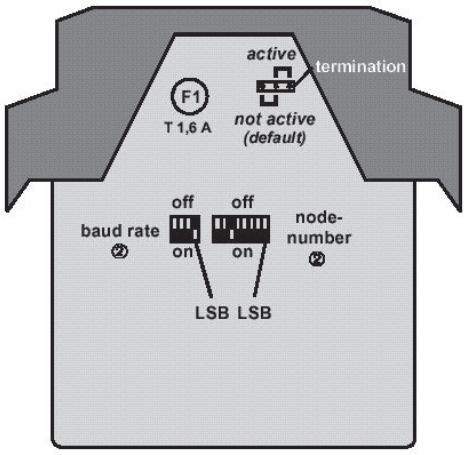
Fig.: 1 Connection CANopen



The construction of suitable cabling must be provided by the user, whereby the general cable specifications must be taken into account.

3.3 *Address settings*

The CANopen-address has to be set on the bus coupler RM 201 via DIP-switches.



DIP switches / Jumper

4 Bit DIP switch

DIP ①	Baud rate
0000	10 kBit
0001	20 kBit ②
0010	50 kBit
0011	100 kBit
0100	125 kBit
0101	250 kBit
0110	500 kBit
0111	800 kBit
1000	1000 kBit
1001	Auto Scan
4321	Switch-Pos.

8 Bit DIP switch

DIP ①	Node-No.
0000 0000	invalid
0000 0001	1
0000 0010	2
0000 0011	3
...	...
0010 0000	32 ②
...	...
0111 1110	126
0111 1111	127
8765 4321	Switch-Pos.

The positions of the switches are shown in binary-code. The number at the right position corresponds to the LSB (DIP-switch-poistion 1), the number at the left position corresponds to the MSB (DIP-switch-position 8).

3.4 *Installation of cables*

When laying the cables, the general hints for cable installation given by the supplier of the master module must be followed:

- Cable run in buildings (inside and outside cabinets)
- Cable run inside and outside buildings
- Potential compensation
- Cable screening
- Measures against interference voltages
- Stub line length
- Bus termination resistors are contained in RM 201, if required it can be switched on by a jumper.
- Earthing

The cable specifications are:

Transmission rate kBit / s	Bus length m	Cross section mm^2	Resistance mΩ/m
1000	≤30	0,25 … 0,34	<70
800	≤50	0,25 0,34	<70
500	≤100	0,34 … 0,60	<60
250	≤250	0,34 … 0,60	<60
125	≤500	0,50 … 0,60	<40
100	≤800	0,75 … 0,80	<26
50	≤1000	0,75 0,80	<26

The recommended cable type should be a shielded twisted pair cable with two pairs according to ISO 11898.

4 General

Due to the concept of decentral in/output modules with CANopen respective PROFIBUS-DP-connection a high degree of flexibility is provided to the application engineer layouting his concept. The compact and cost-effective modules are to be combined to a device with the optimum number of in/outputs. A subsequent system extension is easily done due to the modular concept. The great variety of digital and analog in/output-modules allows the application of this system in many areas. In addition to the standard modules are special modules available.

This manual describes the modular I/O system with CANopen connection through the coupler RM 201.

The required modules were plugged in one carrier consisting of one bus connection and a housing. At present there are available carriers for 3, 5 and 10 modules. Each system allows up to 16 analog inputs and 16 analog outputs. This means 4 analog input modules and 4 analog output modules with 4 channels per module. Up to 8 analog modules RM 224-0 with two galvanic isolated thermocouple inputs, equivalent to 16 analog inputs, are allowed to plug in. The number of digital in/outputs is not restricted. The fieldbus coupler always takes the position left from the other modules .

Maximal amount of modules:

RM 241, RM 242, RM 243	(dig. In)	: 9
RM 251, RM 252	(dig. Out)	: 9
RM 221-x, RM 222-x, RM 224-1	(ana. In)	: 4
RM 224-0	(ana. In)	: 8
RM 231-x	(ana. Out)	: 4

Example:

1 fieldbus coupler, 3 analog input modules, 4 analog output modules,
1 digital input module, 1 digital output module.

This is a valid configuration, since there are not more than 4 analog input and 4 analog output modules. At any time, free slots may be filled up with digital in/output-modules. The limit of 9 in/output-modules has not been reached.

4.1 *Supported I/O-modules*

The following I/O modules are supported by the coupler RM 201 in the **Version V1.25 :**

RM 241	4 x dig. In	3 pole sensor (NPN / PNP)
RM 242	8 x dig. In	potential-bounded 24 V/DC signals
RM 243	4 x dig. In	230 V/AC signals
RM 251	8 x dig. Out	24 V / 1,5 A per output
RM 252	4 x dig. Out	4 x change-over-contact- relays
RM 221-x	4 x ana. In	standard signals / with galvanic isolation between modules
RM 222-x	4 x ana. In	standard signals / with sensor supply
RM 224-1	4 x Temp. In	RTD (Pt100) & TC / full range
RM 224-0	2 x TC. In	TC / full range / galvanic isolation
RM 231-x	4 x ana. Out	standard signals

The specified I/O modules can be combined according to the following design rules:

- there are available basic housing for 3, 5 and 10 modules.
- max. 16 analog inputs are supported.
- max. 16 analog outputs are supported.
- max. 72 digital in- or outputs per unit
- the CANopen coupler has to be placed always in the utter left slot of the housing.

5 Commissioning

5.1 *DIP-Switch-Settings*

The fieldbus coupler RM 201 can be adjusted to the preferred node number and baud rate via DIP-switches

4 Bit DIP-Switch (Baud Rate Selection)

switch position (*)	baud rate
0000	10 kBit
0001	20 kBit = default setting
0010	50 kBit
0011	100 kBit
0100	125 kBit
0101	250 kBit
0110	500 kBit
0111	800 kBit
1000	1000 kBit
1001 ... 1111	invalid

8 Bit DIP-Switch (Node Number Selection)

switch position (*)	node number
0000 0000	invalid
0000 0001	1
0000 0010	2
0000 0011	3
....	....
0010 0000	32 = default setting
....	....
0111 1110	126
0111 1111	127

(*) The switch position is given in binary format, the figure at the right end represents the LSB (DIP-switch-position 1), the figure at the left end represents the MSB (DIP-switch-position 4 for a 4digit switch respective DIP-switch-position 8 for a 8digit switch).

☞ In order to get the optimal benefits of the automatic default-mapping of the modular I/O system a node number smaller than 42 should be selected.

8 Bit DIP-Switch (Service Settings)

switch position (*)	function
1000 0000	invalid
1000 0001	downloading of default settings in EEPROM
1000 0010 … 1000 1111	free

Service-Settings:

The service-settings serve the search and correction of malfunctions. As soon as the diagnostic routine has run, the status will be indicated by the Receive-LED. A fault which cannot be repaired will be indicated by the Alarm-LED and the alarm output. As long as the service-setting is active, the device is unable to operate its normal function (CANopen-Slave-Node). Only after setting a valid baud rate the device will work as usual.

☞ Note:
The read in of the DIP switches status is done once immediately after powering up the device. After changing the DIP switch settings, the device has to be interrupted from the mains to enable the new settings.

Service-Setting 1:

Load EEPROM with default settings.
Some objects are saved nonvolatile in the EEPROM of the fieldbus coupler. So the device can be used after short voltage breakdown with the last settings. The device is delivered with the default settings as described in the object list in the manual.
If the device shows malfunction caused by wrong parameterization via CANopen, the default settings can be restored to the EEPROM with this service routine. The device should operate afterwards as delivered.

Status-Display:

- Five seconds after connection to the mains the yellow Receive-LED should be illuminated. The programming of the EEPROM with default settings is then finished.
- If an error occurs the red ALARM-LED is illuminated and the ALARM-relay pulls in. This indicates an error while writing the default settings to the EEPROM.

Changing the Device Configuration:

A change in the device configuration e.g. by adding a new in/output-module is generally followed by a new programming of the EEPROM of the fieldbus coupler with the default settings. The device operates afterwards as delivered.Via CANopen there is another option to perform a “Reset Node” to reset the device to the default settings.

5.2 *Start-Up-Operation*

Before getting started with the modular I/O system RM 200, the preferred node number and baud rate has to be selected with the DIP-switches of the RM 201 device.
Please note that every node number is to be assigned only once. Assigning the same node number to two devices will result in bus conflicts. Furthermore see to use the same baud rate for all devices of one CAN-network. The modular I/O system RM 200 provides the option to adjust the baud rate automatically at system start. To avoid communication problems mind to terminate the linear bus structure of the CAN-bus with terminal resistors at both ends. The modular I/O system RM 200 provides the option to switch in terminal resistors. Especially at high transmission rates a wrong termination can cause the communication to cease. As a matter of principle the baud rate should be selected as high as necessary and not as high as possible to minimize malfunctions. The following table indicates the maximal network expansion at different given baud rates.

Baud Rate [kBit/s]	max. Net-Extension [m]
500	100
250	250
100	800
50	1000

After switching on an entire unit RM 200 the fieldbus coupler RM 201 begins with the initialization. 5 to 10 seconds later the fieldbus coupler changes into the CANopen state pre-operational. After that the fieldbus coupler generates an emergency message by which any existing error states may be recognized. In this state it is possible to communicate with the device via SDO data transfer. Only after changing in the operational state communication via PDOs is enabled. After transition in the operational state all valid transmit PDOs of the device will be sent immediately once. During the initializing phase the RM 201 should not be reset i.e. reset node and reset communication should be avoided.

5.3 *Object Access via SDOs*

All objects of the modular I/O system RM 200 may be read via SDOs. So-called r/w-objects (read/write) allow in addition to be written via SDOs. To communicate with RM 200 via SDOs the device has be in the CANopen state operational or pre-operational. A SDO consists of 8 usable bytes. It includes the index, subindex, length and value of the object to read or to write.

The modular I/O systems RM 200 operates with an 11 bit identifier according to CAN-specification 2.0A. The following examples are easy to understand with an enhanced CAN-monitor or analyzer.
All examples assume a set node number 2 at the RM 201. So the identifier follows as: 0x602 (0x600 + 2) respectively 0x582 (0x580 + 2). In the examples all data are given in hexadecimal format.

Example 1 (Write 8 Bit Value)

Transmitter	Identifier	1.Byte	2.Byte	3.Byte	4.Byte	5.Byte	6.Byte	7.Byte	8.Byte
PC	602	2F	02	60	01	FF	00	00	00
RM 200	582	60	02	60	01	00	00	00	00

Transmitter: Message-Source

Identifier: Identifier of CAN-Message (here for SDO-Transfers)
PC to RM 200: Identifier = 0x600 + Node-ID
RM 200 to PC: Identifier = 0x580 + Node-ID

1. Byte: Contains informations about the type of data

1. Byte of PC write access
Uint8 / Int8 = 0x2F (write access 8Bit)
Uint16 / Int16 = 0x2B (write access 16Bit)
Uint32 / Int32 = 0x23 (write access 32Bit)
Float = 0x23 (write access 32Bit)

1. Byte of the RM 200 answer
Uint8 / Int8 = 0x60 (acknowledgement 8Bit)
Uint16 / Int16 = 0x60 (acknowledgement 16Bit)
Uint32 / Int32 = 0x60 (acknowledgement 32Bit)
Float = 0x60 (acknowledgement 32Bit)

2. Byte: Index of object, Low-Byte

3. Byte: Index of object, High-Byte

4. Byte: Subindex of object

5.-8. Byte: Usable data of PC write access
8Bit-transmission: 5. Byte = data, 6.,7.,8. Byte = 0x00
16Bit-transmission: 5. Byte = Low-Byte, 6. Byte = High-Byte, 7.,8. Byte = 0x00
32Bit- transmission: 5.,6. Byte = Low-Word, 7.,8. Byte = High-Word

Usable data of the RM 200 answer
At a faultless communication the RM 200 confirms a SDO-write-access by setting all useble data (5. - 8. Byte) to 0x00.

