МАРАФОН
CAN технология

IO Remote

CANopen I/O module family CiA DS401+

Reference Guide

Firmware version 00090009$_h$

Project codes **004***$_h$

Table of Contents

1. Basic features.

1.1 Specifications.

Number of digital inputs	--, 8, 16
Number of digital counters	--, 8, 16
Number of digital outputs	--, 8, 16
Number of analog inputs	8
Analog input resolution (ADC)	16 bit, 0.01% accuracy
Number of analog outputs	8
Analog output resolution (DAC)	16 bit, 0.1% accuracy
CANopen timer cycle	10 mS (100 Hz)
Indicator (CiA 303 part 3)	Green and Red LEDs
Store parameters in non-volatile memory	On command

1.2 Switch functions.

1.2.1 Protocol selection.

Realized by the slide switch.

Position	Protocol
0 (OFF)	Modbus RTU
1 (ON)	CAN / CANopen

1.2.2 CAN Node-ID selection.

Selected with a couple of 16-position switches (4 LSB and 4 MSB). The Node-ID active value may be stored in non-volatile memory.

Value	CAN node-ID
0	Invalid value, the device does not start
1..127	CAN Node-ID 1 to 127
128..254	Invalid values, the device does not start
255	Non-volatile memory cleaning

1.2.3 CAN bit rate index selection.

Selected with four slide switches: LSB is the first switch, MSB – the fourth one. Switch position OFF is 0 bit value, ON – 1 bit value. The bit rate index active value may be stored in non-volatile memory.

Index	Bit rate
0	1000 kbit/s
1	800 kbit/s
2	500 kbit/s
3	250 kbit/s
4	125 kbit/s
6	50 kbit/s

7	20 kbit/s
8	10 kbit/s
15	Non-volatile memory cleaning
Other values	The device does not start.

1.2.4 Non-volatile memory cleaning.

Non-volatile memory regions, used to store CANopen profile parameters, may be cleaned autonomously. To start the unit in the cleaning mode, CAN Node-ID switch must be set to 255 (FF_h) and CAN bit rate index switch to 15 (F_h). When the unit is in the cleaning mode, both red and green status LEDs are ON. To return to the regular functioning, the required switch values must be set and the device power cycled.

1.3 CANopen protocols and services.

Protocol	Mode	Variants
SDO	server	expedited, segmented.
PDO PDO RTR	producer, consumer	event-triggered, triggered by event-timer, remotely-requested, synchronous (cyclic), synchronous (acyclic).
SYNC	producer, consumer	no SYNC counter: zero bytes SYNC frame data length, SYNC counter: one byte SYNC frame data length.
EMCY	producer	
NMT	consumer	NMT slave.
Error control protocols	producer	boot-up protocol, heartbeat producer, life guarding.

1.4 Electronic Data Sheets.

Unit	EDS file
Digital input: 16 inputs	IOremote_R2DIO_16in_16counters.eds
Digital output: 16 outputs	IOremote_R2DIO_16out.eds
Digital I/O: 8 inputs, 8 outputs	IOremote_R2DIO_8in_8counters_8out.eds
Analog input: 8 inputs, 16-bit	IOremote_R2AIO_16bit_8in.eds
Analog output: 8 outputs, 16-bit	IOremote_R2AIO_16bit_8out.eds

2. Documentation.

2.1 Normative references.

IO Remote module family supports device profile for generic I/O modules CiA DS 401 with manufacturer-specific add-ins.

CiA 301   v. 4.2   CANopen application layer and communication profile.

CiA 303 p. 3 v. 1.4   Indicator specification.

CiA 306   v. 1.3   Electronic data sheet specification.

CiA DS 401   v. 3.0   Device profile for generic I/O modules.

2.2 Abbreviations and definitions.

CiA   CAN in Automation http://www.can-cia.org/

CAN-ID   CAN data-link identifier.

COB-ID   CANopen communication object identifier.

NMT   Network management.

EDS   Electronic data sheet.

DCF   Device configuration file.

PDO   Process data object.

RTR   Remote transmission request.

SDO   Service data object.

M   Mandatory object.

O   Optional object.

LSB   Least significant bit/byte.

MSB   Most significant bit/byte.

RO   Read only access.

WO   Write only access.

RW   Read and write access.

RWR   Read / write on process input (TPDO).

RWW   Read / write on process output (RPDO).

3. Revision history.

Revision number is the part of identity object $1018_h\text{sub}3_h$.

Revision 00090009_h.

The device firmware is based on CiA DS 401 version 3.0.
Emergency code 0080_h added: warning, if the CANopen device transits to NMT state Operational and the analog inputs or counters global interrupt (object 6423_h or 2423_h) is disabled.
Filter mask output (object 6208_h) is excluded from the "saving in non-volatile memory" parameters list.

4. Object dictionary.

The table shows overall layout of the IO Remote module object dictionary, as specified in the device EDS files.