Example 2 (Read 8 Bit Value)

Transmitte	Identifier	1.Byte	2.Byte	3.Byte	4.Byte	5.Byte	6.Byte	7.Byte	8.Byte
PC	602	40	02	60	01	00	00	00	00
RM 200	582	4F	02	60	01	FF	00	00	00

Transmitter: Message-Source

Identifier: Identifier of the CAN-Message (here for SDO-Transfers)
PC to RM 200: Identifier = 0x600 + Node-ID
RM 200 an to: Identifier = 0x580 + Node-ID

1. Byte: Contains informations about the type of data

1. Byte of the PC read access
Uint8 / Int8 = 0x40 (read access)
Uint16 / Int16 = 0x40 (read access)
Uint32 / Int32 = 0x40 (read access)
Float = 0x40 (read access)

1. Byte of the RM 200 answer
Uint8 / Int8 = 0x4F (acknowledgement 8Bit)
Uint16 / Int16 = 0x4B (acknowledgement 16Bit)
Uint32 / Int32 = 0x43 (acknowledgement 32Bit)
Float = 0x43 (acknowledgement 32Bit)

2. Byte: Index of the object, Low-Byte

3. Byte: Index of the object, High-Byte

4. Byte: Subindex of the object

5.-8. Byte: Usable data of the PC request
all usable data Bytes (5.-8. Byte) are set to 0x00.

Usable data of the RM 200 answer
8Bit- transmission: 5. Byte = data, 6.,7.,8. Byte = 0x00
16Bit- transmission: 5. Byte =Low-Byte, 6. Byte = High-Byte, 7.,8. Byte = 0x00
32Bit- transmission: 5.,6. Byte = Low-Word, 7.,8. Byte = High-Word

5.4 *EEPROM-Parameter-Storage*

All relevant parameters of the modular I/O system RM 200 are saved nonvolatile in the EEPROM of the fieldbus coupler RM 201. These are communication parameters as i.e. PDO identifier as well as in/output parameter as e.g. the sensor type.
As soon as an object, which is saved nonvolatile in the EEPROM, gets rewritten, the new value is also stored in the EEPROM. Thanks to this feature it is possible to continue working with the unit as usual even after an interruption from the mains. It is not necessary to start the saving of data in the EEPROM with a command sequence as e.g. 'SAVE' in object 0x1010. In general a device gets parameterized only once. At the start up of the modular I/O system RM 200 the last valid settings will be read out from the EEPROM automatically. By checking the startup message (emergency message after power up) the HMI (Human-Machine-Interface) tests if the device operates accordingly or if e.g. an EEPROM read out error (checksum error) has occurred.

The defaults of the EEPROM data are to be restored at any time. To reset all EEPROM data to their default settings the command 'Reset Node' is used, the command 'Reset Communication' resets only the communication parameter to the default settings. If this command is used one has to consider that the reset of EEPROM data takes a certain amount of time. To assure safe operation one should not communicate with the node for at least 10 seconds.

A change in the device configuration of the modular I/O system RM 200, is followed by a reset of all parameters of the device to the origin. In case of trouble or a defective in/output module the device should only be restarted after replacing the defective in/output module against a new one. If the service technician pulls the defective in/output module and performs a restart without the defect in/output module to test the device, all parameters of the device are set to the default settings.

5.5 *Node-Guarding and Life-Guarding*

The failure checks of a CANopen network are performed with Node-Guarding and Life-Guarding procedures.

Node-Guarding:
With Node-Guarding a NMT master (e.g. the HMI) supervises decentral units at the periphery. With Node-Guarding the HMI recognizes the failure of an individual node.

Life-Guarding:
With Life-Guarding each CANopen node checks if the NMT-Master proceeds the once started Node-Guarding continuously within certain time limits. If the Node-Guarding telegram of the NMT-Masters fails, the decentral CAN unit at the periphery recognizes this with Life-Guarding and sets e.g. all outputs in a safe status.

Function:
With Guarding the NMT-Master as e.g. the HMI (Human-Machine-Interface) transmits remote frames (remote transmit request, message request telegrams) to the guarding-identifier of the slaves which are to be supervised. These respond with the guarding message, which has to contain the CAL-state of the slave and a toggle bit, which has to change with each message. If the status or the toggle bit does not match the masters expectation or if no answer is transmitted, the master assumes a slave failure.

The state transmitted with the guarding telegram can take on these values:

prepared / pre operational	=4
operational	=5
toggle bit	=MSB (Bit 7); Value = 0 at the first guarding telegram

If the master requests the guard message in firm cyclic order, the slave recognizes the correct function of the master. If the slave does not receive a message request from the master within the adjusted life-time (guarding-time-out) he assumes a master failure. The slave sets its outputs on error status and sends an emergency telegram. The emergency telegram is a set of 8 Bytes:
[COB-ID emergency message] with 0x10 | 0x00 | 0x01 | 0x00 | 0x10 | 0x00 | 0x00 | 0x00.
After a guarding-time-out the master can restart the procedure by sending a new guarding telegram.

The life-time is calculated with the objects guard-time (0x100C) and life-time-factor (0x100D). The unit of the life-time and guard time is ms.

life-time = guard-time x life-time-factor

If one of the parameters is zero, no supervising of the master happens (no Life-Guarding).

The guarding-identifier (COB-ID node guarding, object 0x100E) usually results from 0x0700 + Node-ID. With a write access the value of the object 0x100E can be altered according to CANopen.

6 Object directory

6.1 *General*

CANopen equipment communicates using objects. Every object has an index and a sub-index via which the object can be addressed. As part of standardisation, CiA has sub-divided the entire address range into different segments with fixed tasks. In addition to DS301 V3.0, ”CAL based Communication Profile for Industrial Systems” and the objects described there, the modular I/O system with CANopen connections also uses parts of the equipment profile WDP-404-12 ”Measuring Devices and Closed Loop Controllers”. The table below serves as a ”reference” for the object directory entries supported by the device. If required, the texts DS301 and WDP-404 can also be obtained from the CiA.

6.2 *Table of Object-Listing*

Meaning of an individual column:

1. Index — Index of the object, 16 bit, given in hexadecimal format
2. Subindex — Subindex of the object, 8 bit, given in hexadecimal format
3. Designation — Designation of the object = name of the variable
4. Type — Type of variable of the object: i8, i16, i32, ui8, ui16, ui32, float, string
5. PDO — Indicates if an object is able to be mapped in a PDO
6. Default — Value of an object at delivery
7. EEP — Indicates if the variable is saved nonvolatile in the EEPROM

Index	Subindex	Designation	Type	Access	PDO	Default	EEP
0x0002	**0x00**	**Dummy**	**ui8**	**rw**	**yes**	**0**	**no**
0x0003	**0x00**	**Dummy**	**ui16**	**rw**	**yes**	**0**	**no**
0x0004	**0x00**	**Dummy**	**ui32**	**rw**	**yes**	**0**	**no**
0x0005	**0x00**	**Dummy**	**i8**	**rw**	**yes**	**0**	**no**
0x0006	**0x00**	**Dummy**	**i16**	**rw**	**yes**	**0**	**no**
0x0007	**0x00**	**Dummy**	**i32**	**rw**	**yes**	**0**	**no**
0x0008	**0x00**	**Dummy**	**float**	**rw**	**yes**	**0.0**	**no**
0x1000	**0x00**	**Device Type**	**ui32**	**ro**	**no**	**0x000F0194**	**no**
0x1001	**0x00**	**Error Register**	**ui8**	**ro**	**no**	**0**	**no**
0x1003	**-**	**Predefined Error Field**	**-**	**-**	**-**	**-**	**-**
0x1003	0x00	Number of Errors	ui8	ro	no	10	no
0x1003	0x01	Standard Error Field 1	ui32	ro	no	0	no
0x1003	0x02	Standard Error Field 2	ui32	ro	no	0	no
0x1003	0x03	Standard Error Field 3	ui32	ro	no	0	no
0x1003	0x04	Standard Error Field 4	ui32	ro	no	0	no
0x1003	0x05	Standard Error Field 5	ui32	ro	no	0	no
0x1003	0x06	Standard Error Field 6	ui32	ro	no	0	no
0x1003	0x07	Standard Error Field 7	ui32	ro	no	0	no
0x1003	0x08	Standard Error Field 8	ui32	ro	no	0	no
0x1003	0x09	Standard Error Field 9	ui32	ro	no	0	no
0x1003	0x0A	Standard Error Field 10	ui32	ro	no	0	no
0x1004	**-**	**Number of PDOs Supported**	**-**	**-**	**-**	**-**	**-**
0x1004	0x00	Number of PDOs Supported	ui32	ro	no	0x0005000A	no
0x1004	0x01	Number of Sync PDOs	ui32	ro	no	0x0005000A	no
0x1004	0x02	Number of Async PDOs	ui32	ro	no	0x0005000A	no
0x1005	**-**	**COB-ID Sync Message**	**ui32**	**rw**	**no**	**0x00000080**	**yes**
0x1008	**-**	**Device Name**	**string**	**ro**	**no**	**MOD I/O**	**no**
0x1009	**-**	**Hardware-Version**	**string**	**ro**	**no**	**HW-V9821**	**no**
0x100A	**-**	**Software-Version**	**string**	**ro**	**no**	**SW-V01.25**	**no**
0x100B	**-**	**Node-ID**	**ui32**	**ro**	**no**	**<Switch>**	**no**
0x100C	**-**	**Guard-Time**	**ui16**	**rw**	**no**	**1000**	**yes**
0x100D	**-**	**Life-Time-Factor**	**ui8**	**rw**	**no**	**3**	**yes**
0x100E	**-**	COB-ID Node Guarding	**ui32**	**rw**	**no**	**0x700 + ID**	**yes**
0x100F	**-**	**Number of SDOs Supported**	**ui32**	**ro**	**no**	**0x00010001**	**no**
0x1014	**-**	**COB-ID Emergency Message**	**ui32**	**rw**	**no**	**0x80 + ID**	**no**

Index	Subindex	Designation	Type	Access	PDO	Default	EEP
0x1400	-	**Receive PDO1 Parameter**	-	-	-	-	-
0x1400	0x00	Number of Entries	ui8	ro	no	3	no
0x1400	0x01	COB-ID Receive PDO1	ui32	rw	no	0x200 + ID	yes
0x1400	0x02	Transmission-Type Receive PDO1	ui8	rw	no	0xFF	yes
0x1400	0x03	Inhibit Time Receive PDO1	ui16	rw	no	0	yes
0x1401	-	**Receive PDO2 Parameter**	-	-	-	-	-
0x1401	0x00	Number of Entries	ui8	ro	no	3	no
0x1401	0x01	COB-ID Receive PDO2	ui32	rw	no	0x300 + ID	yes
0x1401	0x02	Transmission-Type Receive PDO2	ui8	rw	no	0xFF	yes
0x1401	0x03	Inhibit Time Receive PDO2	ui16	rw	no	0	yes
0x1402	-	**Receive PDO3 Parameter**	-	-	-	-	-
0x1402	0x00	Number of Entries	ui8	ro	no	3	no
0x1402	0x01	COB-ID Receive PDO3	ui32	rw	no	0x22A + ID	yes
0x1402	0x02	Transmission-Type Receive PDO3	ui8	rw	no	0xFF	yes
0x1402	0x03	Inhibit Time Receive PDO3	ui16	rw	no	0	yes
0x1403	-	**Receive PDO4 Parameter**	-	-	-	-	-
0x1403	0x00	Number of Entries	ui8	ro	no	3	no
0x1403	0x01	COB-ID Receive PDO4	ui32	rw	no	0x32A + ID	yes
0x1403	0x02	Transmission-Type Receive PDO4	ui8	rw	no	0xFF	yes
0x1403	0x03	Inhibit Time Receive PDO4	ui16	rw	no	0	yes
0x1404	-	**Receive PDO5 Parameter**	-	-	-	-	-
0x1404	0x00	Number of Entries	ui8	ro	no	3	no
0x1404	0x01	COB-ID Receive PDO5	ui32	rw	no	0x254 + ID	yes
0x1404	0x02	Transmission-Type Receive PDO5	ui8	rw	no	0xFF	yes
0x1404	0x03	Inhibit Time Receive PDO5	ui16	rw	no	0	yes
0x1600	-	**Receive PDO1 Mapping**	-	-	-	-	-
0x1600	0x00	Number of Mapped Objects	ui8	rw	no	0	yes
0x1600	0x01	1. Mapped Object	ui32	rw	no	0x00000000	yes
0x1600	0x02	2. Mapped Object	ui32	rw	no	0x00000000	yes
0x1600	0x03	3. Mapped Object	ui32	rw	no	0x00000000	yes
0x1600	0x04	4. Mapped Object	ui32	rw	no	0x00000000	yes
0x1600	0x05	5. Mapped Object	ui32	rw	no	0x00000000	yes
0x1600	0x06	6. Mapped Object	ui32	rw	no	0x00000000	yes
0x1600	0x07	7. Mapped Object	ui32	rw	no	0x00000000	yes
0x1600	0x08	8. Mapped Object	ui32	rw	no	0x00000000	yes
0x1601	-	**Receive PDO2 Mapping**	-	-	-	-	-
0x1601	0x00	Number of Mapped Objects	ui8	rw	no	0	yes
0x1601	0x01	1. Mapped Object	ui32	rw	no	0x00000000	yes
0x1601	0x02	2. Mapped Object	ui32	rw	no	0x00000000	yes
0x1601	0x03	3. Mapped Object	ui32	rw	no	0x00000000	yes
0x1601	0x04	4. Mapped Object	ui32	rw	no	0x00000000	yes
0x1601	0x05	5. Mapped Object	ui32	rw	no	0x00000000	yes
0x1601	0x06	6. Mapped Object	ui32	rw	no	0x00000000	yes
0x1601	0x07	7. Mapped Object	ui32	rw	no	0x00000000	yes
0x1601	0x08	8. Mapped Object	ui32	rw	no	0x00000000	yes
0x1602	-	**Receive PDO3 Mapping**	-	-	-	-	-
0x1602	0x00	Number of Mapped Objects	ui8	rw	no	0	yes
0x1602	0x01	1. Mapped Object	ui32	rw	no	0x00000000	yes
0x1602	0x02	2. Mapped Object	ui32	rw	no	0x00000000	yes
0x1602	0x03	3. Mapped Object	ui32	rw	no	0x00000000	yes
0x1602	0x04	4. Mapped Object	ui32	rw	no	0x00000000	yes
0x1602	0x05	5. Mapped Object	ui32	rw	no	0x00000000	yes
0x1602	0x06	6. Mapped Object	ui32	rw	no	0x00000000	yes
0x1602	0x07	7. Mapped Object	ui32	rw	no	0x00000000	yes
0x1602	0x08	8. Mapped Object	ui32	rw	no	0x00000000	yes
0x1603	-	**Receive PDO4 Mapping**	-	-	-	-	-
0x1603	0x00	Number of Mapped Objects	ui8	rw	no	0	yes
0x1603	0x01	1. Mapped Object	ui32	rw	no	0x00000000	yes
0x1603	0x02	2. Mapped Object	ui32	rw	no	0x00000000	yes
0x1603	0x03	3. Mapped Object	ui32	rw	no	0x00000000	yes
0x1603	0x04	4. Mapped Object	ui32	rw	no	0x00000000	yes
0x1603	0x05	5. Mapped Object	ui32	rw	no	0x00000000	yes

Index	Subindex	Designation	Type	Access	PDO	Default	EEP
0x1603	0x06	6. Mapped Object	ui32	rw	no	0x00000000	yes
0x1603	0x07	7. Mapped Object	ui32	rw	no	0x00000000	yes
0x1603	0x08	8. Mapped Object	ui32	rw	no	0x00000000	yes
0x1604	-	**Receive PDO5 Mapping**	-	-	-	-	-
0x1604	0x00	Number of Mapped Objects	ui8	rw	no	0	yes
0x1604	0x01	1. Mapped Object	ui32	rw	no	0x00000000	yes
0x1604	0x02	2. Mapped Object	ui32	rw	no	0x00000000	yes
0x1604	0x03	3. Mapped Object	ui32	rw	no	0x00000000	yes
0x1604	0x04	4. Mapped Object	ui32	rw	no	0x00000000	yes
0x1604	0x05	5. Mapped Object	ui32	rw	no	0x00000000	yes
0x1604	0x06	6. Mapped Object	ui32	rw	no	0x00000000	yes
0x1604	0x07	7. Mapped Object	ui32	rw	no	0x00000000	yes
0x1604	0x08	8. Mapped Object	ui32	rw	no	0x00000000	yes
0x1800	-	**Transmit PDO1 Parameter**	-	-	-	-	-
0x1800	0x00	Numer of Entries	ui8	ro	no	3	no
0x1800	0x01	COB-ID Transmit PDO1	ui32	rw	no	0x180 + ID	yes
0x1800	0x02	Transmission-Type Transmit PDO1	ui8	rw	no	0xFF	yes
0x1800	0x03	Inhibit Time Transmit PDO1	ui16	rw	no	0	yes
0x1801	-	**Transmit PDO2 Parameter**	-	-	-	-	-
0x1801	0x00	Numer of Entries	ui8	ro	no	3	no
0x1801	0x01	COB-ID Transmit PDO2	ui32	rw	no	0x280 + ID	yes
0x1801	0x02	Transmission-Type Transmit PDO2	ui8	rw	no	0xFF	yes
0x1801	0x03	Inhibit Time Transmit PDO2	ui16	rw	no	0	yes
0x1802	-	**Transmit PDO3 Parameter**	-	-	-	-	-
0x1802	0x00	Numer of Entries	ui8	ro	no	3	no
0x1802	0x01	COB-ID Transmit PDO3	ui32	rw	no	0x1AA + ID	yes
0x1802	0x02	Transmission-Type Transmit PDO3	ui8	rw	no	0xFF	yes
0x1802	0x03	Inhibit Time Transmit PDO3	ui16	rw	no	0	yes
0x1803	-	**Transmit PDO4 Parameter**	-	-	-	-	-
0x1803	0x00	Numer of Entries	ui8	ro	no	3	no
0x1803	0x01	COB-ID Transmit PDO4	ui32	rw	no	0x2AA + ID	yes
0x1803	0x02	Transmission-Type Transmit PDO4	ui8	rw	no	0xFF	yes
0x1803	0x03	Inhibit Time Transmit PDO4	ui16	rw	no	0	yes
0x1804	-	**Transmit PDO5 Parameter**	-	-	-	-	-
0x1804	0x00	Numer of Entries	ui8	ro	no	3	no
0x1804	0x01	COB-ID Transmit PDO5	ui32	rw	no	0x1D4 + ID	yes
0x1804	0x02	Transmission-Type Transmit PDO5	ui8	rw	no	0xFF	yes
0x1804	0x03	Inhibit Time Transmit PDO5	ui16	rw	no	0	yes
0x1805	-	**Transmit PDO6 Parameter**	-	-	-	-	-
0x1805	0x00	Numer of Entries	ui8	ro	no	3	no
0x1805	0x01	COB-ID Transmit PDO6	ui32	rw	no	0x2D4 + ID	yes
0x1805	0x02	Transmission-Type Transmit PDO6	ui8	rw	no	0xFF	yes
0x1805	0x03	Inhibit Time Transmit PDO6	ui16	rw	no	0	yes
0x1806	-	**Transmit PDO7 Parameter**	-	-	-	-	-
0x1806	0x00	Numer of Entries	ui8	ro	no	3	no
0x1806	0x01	COB-ID Transmit PDO7	ui32	rw	no	0x180+ ID	yes
0x1806	0x02	Transmission-Type Transmit PDO7	ui8	rw	no	0xFF	yes
0x1806	0x03	Inhibit Time Transmit PDO7	ui16	rw	no	0	yes
0x1807	-	**Transmit PDO8 Parameter**	-	-	-	-	-
0x1807	0x00	Numer of Entries	ui8	ro	no	3	no
0x1807	0x01	COB-ID Transmit PDO8	ui32	rw	no	0x180 + ID	yes
0x1807	0x02	Transmission-Type Transmit PDO8	ui8	rw	no	0xFF	yes
0x1807	0x03	Inhibit Time Transmit PDO8	ui16	rw	no	0	yes
0x1808	-	**Transmit PDO9 Parameter**	-	-	-	-	-
0x1808	0x00	Numer of Entries	ui8	ro	no	3	no
0x1808	0x01	COB-ID Transmit PDO9	ui32	rw	no	0x180 + ID	yes
0x1808	0x02	Transmission-Type Transmit PDO9	ui8	rw	no	0xFF	yes
0x1808	0x03	Inhibit Time Transmit PDO9	ui16	rw	no	0	yes
0x1809	-	**Transmit PDO10 Parameter**	-	-	-	-	-
0x1809	0x00	Numer of Entries	ui8	ro	no	3	no
0x1809	0x01	COB-ID Transmit PDO10	ui32	rw	no	0x180 + ID	yes
0x1809	0x02	Transmission-Type Transmit PDO10	ui8	rw	no	0xFF	yes

Index	Subindex	Designation	Type	Access	PDO	Default	EEP
0x1809	0x03	Inhibit Time Transmit PDO10	ui16	rw	no	0	yes
0x1A00	-	**Transmit PDO1 Mapping**	-	-	-	-	-
0x1A00	0x00	Number of Mapped Objects	ui8	rw	no	0	yes
0x1A00	0x01	1. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A00	0x02	2. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A00	0x03	3. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A00	0x04	4. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A00	0x05	5. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A00	0x06	6. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A00	0x07	7. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A00	0x08	8. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A01	-	**Transmit PDO2 Mapping**	-	-	-	-	-
0x1A01	0x00	Number of Mapped Objects	ui8	rw	no	0	yes
0x1A01	0x01	1. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A01	0x02	2. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A01	0x03	3. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A01	0x04	4. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A01	0x05	5. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A01	0x06	6. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A01	0x07	7. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A01	0x08	8. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A02	-	**Transmit PDO3 Mapping**	-	-	-	-	-
0x1A02	0x00	Number of Mapped Objects	ui8	rw	no	0	yes
0x1A02	0x01	1. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A02	0x02	2. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A02	0x03	3. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A02	0x04	4. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A02	0x05	5. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A02	0x06	6. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A02	0x07	7. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A02	0x08	8. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A03	-	**Transmit PDO4 Mapping**	-	-	-	-	-
0x1A03	0x00	Number of Mapped Objects	ui8	rw	no	0	yes
0x1A03	0x01	1. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A03	0x02	2. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A03	0x03	3. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A03	0x04	4. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A03	0x05	5. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A03	0x06	6. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A03	0x07	7. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A03	0x08	8. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A04	-	**Transmit PDO5 Mapping**	-	-	-	-	-
0x1A04	0x00	Number of Mapped Objects	ui8	rw	no	0	yes
0x1A04	0x01	1. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A04	0x02	2. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A04	0x03	3. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A04	0x04	4. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A04	0x05	5. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A04	0x06	6. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A04	0x07	7. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A04	0x08	8. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A05	-	**Transmit PDO6 Mapping**	-	-	-	-	-
0x1A05	0x00	Number of Mapped Objects	ui8	rw	no	0	yes
0x1A05	0x01	1. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A05	0x02	2. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A05	0x03	3. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A05	0x04	4. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A05	0x05	5. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A05	0x06	6. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A05	0x07	7. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A05	0x08	8. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A06	-	**Transmit PDO7 Mapping**	-	-	-	-	-