Index (hex)	Sub-index (hex)	Name	Data type	Access	PDO	Store
0002	-	DEFTYPE integer8	integer8	RWR RWW	yes	---
0003	-	DEFTYPE integer16	integer16	RWR RWW	yes	---
0004	-	DEFTYPE integer32	integer32	RWR RWW	yes	---
0005	-	DEFTYPE unsigned8	unsigned8	RWR RWW	yes	---
0006	-	DEFTYPE unsigned16	unsigned16	RWR RWW	yes	---
0007	-	DEFTYPE unsigned32	unsigned32	RWR RWW	yes	---
1000	-	Device type	unsigned32	RO	-	-
1001	-	Error register	unsigned8	RO	yes	-
1002	-	Manufacturer status register	unsigned32	RO	yes	-
1003	---	Pre-defined error field	array	---	---	---
1003	0	number of errors	0 to 8	RW	-	-
1003	1 to 8	Standard error field	unsigned32	RO	-	-
1005	-	COB-ID SYNC message	unsigned32	RW	-	com *
1006	-	Communication cycle period	unsigned32	RW	-	com
1007	-	Synchronous window length	unsigned32	RW	-	com
1008	-	Manufacturer device name	vis-string	RO	-	-
1009	-	Manufacturer hardware version	vis-string	RO	-	-
100A	-	Manufacturer software version	vis-string	RO	-	-
100C	-	Guard time	unsigned16	RW	-	com
100D	-	Life time factor	unsigned8	RW	-	com
1010	---	Store parameters	array	---	---	---
1010	0	highest sub-index supported	6	RO	-	-
1010	1	Save all parameters	unsigned32	RW	-	-
1010	2	Save communication parameters	unsigned32	RW	-	-
1010	3	Save application parameters	unsigned32	RW	-	-
1010	4	---	unsigned32	RW	-	-
1010	5	Save CAN node-ID	unsigned32	RW	-	-
1010	6	Save CAN bit rate index	unsigned32	RW	-	-
1011	---	Restore default parameters	array	---	---	---
1011	0	highest sub-index supported	6	RO	-	-
1011	1	Restore all default parameters	unsigned32	RW	-	-
1011	2	Restore communication default parameters	unsigned32	RW	-	-
1011	3	Restore application default parameters	unsigned32	RW	-	-
1011	4	Restore default parameters for the objects 1005_h, 1014_h, $1400_h sub1_h$, $1401_h sub1_h$, $1402_h sub1_h$, $1403_h sub1_h$, $1800_h sub1_h$, $1801_h sub1_h$, $1802_h sub1_h$, $1803_h sub1_h$	unsigned32	RW	-	-
1011	5	Restore CAN node-ID selection with hardware switch	unsigned32	RW	-	-
1011	6	Restore CAN bit rate index selection with hardware switch	unsigned32	RW	-	-
1014	-	COB-ID EMCY	unsigned32	RW	-	com *
1015	-	Inhibit time EMCY	unsigned16	RW	-	com
1017	-	Producer heartbeat time	unsigned16	RW	-	com
1018	---	Identity object	record	---	---	---
1018	0	highest sub-index supported	4	RO	-	-

1018	1	Vendor-ID	unsigned32	RO	-	-
1018	2	Product code	unsigned32	RO	-	-
1018	3	Revision number	unsigned32	RO	-	-
1018	4	Serial number	unsigned32	RO	-	-
1019	-	Synchronous counter overflow value	unsigned8	RW	-	com
1029	---	Error behavior object	array	---	---	---
1029	0	highest sub-index supported	1	RO	-	-
1029	1	Communication error	unsigned8	RW	-	com
1200	---	SDO server parameter	record	---	---	---
1200	0	highest sub-index supported	2	RO	-	-
1200	1	COB-ID client —> server (rx)	unsigned32	RO	-	-
1200	2	COB-ID server —> client (tx)	unsigned32	RO	-	-
1400	---	RPDO 1 communication parameter	record	---	---	---
1400	0	highest sub-index supported	5	RO	-	-
1400	1	COB-ID of the RPDO 1	unsigned32	RW	-	com *
1400	2	Reception character of the RPDO	unsigned8	RW	-	com
1400	3	Inhibit time; not used for the RPDO	unsigned16	RW	-	-
1400	4	Reserved	unsigned8	RW	-	-
1400	5	Event timer	unsigned16	RW	-	com
1401	---	RPDO 2 communication parameter	record	---	---	---
1401	0	highest sub-index supported	5	RO	-	-
1401	1	COB-ID of the RPDO 2	unsigned32	RW	-	com *
1401	2	Reception character of the RPDO	unsigned8	RW	-	com
1401	3	Inhibit time; not used for the RPDO	unsigned16	RW	-	-
1401	4	Reserved	unsigned8	RW	-	-
1401	5	Event timer	unsigned16	RW	-	com
1402	---	RPDO 3 communication parameter	record	---	---	---
1402	0	highest sub-index supported	5	RO	-	-
1402	1	COB-ID of the RPDO 3	unsigned32	RW	-	com *
1402	2	Reception character of the RPDO	unsigned8	RW	-	com
1402	3	Inhibit time; not used for the RPDO	unsigned16	RW	-	-
1402	4	Reserved	unsigned8	RW	-	-
1402	5	Event timer	unsigned16	RW	-	com
1403	---	RPDO 4 communication parameter	record	---	---	---
1403	0	highest sub-index supported	5	RO	-	-
1403	1	COB-ID of the RPDO 4	unsigned32	RW	-	com *
1403	2	Reception character of the RPDO	unsigned8	RW	-	com
1403	3	Inhibit time; not used for the RPDO	unsigned16	RW	-	-
1403	4	Reserved	unsigned8	RW	-	-
1403	5	Event timer	unsigned16	RW	-	com
1600	---	RPDO 1 mapping parameter	record	---	---	---
1600	0	number of valid object entries	0 to 8	RW	-	com
1600	1 to 8	Mapped application objects	unsigned32	RW	-	com
1601	---	RPDO 2 mapping parameter	record	---	---	---
1601	0	number of valid object entries	0 to 8	RW	-	-
1601	1 to 8	Mapped application objects	unsigned32	RW	-	com
1602	---	RPDO 3 mapping parameter	record	---	---	---
1602	0	number of valid object entries	0 to 8	RW	-	-
1602	1 to 8	Mapped application objects	unsigned32	RW	-	com
1603	---	RPDO 4 mapping parameter	record	---	---	---
1603	0	number of valid object entries	0 to 8	RW	-	-
1603	1 to 8	Mapped application objects	unsigned32	RW	-	com
1800	---	TPDO 1 communication parameter	record	---	---	---
1800	0	highest sub-index supported	6	RO	-	-
1800	1	COB-ID of the TPDO 1	unsigned32	RW	-	com *
1800	2	Transmission character of the TPDO	unsigned8	RW	-	com