Index	Subindex	Designation	Type	Access	PDO	Default	EEP
0x1A06	0x00	Number of Mapped Objects	ui8	rw	no	0	yes
0x1A06	0x01	1. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A06	0x02	2. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A06	0x03	3. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A06	0x04	4. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A06	0x05	5. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A06	0x06	6. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A06	0x07	7. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A06	0x08	8. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A07	-	**Transmit PDO8 Mapping**	-	-	-	-	-
0x1A07	0x00	Number of Mapped Objects	ui8	rw	no	0	yes
0x1A07	0x01	1. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A07	0x02	2. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A07	0x03	3. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A07	0x04	4. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A07	0x05	5. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A07	0x06	6. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A07	0x07	7. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A07	0x08	8. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A08	-	**Transmit PDO9 Mapping**	-	-	-	-	-
0x1A08	0x00	Number of Mapped Objects	ui8	rw	no	0	yes
0x1A08	0x01	1. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A08	0x02	2. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A08	0x03	3. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A08	0x04	4. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A08	0x05	5. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A08	0x06	6. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A08	0x07	7. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A08	0x08	8. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A09	-	**Transmit PDO10 Mapping**	-	-	-	-	-
0x1A09	0x00	Number of Mapped Objects	ui8	rw	no	0	yes
0x1A09	0x01	1. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A09	0x02	2. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A09	0x03	3. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A09	0x04	4. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A09	0x05	5. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A09	0x06	6. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A09	0x07	7. Mapped Object	ui32	rw	no	0x00000000	yes
0x1A09	0x08	8. Mapped Object	ui32	rw	no	0x00000000	yes
0x5000	-	**Error_Reset**	**ui16**	**rw**	**yes**	**0x0000**	**no**
0x5001	-	**Alarm_Output**	**ui16**	**rw**	**no**	**0x0000**	**yes**
0x5002	-	**Slot_IDs**	-	-	-	-	-
0x5002	0x00	Number of Entries	ui8	ro	no	9	no
0x5002	0x01	Slot_ID_1	ui8	ro	no	configuration	no
0x5002	0x02	Slot_ID_2	ui8	ro	no	configuration	no
0x5002	0x03	Slot_ID_3	ui8	ro	no	configuration	no
0x5002	0x04	Slot_ID_4	ui8	ro	no	configuration	no
0x5002	0x05	Slot_ID_5	ui8	ro	no	configuration	no
0x5002	0x06	Slot_ID_6	ui8	ro	no	configuration	no
0x5002	0x07	Slot_ID_7	ui8	ro	no	configuration	no
0x5002	0x08	Slot_ID_8	ui8	ro	no	configuration	no
0x5002	0x09	Slot_ID_9	ui8	ro	no	configuration	no
0x6000	-	**DI_Read_State_8_Input_Lines**	-	-	-	-	-
0x6000	0x00	Number of Entries	ui8	ro	no	configuration	no
0x6000	0x01	DI_Read_State_8_Input_Lines_1	ui8	ro	yes	0	no
0x6000	0x02	DI_Read_State_8_Input_Lines_2	ui8	ro	yes	0	no
0x6000	0x03	DI_Read_State_8_Input_Lines_3	ui8	ro	yes	0	no
0x6000	0x04	DI_Read_State_8_Input_Lines_4	ui8	ro	yes	0	no
0x6000	0x05	DI_Read_State_8_Input_Lines_5	ui8	ro	yes	0	no
0x6000	0x06	DI_Read_State_8_Input_Lines_6	ui8	ro	yes	0	no
0x6000	0x07	DI_Read_State_8_Input_Lines_7	ui8	ro	yes	0	no
0x6000	0x08	DI_Read_State_8_Input_Lines_8	ui8	ro	yes	0	no

Index	Subindex	Designation	Type	Access	PDO	Default	EEP
0x6000	0x09	DI_Read_State_8_Input_Lines_9	ui8	ro	yes	0	no
0x6002	-	**DI_Polarity_8_Input_Lines**	-	-	-	-	-
0x6002	0x00	Number of Entries	ui8	ro	no	configuration	no
0x6002	0x01	DI_Polarity_8_Input_Lines_1	ui8	rw	no	0x00	yes
0x6002	0x02	DI_Polarity_8_Input_Lines_2	ui8	rw	no	0x00	yes
0x6002	0x03	DI_Polarity_8_Input_Lines_3	ui8	rw	no	0x00	yes
0x6002	0x04	DI_Polarity_8_Input_Lines_4	ui8	rw	no	0x00	yes
0x6002	0x05	DI_Polarity_8_Input_Lines_5	ui8	rw	no	0x00	yes
0x6002	0x06	DI_Polarity_8_Input_Lines_6	ui8	rw	no	0x00	yes
0x6002	0x07	DI_Polarity_8_Input_Lines_7	ui8	rw	no	0x00	yes
0x6002	0x08	DI_Polarity_8_Input_Lines_8	ui8	rw	no	0x00	yes
0x6002	0x09	DI_Polarity_8_Input_Lines_9	ui8	rw	no	0x00	yes
0x6200	-	**DO_Write_State_8_Output_Lines**	-	-	-	-	-
0x6200	0x00	Number of Entries	ui8	ro	no	configuration	no
0x6200	0x01	DO_Write_State_8_Output_Lines_1	ui8	rw	yes	0	no
0x6200	0x02	DO_Write_State_8_Output_Lines_2	ui8	rw	yes	0	no
0x6200	0x03	DO_Write_State_8_Output_Lines_3	ui8	rw	yes	0	no
0x6200	0x04	DO_Write_State_8_Output_Lines_4	ui8	rw	yes	0	no
0x6200	0x05	DO_Write_State_8_Output_Lines_5	ui8	rw	yes	0	no
0x6200	0x06	DO_Write_State_8_Output_Lines_6	ui8	rw	yes	0	no
0x6200	0x07	DO_Write_State_8_Output_Lines_7	ui8	rw	yes	0	no
0x6200	0x08	DO_Write_State_8_Output_Lines_8	ui8	rw	yes	0	no
0x6200	0x09	DO_Write_State_8_Output_Lines_9	ui8	rw	yes	0	no
0x6202	-	**DO_Polarity_8_Output_Lines**	-	-	-	-	-
0x6202	0x00	Number of Entries	ui8	ro	no	configuration	no
0x6202	0x01	DO_Polarity_8_Output_Lines_1	ui8	rw	no	0x00	yes
0x6202	0x02	DO_Polarity_8_Output_Lines_2	ui8	rw	no	0x00	yes
0x6202	0x03	DO_Polarity_8_Output_Lines_3	ui8	rw	no	0x00	yes
0x6202	0x04	DO_Polarity_8_Output_Lines_4	ui8	rw	no	0x00	yes
0x6202	0x05	DO_Polarity_8_Output_Lines_5	ui8	rw	no	0x00	yes
0x6202	0x06	DO_Polarity_8_Output_Lines_6	ui8	rw	no	0x00	yes
0x6202	0x07	DO_Polarity_8_Output_Lines_7	ui8	rw	no	0x00	yes
0x6202	0x08	DO_Polarity_8_Output_Lines_8	ui8	rw	no	0x00	yes
0x6202	0x09	DO_Polarity_8_Output_Lines_9	ui8	rw	no	0x00	yes
0x6206	-	**DO_Fault_Mode_8_Output_Lines**	-	-	-	-	-
0x6206	0x00	Number of Entries	ui8	ro	no	configuration	no
0x6206	0x01	DO_Fault_Mode_8_Output_Lines_1	ui8	rw	no	0x00	yes
0x6206	0x02	DO_Fault_Mode_8_Output_Lines_2	ui8	rw	no	0x00	yes
0x6206	0x03	DO_Fault_Mode_8_Output_Lines_3	ui8	rw	no	0x00	yes
0x6206	0x04	DO_Fault_Mode_8_Output_Lines_4	ui8	rw	no	0x00	yes
0x6206	0x05	DO_Fault_Mode_8_Output_Lines_5	ui8	rw	no	0x00	yes
0x6206	0x06	DO_Fault_Mode_8_Output_Lines_6	ui8	rw	no	0x00	yes
0x6206	0x07	DO_Fault_Mode_8_Output_Lines_7	ui8	rw	no	0x00	yes
0x6206	0x08	DO_Fault_Mode_8_Output_Lines_8	ui8	rw	no	0x00	yes
0x6206	0x09	DO_Fault_Mode_8_Output_Lines_9	ui8	rw	no	0x00	yes
0x6207	-	**DO_Fault_State_8_Output_Lines**	-	-	-	-	-
0x6207	0x00	Number of Entries	ui8	ro	no	configuration	no
0x6207	0x01	DO_Fault_State_8_Output_Lines_1	ui8	rw	no	0x00	yes
0x6207	0x02	DO_Fault_State_8_Output_Lines_2	ui8	rw	no	0x00	yes
0x6207	0x03	DO_Fault_State_8_Output_Lines_3	ui8	rw	no	0x00	yes
0x6207	0x04	DO_Fault_State_8_Output_Lines_4	ui8	rw	no	0x00	yes
0x6207	0x05	DO_Fault_State_8_Output_Lines_5	ui8	rw	no	0x00	yes
0x6207	0x06	DO_Fault_State_8_Output_Lines_6	ui8	rw	no	0x00	yes
0x6207	0x07	DO_Fault_State_8_Output_Lines_7	ui8	rw	no	0x00	yes
0x6207	0x08	DO_Fault_State_8_Output_Lines_8	ui8	rw	no	0x00	yes
0x6207	0x09	DO_Fault_State_8_Output_Lines_9	ui8	rw	no	0x00	yes
0x5200	-	**DO_Status_8_Output_Lines**	-	-	-	-	-
0x5200	0x00	Number of Entries	ui8	ro	no	configuration	no
0x5200	0x01	DO_Status_ 8_Output_Lines_1	ui8	ro	yes	0x00	no
0x5200	0x02	DO_Status_ 8_Output_Lines _2	ui8	ro	yes	0x00	no
0x5200	0x03	DO_Status_ 8_Output_Lines _3	ui8	ro	yes	0x00	no
0x5200	0x04	DO_Status_ 8_Output_Lines _4	ui8	ro	yes	0x00	no