1800	3	Inhibit time	unsigned16	RW	-	com
1800	4	Reserved	unsigned8	RW	-	-
1800	5	Event timer	unsigned16	RW	-	com
1800	6	SYNC start value	unsigned8	RW	-	com
1801	---	TPDO 2 communication parameter	record	---	---	---
1801	0	highest sub-index supported	6	RO	-	-
1801	1	COB-ID of the TPDO 2	unsigned32	RW	-	com *
1801	2	Transmission character of the TPDO	unsigned8	RW	-	com
1801	3	Inhibit time	unsigned16	RW	-	com
1801	4	Reserved	unsigned8	RW	-	-
1801	5	Event timer	unsigned16	RW	-	com
1801	6	SYNC start value	unsigned8	RW	-	com
1802	---	TPDO 3 communication parameter	record	---	---	---
1802	0	highest sub-index supported	6	RO	-	-
1802	1	COB-ID of the TPDO 3	unsigned32	RW	-	com *
1802	2	Transmission character of the TPDO	unsigned8	RW	-	com
1802	3	Inhibit time	unsigned16	RW	-	com
1802	4	Reserved	unsigned8	RW	-	-
1802	5	Event timer	unsigned16	RW	-	com
1802	6	SYNC start value	unsigned8	RW	-	com
1803	---	TPDO 4 communication parameter	record	---	---	---
1803	0	highest sub-index supported	6	RO	-	-
1803	1	COB-ID of the TPDO 4	unsigned32	RW	-	com *
1803	2	Transmission character of the TPDO	unsigned8	RW	-	com
1803	3	Inhibit time	unsigned16	RW	-	com
1803	4	Reserved	unsigned8	RW	-	-
1803	5	Event timer	unsigned16	RW	-	com
1803	6	SYNC start value	unsigned8	RW	-	com
1A00	---	TPDO 1 mapping parameter	record	---	---	---
1A00	0	number of valid object entries	0 to 8	RW	-	-
1A00	1 to 8	Mapped application objects	unsigned32	RW	-	com
1A01	---	TPDO 2 mapping parameter	record	---	---	---
1A01	0	number of valid object entries	0 to 8	RW	-	-
1A01	1 to 8	Mapped application objects	unsigned32	RW	-	com
1A02	---	TPDO 3 mapping parameter	record	---	---	---
1A02	0	number of valid object entries	0 to 8	RW	-	-
1A02	1 to 8	Mapped application objects	unsigned32	RW	-	com
1A03	---	TPDO 4 mapping parameter	record	---	---	---
1A03	0	number of valid object entries	0 to 8	RW	-	-
1A03	1 to 8	Mapped application objects	unsigned32	RW	-	com
2000	---	IO Remote hardware configuration	array	---	---	---
2000	0	Number of the configuration parameters	4 to 6	RO	-	-
2000	1	Number of digital inputs 8 bit	unsigned8	RO	-	-
2000	2	Number of digital outputs 8 bit	unsigned8	RO	-	-
2000	3	Number of analog inputs	unsigned8	RO	-	-
2000	4	Number of analog outputs	unsigned8	RO	-	-
2000	5	Analog I/O type index	unsigned8	RO	-	-
2000	6	Analog I/O resolution, bit	unsigned8	RO	-	-
2001	---	Analog inputs or analog outputs range index	array	---	---	---
2001	0	number of analog I/Os	8	RO	-	-
2001	1 to 8	Analog inputs or outputs 1 to 8 range index	unsigned8	RO	-	-
2002	---	Analog inputs rate (SpS) index	array	---	---	---
2002	0	number of analog inputs	8	RO	-	-
2002	1 to 8	Analog inputs 1 to 8 rate index	unsigned8	RO	-	-
2010	---	Status output 8-bit	array	---	---	---
2010	0	number of outputs 8-bit	1 to 2	RO	-	-
2010	1 to 2	Status output, one or two 8-bit	unsigned8	RO	-	-