Index	Subindex	Designation	Type	Access	PDO	Default	EEP
0x5200	0x05	DO_Status_8_Output_Lines_5	ui8	ro	yes	0x00	no
0x5200	0x06	DO_Status_8_Output_Lines_6	ui8	ro	yes	0x00	no
0x5200	0x07	DO_Status_8_Output_Lines_7	ui8	ro	yes	0x00	no
0x5200	0x08	DO_Status_8_Output_Lines_8	ui8	ro	yes	0x00	no
0x5200	0x09	DO_Status_8_Output_Lines_9	ui8	ro	yes	0x00	no
0x5201	-	**DO_Error_Mask_8_Output_Lines**	-	-	-	-	-
0x5201	0x00	Number of Entries	ui8	ro	no	configuration	no
0x5201	0x01	DO_Error_Mask_8_Output_Lines_1	ui8	rw	no	0x0F	yes
0x5201	0x02	DO_Error_Mask_8_Output_Lines_2	ui8	rw	no	0x0F	yes
0x5201	0x03	DO_Error_Mask_8_Output_Lines_3	ui8	rw	no	0x0F	yes
0x5201	0x04	DO_Error_Mask_8_Output_Lines_4	ui8	rw	no	0x0F	yes
0x5201	0x05	DO_Error_Mask_8_Output_Lines_5	ui8	rw	no	0x0F	yes
0x5201	0x06	DO_Error_Mask_8_Output_Lines_6	ui8	rw	no	0x0F	yes
0x5201	0x07	DO_Error_Mask_8_Output_Lines_7	ui8	rw	no	0x0F	yes
0x5201	0x08	DO_Error_Mask_8_Output_Lines_8	ui8	rw	no	0x0F	yes
0x5201	0x09	DO_Error_Mask_8_Output_Lines_9	ui8	rw	no	0x0F	yes
0x5202	-	**DO_Module_Error**	**ui16**	**ro**	**yes**	-	**no**
0x6100	-	**AI_Input_Field_Value**	-	-	-	-	-
0x6100	0x00	Number of Entries	ui8	ro	no	configuration	no
0x6100	0x01	AI_Input_Field_Value_1	ui16	ro	yes	0x00	no
0x6100	0x02	AI_Input_Field_Value_2	ui16	ro	yes	0x00	no
0x6100	0x03	AI_Input_Field_Value_3	ui16	ro	yes	0x00	no
0x6100	0x04	AI_Input_Field_Value_4	ui16	ro	yes	0x00	no
0x6100	0x05	AI_Input_Field_Value_5	ui16	ro	yes	0x00	no
0x6100	0x06	AI_Input_Field_Value_6	ui16	ro	yes	0x00	no
0x6100	0x07	AI_Input_Field_Value_7	ui16	ro	yes	0x00	no
0x6100	0x08	AI_Input_Field_Value_8	ui16	ro	yes	0x00	no
0x6100	0x09	AI_Input_Field_Value_9	ui16	ro	yes	0x00	no
0x6100	0x0A	AI_Input_Field_Value_10	ui16	ro	yes	0x00	no
0x6100	0x0B	AI_Input_Field_Value_11	ui16	ro	yes	0x00	no
0x6100	0x0C	AI_Input_Field_Value_12	ui16	ro	yes	0x00	no
0x6100	0x0D	AI_Input_Field_Value_13	ui16	ro	yes	0x00	no
0x6100	0x0E	AI_Input_Field_Value_14	ui16	ro	yes	0x00	no
0x6100	0x0F	AI_Input_Field_Value_15	ui16	ro	yes	0x00	no
0x6100	0x10	AI_Input_Field_Value_16	ui16	ro	yes	0x00	no
0x6110	-	**AI_Sensor_Type**	-	-	-	-	-
0x6110	0x00	Number of Entries	ui8	ro	no	configuration	no
0x6110	0x01	AI_Sensor_Type_1	ui16	rw	no	configuration	yes
0x6110	0x02	AI_Sensor_Type_2	ui16	rw	no	configuration	yes
0x6110	0x03	AI_Sensor_Type_3	ui16	rw	no	configuration	yes
0x6110	0x04	AI_Sensor_Type_4	ui16	rw	no	configuration	yes
0x6110	0x05	AI_Sensor_Type_5	ui16	rw	no	configuration	yes
0x6110	0x06	AI_Sensor_Type_6	ui16	rw	no	configuration	yes
0x6110	0x07	AI_Sensor_Type_7	ui16	rw	no	configuration	yes
0x6110	0x08	AI_Sensor_Type_8	ui16	rw	no	configuration	yes
0x6110	0x09	AI_Sensor_Type_9	ui16	rw	no	configuration	yes
0x6110	0x0A	AI_Sensor_Type_10	ui16	rw	no	configuration	yes
0x6110	0x0B	AI_Sensor_Type_11	ui16	rw	no	configuration	yes
0x6110	0x0C	AI_Sensor_Type_12	ui16	rw	no	configuration	yes
0x6110	0x0D	AI_Sensor_Type_13	ui16	rw	no	configuration	yes
0x6110	0x0E	AI_Sensor_Type_14	ui16	rw	no	configuration	yes
0x6110	0x0F	AI_Sensor_Type_15	ui16	rw	no	configuration	yes
0x6110	0x10	AI_Sensor_Type_16	ui16	rw	no	configuration	yes
0x7130	-	**AI_Input_Process_Value**	-	-	-	-	-
0x7130	0x00	Number of Entries	ui8	ro	no	configuration	no
0x7130	0x01	AI_Input_Process_Value_1	i16	ro	yes	0	no
0x7130	0x02	AI_Input_Process_Value_2	i16	ro	yes	0	no
0x7130	0x03	AI_Input_Process_Value_3	i16	ro	yes	0	no
0x7130	0x04	AI_Input_Process_Value_4	i16	ro	yes	0	no
0x7130	0x05	AI_Input_Process_Value_5	i16	ro	yes	0	no
0x7130	0x06	AI_Input_Process_Value_6	i16	ro	yes	0	no
0x7130	0x07	AI_Input_Process_Value_7	i16	ro	yes	0	no

Index	Subindex	Designation	Type	Access	PDO	Default	EEP
0x7130	0x08	AI_Input_Process_Value_8	i16	ro	yes	0	no
0x7130	0x09	AI_Input_Process_Value_9	i16	ro	yes	0	no
0x7130	0x0A	AI_Input_Process_Value_10	i16	ro	yes	0	no
0x7130	0x0B	AI_Input_Process_Value_11	i16	ro	yes	0	no
0x7130	0x0C	AI_Input_Process_Value_12	i16	ro	yes	0	no
0x7130	0x0D	AI_Input_Process_Value_13	i16	ro	yes	0	no
0x7130	0x0E	AI_Input_Process_Value_14	i16	ro	yes	0	no
0x7130	0x0F	AI_Input_Process_Value_15	i16	ro	yes	0	no
0x7130	0x10	AI_Input_Process_Value_16	i16	ro	yes	0	no
0x6131	-	**AI_Physical_Unit_Process_Value**	-	-	-	-	-
0x6131	0x00	Number of Entries	ui8	ro	no	configuration	no
0x6131	0x01	AI_Physical_Unit_Process_Value_1	ui16	rw	no	configuration	yes
0x6131	0x02	AI_Physical_Unit_Process_Value_2	ui16	rw	no	configuration	yes
0x6131	0x03	AI_Physical_Unit_Process_Value_3	ui16	rw	no	configuration	yes
0x6131	0x04	AI_Physical_Unit_Process_Value_4	ui16	rw	no	configuration	yes
0x6131	0x05	AI_Physical_Unit_Process_Value_5	ui16	rw	no	configuration	yes
0x6131	0x06	AI_Physical_Unit_Process_Value_6	ui16	rw	no	configuration	yes
0x6131	0x07	AI_Physical_Unit_Process_Value_7	ui16	rw	no	configuration	yes
0x6131	0x08	AI_Physical_Unit_Process_Value_8	ui16	rw	no	configuration	yes
0x6131	0x09	AI_Physical_Unit_Process_Value_9	ui16	rw	no	configuration	yes
0x6131	0x0A	AI_Physical_Unit_Process_Value_10	ui16	rw	no	configuration	yes
0x6131	0x0B	AI_Physical_Unit_Process_Value_11	ui16	rw	no	configuration	yes
0x6131	0x0C	AI_Physical_Unit_Process_Value_12	ui16	rw	no	configuration	yes
0x6131	0x0D	AI_Physical_Unit_Process_Value_13	ui16	rw	no	configuration	yes
0x6131	0x0E	AI_Physical_Unit_Process_Value_14	ui16	rw	no	configuration	yes
0x6131	0x0F	AI_Physical_Unit_Process_Value_15	ui16	rw	no	configuration	yes
0x6131	0x10	AI_Physical_Unit_Process_Value_16	ui16	rw	no	configuration	yes
0x7138	-	**AI_Tare_Zero**	-	-	-	-	-
0x7138	0x00	Number of Entries	ui8	ro	no	configuration	no
0x7138	0x01	AI_Tare_Zero_1	i16	rw	no	0	yes
0x7138	0x02	AI_Tare_Zero_2	i16	rw	no	0	yes
0x7138	0x03	AI_Tare_Zero_3	i16	rw	no	0	yes
0x7138	0x04	AI_Tare_Zero_4	i16	rw	no	0	yes
0x7138	0x05	AI_Tare_Zero_5	i16	rw	no	0	yes
0x7138	0x06	AI_Tare_Zero_6	i16	rw	no	0	yes
0x7138	0x07	AI_Tare_Zero_7	i16	rw	no	0	yes
0x7138	0x08	AI_Tare_Zero_8	i16	rw	no	0	yes
0x7138	0x09	AI_Tare_Zero_9	i16	rw	no	0	yes
0x7138	0x0A	AI_Tare_Zero_10	i16	rw	no	0	yes
0x7138	0x0B	AI_Tare_Zero_11	i16	rw	no	0	yes
0x7138	0x0C	AI_Tare_Zero_12	i16	rw	no	0	yes
0x7138	0x0D	AI_Tare_Zero_13	i16	rw	no	0	yes
0x7138	0x0E	AI_Tare_Zero_14	i16	rw	no	0	yes
0x7138	0x0F	AI_Tare_Zero_15	i16	rw	no	0	yes
0x7138	0x10	AI_Tare_Zero_16	i16	rw	no	0	yes
0x7140	-	**AI_Net_Process_Value**	-	-	-	-	-
0x7140	0x00	Number of Entries	ui8	ro	no	configuration	no
0x7140	0x01	AI_Net_Process_Value_1	i16	ro	yes	0	no
0x7140	0x02	AI_Net_Process_Value_2	i16	ro	yes	0	no
0x7140	0x03	AI_Net_Process_Value_3	i16	ro	yes	0	no
0x7140	0x04	AI_Net_Process_Value_4	i16	ro	yes	0	no
0x7140	0x05	AI_Net_Process_Value_5	i16	ro	yes	0	no
0x7140	0x06	AI_Net_Process_Value_6	i16	ro	yes	0	no
0x7140	0x07	AI_Net_Process_Value_7	i16	ro	yes	0	no
0x7140	0x08	AI_Net_Process_Value_8	i16	ro	yes	0	no
0x7140	0x09	AI_Net_Process_Value_9	i16	ro	yes	0	no
0x7140	0x0A	AI_Net_Process_Value_10	i16	ro	yes	0	no
0x7140	0x0B	AI_Net_Process_Value_11	i16	ro	yes	0	no
0x7140	0x0C	AI_Net_Process_Value_12	i16	ro	yes	0	no
0x7140	0x0D	AI_Net_Process_Value_13	i16	ro	yes	0	no
0x7140	0x0E	AI_Net_Process_Value_14	i16	ro	yes	0	no
0x7140	0x0F	AI_Net_Process_Value_15	i16	ro	yes	0	no

Index	Subindex	Designation	Type	Access	PDO	Default	EEP
0x7140	0x10	AI_Net_Process_Value_16	i16	ro	yes	0	no
0x6150	-	**AI_Status**	-	-	-	-	-
0x6150	0x00	Number of Entries	ui8	ro	no	configuration	no
0x6150	0x01	AI_Status_1	ui8	ro	yes	0	no
0x6150	0x02	AI_Status_2	ui8	ro	yes	0	no
0x6150	0x03	AI_Status_3	ui8	ro	yes	0	no
0x6150	0x04	AI_Status_4	ui8	ro	yes	0	no
0x6150	0x05	AI_Status_5	ui8	ro	yes	0	no
0x6150	0x06	AI_Status_6	ui8	ro	yes	0	no
0x6150	0x07	AI_Status_7	ui8	ro	yes	0	no
0x6150	0x08	AI_Status_8	ui8	ro	yes	0	no
0x6150	0x09	AI_Status_9	ui8	ro	yes	0	no
0x6150	0x0A	AI_Status_10	ui8	ro	yes	0	no
0x6150	0x0B	AI_Status_11	ui8	ro	yes	0	no
0x6150	0x0C	AI_Status_12	ui8	ro	yes	0	no
0x6150	0x0D	AI_Status_13	ui8	ro	yes	0	no
0x6150	0x0E	AI_Status_14	ui8	ro	yes	0	no
0x6150	0x0F	AI_Status_15	ui8	ro	yes	0	no
0x6150	0x10	AI_Status_16	ui8	ro	yes	0	no
0x5100	-	**AI_In_Filter**	-	-	-	-	-
0x5100	0x00	Number of Entries	ui8	ro	no	configuration	no
0x5100	0x01	AI_In_Filter_1	ui8	rw	no	51	yes
0x5100	0x02	AI_In_Filter_2	ui8	rw	no	51	yes
0x5100	0x03	AI_In_Filter_3	ui8	rw	no	51	yes
0x5100	0x04	AI_In_Filter_4	ui8	rw	no	51	yes
0x5100	0x05	AI_In_Filter_5	ui8	rw	no	51	yes
0x5100	0x06	AI_In_Filter_6	ui8	rw	no	51	yes
0x5100	0x07	AI_In_Filter_7	ui8	rw	no	51	yes
0x5100	0x08	AI_In_Filter_8	ui8	rw	no	51	yes
0x5100	0x09	AI_In_Filter_9	ui8	rw	no	51	yes
0x5100	0x0A	AI_In_Filter_10	ui8	rw	no	51	yes
0x5100	0x0B	AI_In_Filter_11	ui8	rw	no	51	yes
0x5100	0x0C	AI_In_Filter_12	ui8	rw	no	51	yes
0x5100	0x0D	AI_In_Filter_13	ui8	rw	no	51	yes
0x5100	0x0E	AI_In_Filter_14	ui8	rw	no	51	yes
0x5100	0x0F	AI_In_Filter_15	ui8	rw	no	51	yes
0x5100	0x10	AI_In_Filter_16	ui8	rw	no	51	yes
0x5103	-	**AI_Comp_Pro**	-	-	-	-	-
0x5103	0x00	Number of Entries	ui8	ro	no	configuration	no
0x5103	0x01	AI_Comp_ Pro _1	i16	ro	no	0	no
0x5103	0x02	AI_Comp_ Pro _2	i16	ro	no	0	no
0x5103	0x03	AI_Comp_ Pro _3	i16	ro	no	0	no
0x5103	0x04	AI_Comp_ Pro _4	i16	ro	no	0	no
0x5103	0x05	AI_Comp_ Pro _5	i16	ro	no	0	no
0x5103	0x06	AI_Comp_ Pro _6	i16	ro	no	0	no
0x5103	0x07	AI_Comp_ Pro _7	i16	ro	no	0	no
0x5103	0x08	AI_Comp_ Pro _8	i16	ro	no	0	no
0x5104	-	**AI_Comp_Filter**	-	-	-	-	-
0x5104	0x00	Number of Entries	ui8	ro	no	configuration	no
0x5104	0x01	AI_Comp_ Filter _1	ui8	rw	no	26	yes
0x5104	0x02	AI_Comp_ Filter _2	ui8	rw	no	26	yes
0x5104	0x03	AI_Comp_ Filter _3	ui8	rw	no	26	yes
0x5104	0x04	AI_Comp_ Filter _4	ui8	rw	no	26	yes
0x5104	0x05	AI_Comp_ Filter _5	ui8	rw	no	26	yes
0x5104	0x06	AI_Comp_ Filter _6	ui8	rw	no	26	yes
0x5104	0x07	AI_Comp_ Filter _7	ui8	rw	no	26	yes
0x5104	0x08	AI_Comp_ Filter _8	ui8	rw	no	26	yes
0x5105	-	**AI_Comp_Stat**	-	-	-	-	-
0x5105	0x00	Number of Entries	ui8	ro	no	configuration	no
0x5105	0x01	AI_Comp_ Stat _1	ui8	ro	no	0	no
0x5105	0x02	AI_Comp_ Stat _2	ui8	ro	no	0	no
0x5105	0x03	AI_Comp_ Stat _3	ui8	ro	no	0	no