2011	-	Inhibit time EMCY for digital output lines short circuit	unsigned16	RW	-	com
2012	---	Short circuit emergency mask output 8-bit	array	---	---	---
2012	0	number of outputs 8-bit	1 to 2	RO	-	-
2012	1 to 2	Short circuit emergency mask one or two 8-bit	unsigned8	RW	-	com
2018	-	Analog outputs status	unsigned8	RO	-	-
2019	-	Inhibit time EMCY for the analog outputs short circuit or load dump	unsigned16	RW	-	com
201A	---	Short circuit emergency mask for the analog outputs	unsigned8	RW	-	com
201B	---	Load dump emergency mask for the analog outputs	unsigned8	RW	-	com
2020	---	Mapping of the physical and logical counters	array	---	---	---
2020	0	number of logical counters	6	RO	-	-
2020	1 to 6	Physical counter number	unsigned8	RW	-	-
2021	---	Read logical counter 32-bit	array	---	---	---
2021	0	number of logical counters	6	RO	-	-
2021	1 to 6	Read logical counter 1 to 6	unsigned32	RO	yes	-
2022	-	Physical counters reset mask	unsigned16	WO	yes	-
2030	---	Slew rate update clock index	array	---	---	---
2030	0	number of analog outputs	8	RO	-	-
2030	1 to 8	Analog outputs 1 to 8 slew rate update clock index	unsigned8	RW	-	app
2031	---	Slew rate step size index	array	---	---	---
2031	0	number of analog outputs	8	RO	-	-
2031	1 to 8	Analog outputs 1 to 8 slew rate step size index	unsigned8	RW	-	app
2402	---	Read or set physical counter 32-bit	array	---	---	---
2402	0	number of physical counters	8, 16	RO	-	-
2402	1 to 16	Read or set physical counters 1 to 16	unsigned32	RW	-	-
2421	---	Physical counter interrupt trigger selection	array	---	---	---
2421	0	number of physical counters	8, 16	RO	-	-
2421	1 to 16	Physical counters 1 to 16 interrupt trigger selection	unsigned8	RW	-	app
2422	---	Physical counter interrupt source	array	---	---	---
2422	0	number of interrupt source banks	1	RO	-	-
2422	1	Interrupt source bank 1	unsigned32	RO	-	-
2423	-	Counters global interrupt enable	boolean	RW	-	com
2424	---	Physical counter interrupt upper limit	array	---	---	---
2424	0	number of physical counters	8, 16	RO	-	-
2424	1 to 16	Physical counters 1 to 16 interrupt upper limit	unsigned32	RW	-	app
2426	---	Physical counter interrupt delta	array	---	---	---
2426	0	number of physical counters	8, 16	RO	-	-
2426	1 to 16	Physical counters 1 to 16 interrupt delta	unsigned32	RW	-	app
6000	---	Read input 8-bit	array	---	---	---
6000	0	number of inputs 8-bit	1 to 2	RO	-	-
6000	1 to 2	Read input, one or two 8-bit	unsigned8	RO	yes	-
6002	---	Polarity input 8-bit	array	---	---	---
6002	0	number of inputs 8-bit	1 to 2	RO	-	-
6002	1 to 2	Polarity input, one or two 8-bit	unsigned8	RW	-	app
6005	-	Global interrupt enable digital input	boolean	RW	-	com
6006	---	Interrupt mask any change 8-bit	array	---	---	---
6006	0	number of inputs 8-bit	1 to 2	RO	-	-
6006	1 to 2	Interrupt mask any change, one or two 8-bit	unsigned8	RW	-	app
6007	---	Interrupt mask low-to-high 8-bit	array	---	---	---
6007	0	number of inputs 8-bit	1 to 2	RO	-	-
6007	1 to 2	Interrupt mask low-to-high, one or two 8-bit	unsigned8	RW	-	app
6008	---	Interrupt mask high-to-low 8-bit	array	---	---	---
6008	0	number of inputs 8-bit	1 to 2	RO	-	-
6008	1 to 2	Interrupt mask high-to-low, one or two 8-bit	unsigned8	RW	-	app
6200	---	Write output 8-bit	array	---	---	---
6200	0	number of outputs 8-bit	1 to 2	RO	-	-
6200	1 to 2	Write output, one or two 8-bit	unsigned8	RWW	yes	app
6202	---	Change polarity output 8-bit	array	---	---	---
6202	0	number of outputs 8-bit	1 to 2	RO	-	-
6202	1 to 2	Change polarity output, one or two 8-bit	unsigned8	RW	-	app
6206	---	Error mode output 8-bit	array	---	---	---

6206	0	number of outputs 8-bit	1 to 2	RO	-	-
6206	1 to 2	Error mode output, one or two 8-bit	unsigned8	RW	-	app
6207	---	Error value output 8-bit	array	---	---	---
6207	0	number of outputs 8-bit	1 to 2	RO	-	-
6207	1 to 2	Error value output, one or two 8-bit	unsigned8	RW	-	app
6208	---	Filter mask output 8-bit	array	---	---	---
6208	0	number of outputs 8-bit	1 to 2	RO	-	-
6208	1 to 2	Filter mask output, one or two 8-bit	unsigned8	RW	-	-
6401	---	Read analog input 16-bit	array	---	---	---
6401	0	number of analog inputs	8	RO	-	-
6401	1 to 8	Read analog inputs 1 to 8	integer16	RO	yes	-
6403	---	Read analog input float	array	---	---	---
6403	0	number of analog inputs	8	RO	-	-
6403	1 to 8	Read analog inputs 1 to 8 float	real32	RO	yes	-
6411	---	Write analog output 16-bit	array	---	---	---
6411	0	number of analog outputs	8	RO	-	-
6411	1 to 8	Write analog outputs 1 to 8	integer16	RWW	yes	app
6421	---	Analog input interrupt trigger selection	array	---	---	---
6421	0	number of analog inputs	8	RO	-	-
6421	1 to 8	Analog input interrupt trigger selection 1 to 8	unsigned8	RW	-	app
6422	---	Analog input interrupt source	array	---	---	---
6422	0	number of interrupt source banks	1	RO	-	-
6422	1	Interrupt source bank 1	unsigned32	RO	-	-
6423	-	Analog input global interrupt enable	boolean	RW	-	com
6424	---	Analog input interrupt upper limit integer	array	---	---	---
6424	0	number of analog inputs	8	RO	-	-
6424	1 to 8	Analog inputs 1 to 8 interrupt upper limit integer	integer32	RW	-	app
6425	---	Analog input interrupt lower limit integer	array	---	---	---
6425	0	number of analog inputs	8	RO	-	-
6425	1 to 8	Analog inputs 1 to 8 interrupt lower limit integer	integer32	RW	-	app
6426	---	Analog input interrupt delta unsigned	array	---	---	---
6426	0	number of analog inputs	8	RO	-	-
6426	1 to 8	Analog inputs 1 to 8 interrupt delta unsigned	unsigned32	RW	-	app
642E	---	Analog input offset float	array	---	---	---
642E	0	number of analog inputs	8	RO	-	-
642E	1 to 8	Analog inputs 1 to 8 offset float	real32	RW	-	app
642F	---	Analog input pre-scaling float	array	---	---	---
642F	0	number of analog inputs	8	RO	-	-
642F	1 to 8	Analog inputs 1 to 8 pre-scaling float	real32	RW	-	app
6443	---	Analog output error mode	array	---	---	---
6443	0	number of analog outputs	8	RO	-	-
6443	1 to 8	Analog outputs 1 to 8 error mode	unsigned8	RW	-	app
6444	---	Analog output error value integer	array	---	---	---
6444	0	number of analog outputs	8	RO	-	-
6444	1 to 8	Analog outputs 1 to 8 error value integer	integer32	RW	-	app

Notes.

1. The table lists all objects for IO Remote modules family. The specific type of the device may not implement some of them.
2. Parameters, that may be stored in non-volatile memory are marked with **com** (communication parameters) or **app** (application parameters).
    Asterisk marks communication parameters, that specify pre-defined connection set CAN-IDs, which default values may be restored separately.

5. CiA 301 communication objects.

0002$_h$ to 0007$_h$
Data type definition objects.
Objects 0002$_h$ and 0005$_h$ are of 1 byte size, 0003$_h$ and 0006$_h$ - 2 bytes, 0004$_h$ and 0007$_h$ - 4 bytes size. Any object value may be written successfully without consequences and zero is returned on read access.

1000$_h$
Device type.
Default value: 00%%0191$_h$.

Additional information			General information
Special functionality	M	I/O functionality	Device profile number
00$_h$	0/1	01$_h$ to 0F$_h$	0191$_h$ = 401$_d$
31 24	23	22 16	15 0

I/O functionality, 1 – implemented, 0 – not implemented:
bit 16: digital input(s),
bit 17: digital output(s),
bit 18: analog input(s),
bit 19: analog output(s).

Bit 23, Mapping of PDOs:
0 – pre-defined generic PDO mapping is supported,
1 – device-specific PDO mapping is supported.
Set to 1 for IO Remote digital inputs (device-specific counters PDO mapping).

1001$_h$
Error register.

Bit	M/O	Meaning
0	M	Generic error
1	O	Current
2	O	Voltage
3	O	Temperature
4	O	Communication error
5	O	Device profile specific
6	O	Reserved
7	O	Manufacture-specific

Error register is reset to zero when the device exits error mode (object 1029$_h$) or NMT command Reset Node is processed. Bit 1 (current) is set in accordance with the condition of digital outputs (short circuit) or analog outputs (short circuit or load dump).

1002$_h$
Manufacturer status register.
Default value 00000000$_h$.

1003$_h$
Pre-defined error field.
Stores up to 8 errors at sub-indexes 1 to 8:

Additional information		Error code	
31	16	15	0

1005$_h$
COB-ID SYNC message.
Default value 80$_h$.

X	0/1	0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	11-bit CAN-ID
		1	29-bit CAN-ID	
31	30	29	28-11	10-0

Bit(s)	Value	Description
31	X	do not care
30	0	CANopen device does not generate SYNC message
	1	CANopen device generates SYNC message
29	0	11-bit CAN-ID valid (CAN base frame)
	1	29-bit CAN-ID valid (CAN extended frame)
28 to 0	X	29-bit CAN-ID of the CAN extended frame
10 to 0	X	11-bit CAN-ID of the CAN base frame

An attempt to set bit 29 to 1 is responded with the SDO abort transfer service (abort code 0609 0030$_h$, invalid value for parameter). The first transmission of SYNC message starts within 1 CANopen timer cycle after setting bit 30 to 1. By setting bit 30 to 1 while the synchronous counter overflow value is greater than 0 the first SYNC message starts with the counter reset to 1. It is not allowed to change bits 0 to 28 while the device generates SYNC (bit 30 = 1). An attempt is responded with the SDO abort transfer service (abort code 0601 0000$_h$, unsupported access to an object).

1006$_h$
Communication cycle period.
This period defines the SYNC interval and is given in microseconds.
Default value 0 (object disabled).

CANopen device does not generate SYNC message (bit 30 of 1005$_h$ object is set to 0):
Defines communication cycle period time out. If the SYNC message has not been received within the configured time out, SYNC error is being registered. Setting the value to 0 stops SYNC time out control.

CANopen device generates SYNC message (bit 30 of 1005$_h$ object is set to 1):
If the value is set to 0 the transmission of SYNC messages disabled and SYNC counter value (object 1019$_h$) is reset to 1. By changing the value from 0 the transmission of SYNC messages starts within 1 CANopen timer cycle.

Exact SYNC period value depends on the resolution of the CANopen timer. If the communication cycle period is other then zero, but less then the CANopen timer cycle, SYNC messages are generated with the CANopen timer period. In other cases actual SYNC interval will be integer number of the CANopen timer cycles, not exceeding the communication cycle period.