Index	Subindex	Designation	Type	Access	PDO	Default	EEP
0x5105	0x04	AI_Comp_ Stat _4	ui8	ro	no	0	no
0x5105	0x05	AI_Comp_ Stat _5	ui8	ro	no	0	no
0x5105	0x06	AI_Comp_ Stat _6	ui8	ro	no	0	no
0x5105	0x07	AI_Comp_ Stat _7	ui8	ro	no	0	no
0x5105	0x08	AI_Comp_ Stat _8	ui8	ro	no	0	no
0x5106	**-**	**AI_In_Comp_En**	**-**	**-**	**-**	**-**	**-**
0x5106	0x00	Number of Entries	ui8	ro	no	configuration	no
0x5106	0x01	AI_In_Comp_En_1	ui8	rw	no	1	yes
0x5106	0x02	AI_In_Comp_En_2	ui8	rw	no	1	yes
0x5106	0x03	AI_In_Comp_En_3	ui8	rw	no	1	yes
0x5106	0x04	AI_In_Comp_En_4	ui8	rw	no	1	yes
0x5106	0x05	AI_In_Comp_En_5	ui8	rw	no	1	yes
0x5106	0x06	AI_In_Comp_En_6	ui8	rw	no	1	yes
0x5106	0x07	AI_In_Comp_En_7	ui8	rw	no	1	yes
0x5106	0x08	AI_In_Comp_En_8	ui8	rw	no	1	yes
0x5106	0x09	AI_In_Comp_En_9	ui8	rw	no	1	yes
0x5106	0x0A	AI_In_Comp_En_10	ui8	rw	no	1	yes
0x5106	0x0B	AI_In_Comp_En_11	ui8	rw	no	1	yes
0x5106	0x0C	AI_In_Comp_En_12	ui8	rw	no	1	yes
0x5106	0x0D	AI_In_Comp_En_13	ui8	rw	no	1	yes
0x5106	0x0E	AI_In_Comp_En_14	ui8	rw	no	1	yes
0x5106	0x0F	AI_In_Comp_En_15	ui8	rw	no	1	yes
0x5106	0x10	AI_In_Comp_En_16	ui8	rw	no	1	yes
0x5107	**-**	**AI_Channel_Error**	**ui16**	**ro**	**yes**	**-**	**no**
0x5108	**-**	**AI_Comp_Error**	**ui8**	**ro**	**yes**	**-**	**no**
0x7300	**-**	**AO_Output_Process_Value**	**-**	**-**	**-**	**-**	**-**
0x7300	0x00	Number of Entries	ui8	ro	no	configuration	no
0x7300	0x01	AO_Output_Process_Value_1	i16	rw	yes	0	no
0x7300	0x02	AO_Output_Process_Value_2	i16	rw	yes	0	no
0x7300	0x03	AO_Output_Process_Value_3	i16	rw	yes	0	no
0x7300	0x04	AO_Output_Process_Value_4	i16	rw	yes	0	no
0x7300	0x05	AO_Output_Process_Value_5	i16	rw	yes	0	no
0x7300	0x06	AO_Output_Process_Value_6	i16	rw	yes	0	no
0x7300	0x07	AO_Output_Process_Value_7	i16	rw	yes	0	no
0x7300	0x08	AO_Output_Process_Value_8	i16	rw	yes	0	no
0x7300	0x09	AO_Output_Process_Value_9	i16	rw	yes	0	no
0x7300	0x0A	AO_Output_Process_Value_10	i16	rw	yes	0	no
0x7300	0x0B	AO_Output_Process_Value_11	i16	rw	yes	0	no
0x7300	0x0C	AO_Output_Process_Value_12	i16	rw	yes	0	no
0x7300	0x0D	AO_Output_Process_Value_13	i16	rw	yes	0	no
0x7300	0x0E	AO_Output_Process_Value_14	i16	rw	yes	0	no
0x7300	0x0F	AO_Output_Process_Value_15	i16	rw	yes	0	no
0x7300	0x10	AO_Output_Process_Value_16	i16	rw	yes	0	no
0x6310	**-**	**AO_Output_Type**	**-**	**-**	**-**	**-**	**-**
0x6310	0x00	Number of Entries	ui8	ro	no	configuration	no
0x6310	0x01	AO_Output_Type_1	ui16	rw	no	configuration	yes
0x6310	0x02	AO_Output_Type_2	ui16	rw	no	configuration	yes
0x6310	0x03	AO_Output_Type_3	ui16	rw	no	configuration	yes
0x6310	0x04	AO_Output_Type_4	ui16	rw	no	configuration	yes
0x6310	0x05	AO_Output_Type_5	ui16	rw	no	configuration	yes
0x6310	0x06	AO_Output_Type_6	ui16	rw	no	configuration	yes
0x6310	0x07	AO_Output_Type_7	ui16	rw	no	configuration	yes
0x6310	0x08	AO_Output_Type_8	ui16	rw	no	configuration	yes
0x6310	0x09	AO_Output_Type_9	ui16	rw	no	configuration	yes
0x6310	0x0A	AO_Output_Type_10	ui16	rw	no	configuration	yes
0x6310	0x0B	AO_Output_Type_11	ui16	rw	no	configuration	yes
0x6310	0x0C	AO_Output_Type_12	ui16	rw	no	configuration	yes
0x6310	0x0D	AO_Output_Type_13	ui16	rw	no	configuration	yes
0x6310	0x0E	AO_Output_Type_14	ui16	rw	no	configuration	yes
0x6310	0x0F	AO_Output_Type_15	ui16	rw	no	configuration	yes
0x6310	0x10	AO_Output_Type_16	ui16	rw	no	configuration	yes
0x5300	**-**	**AO_Out_Status**	**-**	**-**	**-**	**-**	**-**

Object directory

Index	Subindex	Designation	Type	Access	PDO	Default	EEP
0x5300	0x00	Number of Entries	ui8	ro	no	configuration	no
0x5300	0x01	AO_Out_Status_1	ui8	ro	yes	0x00	no
0x5300	0x02	AO_Out_Status_2	ui8	ro	yes	0x00	no
0x5300	0x03	AO_Out_Status_3	ui8	ro	yes	0x00	no
0x5300	0x04	AO_Out_Status_4	ui8	ro	yes	0x00	no
0x5300	0x05	AO_Out_Status_5	ui8	ro	yes	0x00	no
0x5300	0x06	AO_Out_Status_6	ui8	ro	yes	0x00	no
0x5300	0x07	AO_Out_Status_7	ui8	ro	yes	0x00	no
0x5300	0x08	AO_Out_Status_8	ui8	ro	yes	0x00	no
0x5300	0x09	AO_Out_Status_9	ui8	ro	yes	0x00	no
0x5300	0x0A	AO_Out_Status_10	ui8	ro	yes	0x00	no
0x5300	0x0B	AO_Out_Status_11	ui8	ro	yes	0x00	no
0x5300	0x0C	AO_Out_Status_12	ui8	ro	yes	0x00	no
0x5300	0x0D	AO_Out_Status_13	ui8	ro	yes	0x00	no
0x5300	0x0E	AO_Out_Status_14	ui8	ro	yes	0x00	no
0x5300	0x0F	AO_Out_Status_15	ui8	ro	yes	0x00	no
0x5300	0x10	AO_Out_Status_16	ui8	ro	yes	0x00	no
0x5302	**-**	**AO_Channel_Error**	**ui16**	**ro**	**yes**	**-**	**no**

7 Description of Individual Objects

7.1 *Structure of Object list according to WDP-404*

Index	Type of Data
5000 ... 5FFF	Manufacturer Specific Range
6000 ... 6FFF	Float, Unsigned Integers
7000 ... 7FFF	Integer 16
8000 ... 8FFF	Integer 24
9000 ... 9FFF	Integer 32

Index	Type of Data
X000 ... X0FF	Digital Input Block
X100 ... X1FF	Analog Input Block
X200 ... X2FF	Digital Output Block
X300 ... X3FF	Analog Output Block
X400 ... X4FF	Controller Block
X500 ... X5FF	Alarm Function Block
X600 ... XEFF	reserved
XF00 ... XFFF	Device Function Block

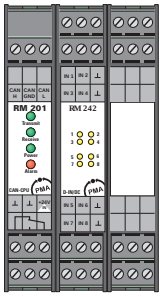
7.2 *General Hints*

The modular I/O system RM 200 can bear up to 10 modules as maximum, that is 1 fieldbus coupler and 9 in/output modules. Per unit up to 4 analog input modules and up to 4 analog output modules with 4 channels each may be plugged in. Limitations are 16 analog inputs and 16 analog outputs. The number of digital in/outputs is not restricted. (see chapter General)

The object list printed in this manual contains for every object the maximum number of all possible subindexes. For the actual application not all subindexes are needed to address the available in/outputs.

The following examples illustrate this situation.

1. Example: 3-fold unit RM 211 with 1 x RM 201 and 1 x RM 242



This minimal application with only one digital input module provides the following objects (index|subindex) for communication purpose:

digital inputs:

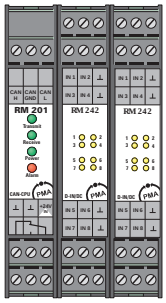
0x6000|0x00 number of digital input modules = 1 (number of subindexes)
0x6000|0x01 ucDI_Input_8Bit[1]

0x6002|0x00 number of digital input modules = 1 (number of subindexes)
0x6002|0x01 ucDI_Polarity_8Bit [1]

All other objects as there are for digital outputs, analog inputs and analog outputs are not available in this configuration.

With the particular number of subindexes one can find out the number of the available digital in/output modules and the corresponding number of in/output-channels.