1007$_h$
Synchronous window length.
Default value 0 (object disabled).

The object indicates the configured time window for synchronous PDOs and is given in microseconds. If the value is set to 0 the synchronous window is disabled. If the value exceeds communication cycle period (object 1006_h) it also does not impact synchronous PDOs processing.

After SYNC message reception the following operations are executed for synchronous PDOs:

1. Writing into object dictionary (activation) of the objects, which values were received during the previous SYNC cycle.
2. Setting of the synchronous TPDOs to send in the CAN network.
3. Reception of the synchronous RPDOs to be activated in the following SYNC cycle.

If the synchronous window length expires all synchronous PDOs are discarded until the next SYNC message is received.

Exact length of the synchronous window depends on the resolution of the CANopen timer. As SYNC object is asynchronous, window length jitter is possible within one CANopen timer cycle.

1008_h
Manufacturer device name.

1009_h
Manufacturer hardware version.

$100A_h$
Manufacturer software version.

$100C_h$
Guard time.
Default value 0 (life guarding disabled).

Guard time value is given in multiple of ms. The guard time multiplied with the life time factor (object $100D_h$) gives the life time for the life guarding protocol. Exact guard time value is integer number of the CANopen timer cycles and is rounded up.

$100D_h$
Life time factor.
Default value 0 (life guarding disabled).

The life time factor multiplied with the guard time (object $100C_h$) gives the life time for the life guarding protocol.

1010_h
Store parameters.

Sub-index 1:
Save all parameters.
Value on read access 00000001_h (CANopen device saves parameters on command).
Stores all parameters, referred by sub-indexes 2, 3, 5, 6.

Sub-index 2:
Save communication parameters.
Value on read access: 00000001_h (CANopen device saves parameters on command).

Sub-index 3:
Save application parameters.
Value on read access: 00000001_h (CANopen device saves parameters on command).

Sub-index 4:

Value on read access: 00000000_h (CANopen device does not save parameters).
On reception of the correct signature in the sub-index, the device responds with the SDO abort transfer service (abort code 0606 0000$_h$, access failed due to a hardware error).

Sub-index 5:
Save CAN node-ID.
Value on read access: 00000001_h (CANopen device saves parameters on command).
Stores active node-ID, then its value will not depend on hardware switch.

Sub-index 6:
Save CAN bit rate index.
Value on read access: 00000001_h (CANopen device saves parameters on command).
Stores active bit rate index, then its value will not depend on hardware switch.

In order to avoid storage of parameters by mistake, storage is only executed when a specific signature ("save") is written to the appropriate sub-index.

e	v	a	s
65_h	76_h	61_h	73_h

MSB   LSB

On reception of the correct signature in the appropriate sub-index the device stores the parameters in non-volatile memory and then confirms the SDO transmission (SDO download initiate response). If the storing failed, the device responds with the SDO abort transfer service (abort code 0606 0000$_h$, access failed due to a hardware error). If a wrong signature is written, the device refuses to store and responds with the SDO abort transfer service (abort code 0800 0020$_h$, data cannot be transferred or stored to the application).

On read access to the appropriate sub-index the CANopen device provides information about its storage functionality with the following format:

Bit(s)	Value	Description
31 to 2	0	Reserved.
1	0	CANopen device does not save parameters autonomously.
	1	CANopen device saves parameters autonomously.
0	0	CANopen device does not save parameters on command.
	1	CANopen device saves parameters on command.

The parameters will be loaded from non-volatile memory when the CANopen device receives NMT command Reset Node, Reset Communication (sub-indexes 2, 4, 5, 6) or power cycled.

After communication parameters are saved, the CANopen device CAN-IDs become static, not dependent on the device node-ID. To revert the dependence, default communication parameters of the pre-defined connection set (object 1011sub4) must be restored.

1011$_h$
Restore default parameters.

Sub-index 1:
Restore all parameters.
Value on read access: 00000001_h (CANopen device restores parameters).

Restores default parameters, referred by sub-indexes 2, 3, 5, 6.

Sub-index 2:
Restore communication default parameters.
Value on read access: 00000001_h (CANopen device restores parameters).

Sub-index 3:
Restore application default parameters.
Value on read access: 00000001_h (CANopen device restores parameters).

Sub-index 4:
Restore default parameters:
1005_h,
1014_h,
1400_hsub1_h, 1401_hsub1_h, 1402_hsub1_h, 1403_hsub1_h,
1800_hsub1_h, 1801_hsub1_h, 1802_hsub1_h, 1803_hsub1_h.
Default values of the parameters specify pre-defined connection set CAN-IDs, taking into account
CANopen device node-ID.
Value on read access: 00000001_h (CANopen device restores parameters).

Sub-index 5:
Restore CAN node-ID selection with hardware switch.
Value on read access: 00000001_h (CANopen device restores parameters).

Sub-index 6:
Restore CAN bit rate index selection with hardware switch.
Value on read access: 00000001_h (CANopen device restores parameters).

In order to avoid the restoring of default parameters by mistake, restoring is only executed when a
specific signature ("load") is written to the appropriate sub-index.

d	a	o	l
64_h	61_h	$6F_h$	$6C_h$
MSB			LSB

On reception of the correct signature in the appropriate sub-index the CANopen device restores the
default parameters and then confirms the SDO transmission (SDO download initiate response). If
the restoring failed, the CANopen device responds with the SDO abort transfer service (abort code
$0606\ 0000_h$, access failed due to a hardware error). If a wrong signature is written, the CANopen
device refuses to restore the defaults and responds with the SDO abort transfer service (abort code
$0800\ 0020_h$, data cannot be transferred or stored to the application).

The default values are set valid after the CANopen device is reset (NMT service reset node,
NMT service reset communication for sub-indexes 2, 4, 5, 6) or power cycled.

On read access to the appropriate sub-index the CANopen device provides information about
its default parameter restoring capability with the following format:

Bit(s)	Value	Description
31 to 1	0	Reserved.
0	0	CANopen device does not restore default parameters.
	1	CANopen device restores parameters.

1014ₕ

COB-ID EMCY.

Default value: 80ₕ + Node-ID.

0/1	0	0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	11-bit CAN-ID
		1	29-bit CAN-ID	
31	30	29	28-11	10-0

Bit(s)	Value	Description
31	0	EMCY exists / is valid
	1	EMCY does not exist / is not valid
30	0	reserved (always 0)
29	0	11-bit CAN-ID valid (CAN base frame)
	1	29-bit CAN-ID valid (CAN extended frame)
28 to 0	X	29-bit CAN-ID of the CAN extended frame
10 to 0	X	11-bit CAN-ID of the CAN base frame

An attempt to set bit 29 to 1 is responded with the SDO abort transfer service (abort code 0609 0030ₕ, invalid value for parameter). It is not allowed to change bits 0 to 28 while the object exists (bit 31 = 1). An attempt is responded with the SDO abort transfer service (abort code 0601 0000ₕ, unsupported access to an object).

1015ₕ

Inhibit time EMCY.

Default value: 0.

The value is given in multiples of 100 mcS. The value 0 disables the inhibit time.

Emergency messages, that occur during the inhibit interval, are not send in the CAN network even after this time. Each emergency event, however, is stored in the error register (object 1001ₕ) and pre-defined error field (object 1003ₕ).

Exact length of the EMCY inhibit time depends on the resolution of the CANopen timer. As EMCY object is asynchronous, inhibit time jitter is possible within one CANopen timer cycle.

1017ₕ

Producer heartbeat time.

Default value: 0.

The value is given in multiples of 1 ms. The value 0 disables the heartbeat.

Exact producer heartbeat time value depends on the resolution of the CANopen timer. If the heartbeat time is other then zero, but less then the CANopen timer cycle, heartbeat messages are generated with the CANopen timer period. In other cases actual heartbeat period will be integer number of the CANopen timer cycles, not exceeding the producer heartbeat time.

1018ₕ

Identity object.

<u>Sub-index 1</u>:

Vendor-ID.

Value: 000000BEₕ (Marathon Ltd).

<u>Sub-index 2</u>:

Product code.

Project code	The device variant
31	16 15	0

Project code	Description
41_h	Digital input, counters and digital output R2DIO
42_h	Analog I/O R2AIO 16-bit

Sub-index 3:
Revision number.

Major revision number	Minor revision number
31 16	15 0

The major revision number identifies a specific CANopen behavior. If the CANopen functionality is different, the major revision number is incremented. The minor revision number identifies different versions of the CANopen device with the same CANopen behavior.
Major and minor revision numbers are set independently. Major revision number increment does not reset minor revision number.

Sub-index 4:
Serial number.

1019_h
Synchronous counter overflow value.
Default value: 0.

This object indicates the configured highest value the synchronous counter supports:

Value	Description
0	The SYNC message shall be transmitted as a CAN message of data length 0.
1	Reserved.
2..240	The SYNC message shall be transmitted as a CAN message of data length 1. The first data byte contains the counter.
241..255	Reserved.

If the value is greater than 1, the SYNC message has a data length of 1 byte. An EMCY message (error code 8240_h, unexpected SYNC data length) is transmitted by a SYNC consumer in the case the configured data length of the SYNC message does not meet the data length of a received SYNC message.
A change of the value is responded with the SDO abort transfer service (abort code 0800 0022_h, data cannot be transferred or stored to the application because of the present device state) in case the sync cycle period (object 1006_h) is unequal to 0.

1029_h
Error behavior object.
Default value: 0.
If a serious CANopen device failure is detected in NMT state Operational, the device enters by default autonomously the NMT state Pre-operational. The CANopen device is configurable to enter alternatively the NMT state Stopped or remain in the current NMT state.

Sub-index 1:
Communication error.
CANopen device failures include the following communication errors:
• Bus-off conditions of the CAN interface.
• Life guarding event with the state 'occurred' and the reason 'time out'.
• CANopen cache overflow (error is not specified by CiA 301).
Error class values:
   0 Change to NMT state Pre-operational (only if currently in NMT state Operational).

1 No change of the NMT state.
2 Change to NMT state Stopped.

If CANopen cache overflow error occurs and the device is in NMT state other then Operational, write access to CAN network is logically disconnected. All pending CAN frames will be discarded. The device is logically reconnected to CAN network when receiving any NMT command.

1200$_h$
SDO server parameter.

Sub-index 1:
COB-ID client —> server (rx).
Value: 600$_h$ + Node-ID.

Sub-index 2:
COB-ID server —> client (tx).
Value: 580$_h$ + Node-ID.

Structure of SDO server COB-ID:

0/1	0	0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	11-bit CAN-ID
		1	29-bit CAN-ID	
31	30	29	28-11	10-0

Bit(s)	Value	Description
31	0	SDO exists / is valid
	1	SDO does not exist / is not valid
30	0	Value is assigned statically
	1	Value is assigned dynamically
29	0	11-bit CAN-ID valid (CAN base frame)
	1	29-bit CAN-ID valid (CAN extended frame)
28 to 0	X	29-bit CAN-ID of the CAN extended frame
10 to 0	X	11-bit CAN-ID of the CAN base frame

An SDO exists only if at both sub-index 1 and sub-index 2 the bit valid (bit 31) is set to 0.
SDO server parameters are static, non-modifiable and not stored in non-volatile memory.