2. Example: 3-fold unit RM 211 with 1 x RM 201 and 2 x RM 242



This unit with two digital input module provides the following objects (index|subindex) for communication purpose:

digital inputs:

0x6000|0x00 number of digital input modules = 2 (number of subindexes)
0x6000|0x01 ucDI_Input_8Bit[1]
0x6000|0x02 ucDI_Input_8Bit[2]

0x6002|0x00 number of digital input modules = 2 (number of subindexes)
0x6002|0x01 ucDI_Polarity_8Bit [1]
0x6002|0x02 ucDI_Polarity_8Bit [2]

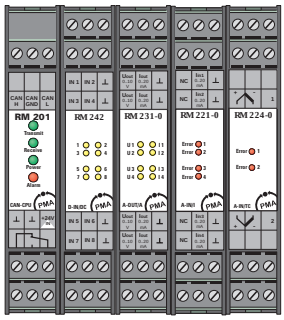
All other objects for digital outputs, analog inputs and analog outputs are not available in this configuration. With the particular number of subindexes one can find out the number of the available digital in/output modules and the corresponding number of in/output-channels.

As a matter of principle for the allocation of modules/channels to the particular subindexes applies the following rule:

The IN/OUTPUT-modules are numbered beginning with the fieldbus coupler from the left to the right. The numbering has to be done separately for the different types of in/output modules digital in, digital out, analog in and analog out.

In this example the first digital input module (directly besides the fieldbus coupler) is addressed with subindex 1 and the second digital input module (at the utter right position in the unit) with subindex 2.

3. Example: 5-fold unit RM 212 with 1 x RM 201, 1 x RM 242, 1 x RM 231-0, 1 x RM 221-0, 1 x RM 224-0



This unit with one digital input module, one analog output module and two analog input modules provides the following objects (index|subindex) for communication purpose:

digital inputs:

0x6000|0x00 number of digital input modules = 1 (number of subindexes)
0x6000|0x01 ucDI_Input_8Bit[1]

0x6002|0x00 number of digital input modules = 1 (number of subindexes)
0x6002|0x01 ucDI_Polarity_8Bit [1]

analog outputs:

0x7300|0x00 number of analog output channels = 4 (number of subindexes)
0x7300|0x01 iAO_Output_Pro[1]
0x7300|0x02 iAO_Output_Pro[2]
0x7300|0x03 iAO_Output_Pro[3]
0x7300|0x04 iAO_Output_Pro[4]

0x6310|0x00 number of analog output channels = 4 (number of subindexes)
0x6310|0x01 uiAO_Output_Type[1]
0x6310|0x02 uiAO_Output_Type[2]
0x6310|0x03 uiAO_Output_Type[3]
0x6310|0x04 uiAO_Output_Type[4]

0x6330|0x00 number of analog output channels = 4 (number of subindexes)

0x6330|0x01	uiAO_Out_Fld[1]
0x6330|0x02	uiAO_Out_Fld[2]
0x6330|0x03	uiAO_Out_Fld[3]
0x6330|0x04	uiAO_Out_Fld[4]

0x5300|0x00	number of analog output channels = 4 (number of subindexes)
0x5300|0x01	ucAO_Out_Status[1]
0x5300|0x02	ucAO_Out_Status[2]
0x5300|0x03	ucAO_Out_Status[3]
0x5300|0x04	ucAO_Out_Status[4]

analog inputs:

0x6100|0x00	number of analog input channels = 6 (number of subindexes)	
0x6100|0x01	uiAI_Input_Fld[1]	(RM 221-0, channel 1, Slot 4
0x6100|0x02	uiAI_Input_Fld[2]	(RM 221-0, channel 2, Slot 4
0x6100|0x03	uiAI_Input_Fld[3]	(RM 221-0, channel 3, Slot 4
0x6100|0x04	uiAI_Input_Fld[4]	(RM 221-0, channel 4, Slot 4
0x6100|0x05	uiAI_Input_Fld[5]	(RM 224-0, channel 1, Slot 5
0x6100|0x06	uiAI_Input_Fld[6]	(RM 224-0, channel 2, Slot 5

0x6110|0x00	number of analog input channels = 6 (number of subindexes)	
0x6110|0x01	uiAI_Sensor_Type[1]	(RM 221-0, channel 1, Slot 4
0x6110|0x02	uiAI_Sensor_Type[2]	(RM 221-0, channel 2, Slot 4
0x6110|0x03	uiAI_Sensor_Type[3]	(RM 221-0, channel 3, Slot 4
0x6110|0x04	uiAI_Sensor_Type[4]	(RM 221-0, channel 4, Slot 4
0x6110|0x05	uiAI_Sensor_Type[5]	(RM 224-0, channel 1, Slot 5
0x6110|0x06	uiAI_Sensor_Type[6]	(RM 224-0, channel 2, Slot 5

…

All other objects for digital outputs and analog inputs are not available in this configuration. With the particular number of subindexes one can find out the number of the available digital in/output modules and the corresponding number of in/output channels.

Attention: In contrast to digital in/outputs analog in/output modules have 4 channels. That's why 4 subindexes per in/output module are needed to address each channel.

With a combination of RM 221-x, RM 222-x, RM 224-1 and RM 224-0 one should bear in mind, that modules RM 224-0 have to be placed right from the modules RM 221-x, RM 222-x repectively RM 224-1. This procedure makes it easier to allocate the analog channels to the particular modules. Please note that the maximal possible number of 16 analog input channels per unit is not exceeded.

If the position of the module RM 221-0 and RM 224-0 are exchanged (slot 4: RM 224-0, slot 5: RM 221-0) then there is no change of the channel sequence. At first the modules with 4 channels are addressed, after that the modules with 2 channels.

7.3 *Digital Inputs*

0x6000 **ucDI_Input_8Bit[9]**

Value	=	state of digital inputs XOR polarity register
Type	=	ui8 / ro
Default	=	none
EEP	=	no
PDO	=	yes, typically mapped

0x6002 **ucDI_Polarity_8Bit[9]**

Value	=	polarity register for interconnection with digital inputs
Type	=	ui8 / rw
Default	=	0x00
EEP	=	yes
PDO	=	no

7.4 *Digital Outputs*

0x6200 **ucDO_Output_8Bit[9]**

Output = value XOR polarity register
Type = ui8 / rw
Default = 0x00
EEP = no storage
PDO = yes, typically mapped

0x6202 **ucDO_Polarity_8Bit[9]**

Value = polarity register for interconnection with digital outputs
Type = ui8 / rw
Default = 0x00
EEP = yes
PDO = no

0x6206 **ucDO_Fault_Mode_8Bit[9]**

Value = Bit set, if the value in ucDO_Fault_State_8Bit[9] shall be given out at a fault condition
The following error-events are possible:

1. The communication via CAN-Bus is disturbed.
   As soon as the CAN Controller changes into the state ‘Bus-Off’ or during the Life-Guarding process a failure is recognized, the value, defined through the objects 0x6206 and 0x6207 is given out. The outputs keep their values until the object 0x6200 or 0x6202 is written with a new value.
2. There is a short-circuit or an open-circuit at at least one digital output and the mask ucDO_Error_Mask allows the failure recognition. All outputs change to the value which is defined by the objects 0x6206 and 0x6207, until the object 0x6200 or 0x6202 is written with a new value.
   The error status can be reset via object 0x5000.

Type = ui8 / rw
Default = 0x00
EEP = yes
PDO = no

0x6207 **ucDO_Fault_State_8Bit[9]**

Value = state of outputs during fault-event, if the particular bit is set in ucDO_Fault_Mode_8Bit[9]. The value is given directly to the output, without interconnection with the polarity register
Type = ui8 / rw
Default = 0x00
EEP = yes
PDO = no

0x5200 ucDO_Status[9]

Value = present status of the digital outputs

meaning of an individual bit

0: short-circuit at channel 1 (1 & 2)
1: short-circuit at channel 2 (3 & 4)
2: short-circuit at channel 3 (5 & 6)
3: short-circuit at channel 4 (7 & 8)
4: open-circuit at channel 1 (1 & 2)
5: open-circuit at channel 2 (3 & 4)
6: open-circuit at channel 3 (5 & 6)
7: open-circuit at channel 4 (7 & 8)

Modules with 4 channels, each channel is allocated to 1 bit.
Modules with 8 channels, two channels are combined to 1 bit.

Type = ui8 / ro
Default = none
EEP = no
PDO = yes

0x5201 ucDO_Error_Mask[9]

Value = bitmask for interconnection with ucDO_status.
With the ucDO_Error_Mask it is determined, if a short-circuit respectively an open-circuit is interpreted as failure.
In case of failure an appropriate emergency message is sent via the CAN-Bus and the outputs are set in dependence of the objects 0x6206 and 0x6207.
The clearing of a bit is recommended e.g. if a not wired output (open-circuit) should not trigger a failure state (default). Typically a short-circuit at the outputs leads to a failure message (bit is set).

meaning of an individual bit:

0: short-circuit at channel 1 (1 & 2)
1: short-circuit at channel 2 (3 & 4)
2: short-circuit at channel 3 (5 & 6)
3: short-circuit at channel 4 (7 & 8)
4: open-circuit at channel 1 (1 & 2)
5: open-circuit at channel 2 (3 & 4)
6: open-circuit at channel 3 (5 & 6)
7: open-circuit at channel 4 (7 & 8)

Modules with 4 channels, each channel is allocated to 1 bit.
Modules with 8 channels, two channels are combined to 1 bit.

Type = ui8 / rw
Default = 0x0F, that means, only short-circuits shall lead to a failure message.
EEP = yes
PDO = no

0x5202 uiDO_Module_Error

Value = If a digital output module has an error, the bit, which is allocated to the particular module in uiDO_Module_Error gets set.
A module is defined as faulty, if at least one bit in ucDO_Status[] of the allocated module is set and the error mask ucDO_Error_Mask[] masks this bit.
Bit = 1, if (ucDO_Status[] & ucDO_Error_Mask[] != 0x00)
Bit = 0, if (ucDO_Status[] & ucDO_Error_Mask[] == 0x00)

meaning of an individual bit:

0: failure in 1. digital output module
1: failure in 2. digital output module
2: failure in 3. digital output module
3: failure in 4. digital output module
4: failure in 5. digital output module
5: failure in 6. digital output module
6: failure in 7. digital output module
7: failure in 8. digital output module
8: failure in 9. digital output module
9: not used, always 0
10: not used, always 0
11: not used, always 0
12: not used, always 0
13: not used, always 0
14: not used, always 0
15: not used, always 0

Type = ui16 / ro
Default = none
EEP = no
PDO = yes

Notes to the digital output module RM 251:

The digital output module RM 251 recognizes open-circuits and short-circuits for two neighbouring outputs each. The following errors can be recognized:

- Not connected output supply and outputs ‘LOW’: Open-circuit
- Not connected output supply and outputs ‘HIGH’: Short-circuit
- Open-circuit at at least one output and outputs ‘LOW’: Open-circuit
- Short-circuit at at least one output and outputs ‘HIGH’: Short circuit

The module RM 251 does not provide greater detail on which one of the two neighbouring channels are faulty. If more precise error localisation is required, an 8-channel digital input module (RM 242) can be used to monitor the outputs. In addition, it is possible to switch two neighbouring channels in parallel in order to be able to evaluate the obtained error messages better.

In order that the error flags which have been set are automatically deleted after the error occurred, the outputs must be reset to the status they were at when the error was recognized. As this is not always possible whilst a process is under way, the error flags of faulty RM 251 modules can be deleted by writing the object 0x5000 (Error_Reset) with the value 0x0002 (digital output module).

The minimum load which does not result in being interpreted as an open-circuit, is usually 50 kOhm (with 24 VDC supply and 25 °C ambient temperature). The status LEDs of the RM 251 indicate a fault by blinking at a steady rate. The object ucDO_Status[9] (0x5299) together with the object ucDO_Error_Mask[9] (0x5201), serves as error information.

7.5 *Analog Inputs*

0x6100 uiAI_Input_Fld[16]

Value = ADC value, unprocessed and not normalized(scaled and formatted)
Type = ui16 / ro
Default = none
EEP = no
PDO = yes

0x6110 uiAI_Sensor_Type[16]

Value = valid values are:

1	(0x01):	TC Type J:	-210.0 °C	...	+1200.0 °C
2	(0x02):	TC Type K:	-270.0 °C	...	+1370.0 °C
3	(0x03):	TC Type L:	-200.0 °C	...	+900.0 °C
4	(0x04):	TC Type E:	-270.0 °C	...	+1000.0 °C
5	(0x05):	TC Type T:	-270.0 °C	...	+400.0 °C
6	(0x06):	TC Type S:	-50.0 °C	...	+1760.0 °C
7	(0x07):	TC Type R:	-50.0 °C	...	+1760.0 °C
8	(0x08):	TC Type B:	+25.0 °C	...	+1820.0 °C
9	(0x09):	TC Type N:	-196.0 °C	...	+1299.6 °C
10	(0x0A):	TC Type W:	0.0 °C	...	+2299.3 °C
30	(0x1E):	RTD(Pt100):	-200.0 °C	...	+850.0 °C
40	(0x28):	0..10 V			
41	(0x29):	-10..+10 V			
51	(0x33):	4..20 mA			
52	(0x34):	0..20 mA			

Bit 13: determines the behaviour at range overflow (e.g. Sensor break for thermocouple)
0: the upper limit value is transmitted (default)
1: the lower limit value is transmitted

Bit 14:
0: interference pulses get suppressed (default)
1: no interference pulse suppression (for high speed signal processing)

Bit 15:
0: channel active (default)
1: channel inactive, process value always 0

Type = ui16 / rw
Default = temperature: 30 (0x1E) = RTD(Pt100) (for RM 224-1)
4 (0x04) = TC type E (for RM 224-0)
voltage: 41 (0x29) = -10 ... +10 V
current: 52 (0x34) = 0 ... 20 mA
EEP = yes
PDO = no

Bits 13 and 15 of the objects uiAI_Sensor_Type[] can be set and cleared independently of the selected type of sensor. It is e.g. possible to deactivate a channel, by interconnecting 0x8000 (Bit 15) with object uiAI_Sensor_Type[] to OR. By clearing of Bit 14 (0x4000) individual interference pulses are suppressed (default). If high speed signals are processed it is recommended to set bit 14, otherwise quick signal changes may be interpreted as failure.

☞ Hints on interference pulse suppression:
An alteration of more than 5 % of the ADC range within 25 ms up to 200 ms (depending on the number and types of analog inputs) is interpreted as an interference pulse. With activated interference pulse suppression a square-wave signal would be recognized and processed but every signal slope would be interpreted as an interference pulse.

0x7130 iAI_Input_Pro[16]

Value	=	process value, processed and normalized (scaled and formatted) physical unit see uiAI_Phy_Unit_Pro[16]
Type	=	i16 / ro
Default	=	none
EEP	=	no
PDO	=	yes, typically mapped

Normalization:
The process value is normalized (scaled and formatted) in different ways according to the measured physical unit. At delivery the following values are valid: the number of decimal places is fixed and can not be altered.
Normierung:

Temperature (unit = °C, 1 decimal place, RTD,Pt100)
-200,0 ... +850 °C = -2000 ... +8500

Voltage (unit = V, 3 decimal places)
0 ... 10,000 V = 0 ... 10000
-10,000 V ... +10,000 V = -10000 ... +10000

Current (unit = mA, 3 decimal places)
0 ... 20 mA = 0 ... 20000
4 ... 20 mA = 0 ... 16000

☞ Hint:
In case of sensor breakage or short-circuit the allocated bit in object 0x6150 ucAI_Status[16] is set. The process value takes on the highest respectively the lowest values in case of failure.

0x6131 uiAI_Phy_Unit_Pro[16]

Value	=	physical unit of the process value extract from the possible units: 0x301*: °C 0x302*: °F 0x303*: K 0x601*: V 0x611*: A * = Factor (least significant 4 Bit) C: 0.000001 (µ) D: 0.001 (m) E: 0.01 (c) F: 0.1 (d) 0: 1 1: 10 (da) 2: 100 (h)
Type	=	ui16 / rw
Default	=	temperature: 0x3010 → factor = 1 [°C] voltage: 0x6010 → factor = 1 [V] current: 0x611D → factor = 0.001 [mA]
EEP	=	yes
PDO	=	no

Beyond the indicated default the following values are also possible:
temperature: 0x3020 → factor = 1 [°F] (see display in Fahrenheit)
0x3030 → factor = 1 [K]

Hint:
be altered to any whatever value. The normalization of the process values is always done as described in 0x7130 iAI_Input_Pro[].

display in Fahrenheit:
The thermocouples of the types S, R, B and W can capture temperatures, which cannot be displayed in Int16-format with the unit 1/10 ° Fahrenheit. That's why the real temperature measured with the types S, R, B and W is displayed reduced by 2000 °F. A real temperature of 2513.4 °F would be transmitted as 5314 ((2513.4 - 2000.0) x 10 = 5314).

0x7138 iAI_Tare_Zero[16]

Value	=	free selectable offset for the calculation of iAI_Net_Pro[16]
Type	=	i16 / rw
Default	=	0
EEP	=	yes
PDO	=	no

0x7140 iAI_Net_Pro[16]

Value	=	iAI_Input_Pro[] - iAI_Tare_Zero[]
Typ	=	i16 / ro
Default	=	none
EEP	=	no
PDO	=	yes

0x6150 ucAI_Status[16]

Value	=	status of the analog inputs

meaning of individual bits:

0:	invalid measuring result, event see bits 1 to 7
1:	overflow of measured value (> highest calibrated value)
2:	underflow of measured value (< lowest calibrated value)
3:	calibration failure (calibration data incorrect)
4:	fault counting limit (to many faults per time unit)
5:	reserved
6:	reserved
7:	reserved

Type	=	ui8 / ro
Default	=	none
EEP	=	no
PDO	=	yes

Hint:
The fault-counting-limit (to many faults per time unit) is only effective, if the interference pulse suppression is activated.

0x5100 ucAI_In_Filter[16]

Value	=	filter constant (FK)
Type	=	ui8 / rw
Default	=	51
EEP	=	yes
PDO	=	no

Averaging:
The measured analog values may processed as sliding average. It applies the following equation:

$\alpha = (FK+1) / 256$
$Y[n+1] = \alpha * X + (1 - \alpha) * Y[n]$

For ucAI_In_Filter[]=255 (means $\alpha = 1$)the analog value is not submitted to averaging.
The maximal averaging is calculated with ucAI_In_Filter[]=0 (means $\alpha = 1/256$).

The cut-off frequency of the low-pass filter of 1. order is calculated with Ta (scanning time) from 25 ms to 200 ms. The exact scanning time depends on the types and numbers of the plugged input modules.

0x5103 iAI_Comp_Pro[8]

Value	=	temperature of the terminals 1/10 °C
Type	=	i16 / ro
Default	=	none
EEP	=	no
PDO	=	no

0x5104 ucAI_Comp_Filter[8]

Value	=	filter constant , see objekt 0x5100
Type	=	ui8 / rw
Default	=	26
EEP	=	yes
PDO	=	none

0x5105 ucAI_Comp_Stat[8]

Value = status of cold junction compensation

meaning of individual bits:

0:	invalid measuring result, event see bits 1 to 7
1:	overflow of measured value (> highest calibrated value)
2:	underflow of measured value (< lowest calibrated value)
3:	calibration failure (calibration data incorrect)
4:	fault counting limit (to many faults per time unit)
5:	communication error
6:	reserved
7:	reserved

Type	=	ui8 / ro
Default	=	none
EEP	=	no
PDO	=	no

0x5106 ucAI_Comp_En[16]

Value	=	activation / deactivation cold junction compensation 0: cold junction compensation deactivated 1: cold junction compensation activated
Type	=	ui8 / rw
Default	=	1 (cold junction compensation active)
EEP	=	yes
PDO	=	no

0x5107 uiAI_Channel_Error

Value = If an analog input channel shows an error, the bit which is allocated to the module is set in uiAI_Channel_Error. A channel is valued as faulty, if the LSB in ucAI_Status[] of the allocated channel is set.

Meaning of individual bits:

0:	failure of 1. analog input channel
1:	failure of 2. analog input channel
2:	failure of 3. analog input channel
3:	failure of 4. analog input channel
4:	failure of 5. analog input channel
5:	failure of 6. analog input channel
6:	failure of 7. analog input channel
7:	failure of 8. analog input channel
8:	failure of 9. analog input channel
9:	failure of 10. analog input channel
10:	failure of 11. analog input channel
11:	failure of 12. analog input channel
12:	failure of 13. analog input channel
13:	failure of 14. analog input channel
14:	failure of 15. analog input channel
15:	failure of 16. analog input channel

Type = ui16 / ro
Default = none
EEP = no
PDO = yes

0x5108 ucAI_Comp_Error

Wert = If the cold junction compensation of a module shows an error, the bit which is allocated to the module is set in ucAI_Comp_Error.
A module is valued as faulty, if the LSB in ucAI_Comp_Stat[] of the allocated module is set.

Meaning of individual bits:

0:	failure of 1. analog input channel
1:	failure of 2. analog input channel
2:	failure of 3. analog input channel
3:	failure of 4. analog input channel
4:	failure of 5. analog input channel
5:	failure of 6. analog input channel
6:	failure of 7. analog input channel
7:	failure of 8. analog input channel

Type = ui8 / ro
Default = none
EEP = no
PDO = yes

7.6 *Analog Outputs*

0x7300 iAO_Output_Pro[16]

Value	=	process value to be displayed, processed and normalized
Type	=	i16 / rw
Default	=	0
EEP	=	no
PDO	=	yes, typically mapped

Normalization:
The process value is normalized (scaled and formatted) in different ways according to the unit to be displayed. At delivery the following values are set

Voltages (unit = V, 3 decimal places)

0 ... 10,000 V	=	0 ... 10000
-10,000 V ... +10,000 V	=	-10000 ... +10000

Currents (unit = mA, 3 decimal places)

0 ... 20 mA	=	0 ... 20000
4 ... 20 mA	=	0 ... 16000

☞ Hint:
If the CAN-controller changes into the bus-off state (e.g. in case of a short-circuit on the CAN-bus) or an Life-Guarding-Time-Out error during the Life-Guarding procedure is detected, all analog outputs are set depending on bit 15 of the value of the output type either to the process value 0 or to the value before the error occurred (See object 0x6310).

0x6310 uiAO_Output_Type[16]

Value = valid values are:

10:	0 ... 10 V
11:	-10 ... +10 V
20:	0 ... 20 mA
21:	4 ... 20 mA

Bit 15 defines the behaviour in case of a bus error:
0: output of process value 0 (default)
1: keep the output value before the error occurred.

Bit 15 of object uiAO_Output_Type[] can be set or reset independent of the selected output format

Type	=	ui16 / rw
Default	=	voltages: 10 = 0 ... 10 V
EEP	=	yes
PDO	=	no

0x5300 ucAO_Out_Status[16]

Value = status of analog outputs
meaning of individual bits
0: invalid measuring result, event see bits 1 to 7
1: calibration failure (calibration data incorrect)
2: reserved
3: failure (failure at data transmission to the DAC)
4: reserved
5: reserved
6: reserved
7: reserved

Type = ui8 / ro
Default = none
EEP = no
PDO = yes

☞ Hint:
All written bits in the DAC(Digital-Analog-Converter) are read back by the micro-controller as routine check. If a deviation is detected (e.g. a bit has toggled) Bit 3 of ucAO_Out_Status[] is set.
Bit 0 is set, as soon as one bit is set between 1 and 7.

0x5302 uiAO_Channel_Error

Value = If an analog output channel shows an error, the bit which is allocated to the module is set in uiAO_Channel_Error. A channel is valued as faulty, if the LSB in ucAO_Out_Status[] of the allocated channel is set

meaning of individual bits:

0: failure of 1. analog output channel
1: failure of 2. analog output channel
2: failure of 3. analog output channel
3: failure of 4. analog output channel
4: failure of 5. analog output channel
5: failure of 6. analog output channel
6: failure of 7. analog output channel
7: failure of 8. analog output channel
8: failure of 9. analog output channel
9: failure of 10. analog output channel
10: failure of 11. analog output channel
11: failure of 12. analog output channel
12: failure of 13. analog output channel
13: failure of 14. analog output channel
14: failure of 15. analog output channel
15: failure of 16. analog output channel

Type = ui16 / ro
Default = none
EEP = no
PDO = yes