1400$_h$ – 1403$_h$
RPDO communication parameters.
1800$_h$ – 1803$_h$
TPDO communication parameters.

Sub-index 1:
COB-ID of the PDO.
Default values for existing PROs:
1400$_h$ (RPDO 1): 200$_h$ + Node-ID;
1401$_h$ (RPDO 2): 300$_h$ + Node-ID;
1402$_h$ (RPDO 3): 400$_h$ + Node-ID;
1403$_h$ (RPDO 4): 500$_h$ + Node-ID.
1800$_h$ (TPDO 1): 180$_h$ + Node-ID;
1801$_h$ (TPDO 2): 280$_h$ + Node-ID;
1802$_h$ (TPDO 3): 380$_h$ + Node-ID;
1803$_h$ (TPDO 4): 480$_h$ + Node-ID.

0/1	0/1	0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	11-bit CAN-ID
		1	29-bit CAN-ID	
31	30	29	28-11	10-0

Bit(s)	Value	Description
31	0	PDO exists / is valid
	1	PDO does not exist / is not valid
30	0	RTR allowed on this PDO
	1	No RTR allowed on this PDO
29	0	11-bit CAN-ID valid (CAN base frame)
	1	29-bit CAN-ID valid (CAN extended frame)
28 to 0	X	29-bit CAN-ID of the CAN extended frame
10 to 0	X	11-bit CAN-ID of the CAN base frame

The bit valid (bit 31) selects which PDOs are used in the NMT state Operational.
An attempt to set bit 29 to 1 is responded with the SDO abort transfer service (abort code 0609 0030$_h$, invalid value for parameter). It is not allowed to change bits 0 to 28 while the object exists (bit 30 = 1). An attempt is responded with the SDO abort transfer service (abort code 0601 0000$_h$, unsupported access to an object).

Sub-index 2:
Reception / transmission character of the PDO.
Default value: 255.

PDO character	Reception / transmission of the PDO				
	cyclic	a-cyclic	synchronous	event-driven	RTR only
0		X	X		
1-240	X		X		
241-251	reserved				
252			X		X
253				X	X
254				X	
255				X	

Synchronous RPDO (characters 0-240) means that the CANopen device actuates the received data with the reception of the next SYNC. Event-driven means that the RPDO may be received at any time and the data will be actualized immediately.
Synchronous TPDO means that the PDO is transmitted after the SYNC. The device starts sampling of the data with the reception of the SYNC. In case it is acyclic (TPDO character 0) the device internal event is given and with the next SYNC the sampling is started and the PDO is transmitted afterwards. In case it is cyclic (TPDO characters 1-240) the sampling is started with the reception of every SYNC, every 2nd SYNC, every 3rd SYNC and so on, depending on the given PDO character value and the PDO is transmitted afterwards. The TPDO will be re-synchronized after setting the sub-index.
RTR-only (TPDO characters 252, 253) means that the PDO is not transmitted normally and shall be requested via RTR. In case it is synchronous the sampling is started with the next SYNC after RTR and the PDO is transmitted afterwards. In case it is event-driven the device starts sampling with the reception of the RTR and transmits the PDO immediately.
Event-driven means that the PDO may be transmitted at any time based on the occurrence of the device internal event. The definition of the event is specified in the device application profile.
An attempt to change the value of the transmission type to any not supported value is responded with the SDO abort transfer service (abort code 0609 0030$_h$, invalid value for parameter).

Sub-index 3:
Inhibit time.
Default value: 0 (disabled).
The time is the minimum interval for PDO transmission if the transmission type is set to 254 or 255. The value is defined as multiple of 100 mcS. The value of 0 disables the inhibit time. It is not allowed to change the value while the PDO exists (bit 31 of sub-index 1 is set to 0). An attempt is responded with the SDO abort transfer service (abort code 0601 0000$_h$, unsupported access to an object).
Exact length of the TPDO inhibit time depends on the resolution of the CANopen timer. As the TPDO object is asynchronous, inhibit time jitter is possible within one CANopen timer cycle.
For the RPDO any value may be written successfully without consequences and zero is returned on read access.

Sub-index 4:
Reserved.
Any value may be written successfully without consequences and zero is returned on read access.

Sub-index 5:
Event timer.
Default value: 0 (disabled).
The time is the maximum interval for PDO transmission if the transmission type is set to 254 or 255. The value is defined as multiple of 1 ms. The value of 0 disables the event-timer.
Exact length of the event timer period depends on the resolution of the CANopen timer. If the event timer value is other then zero, but less then the CANopen timer cycle, TPDO objects are generated with the CANopen timer period. In other cases actual TPDO transmission interval will be integer number of the CANopen timer cycles, not exceeding the event timer value. As the TPDO object is asynchronous, event time jitter is possible within one CANopen timer cycle.
The RPDO uses the time for deadline monitoring. The deadline monitoring is activated within the next reception of an RPDO after configuring the event-timer. Deadline counter is reset only after all RPDO values have been successfully loaded into application object dictionary.
Exact length of the deadline control period depends on the resolution of the CANopen timer. As the RPDO object is asynchronous, deadline control period jitter is possible within one CANopen timer cycle.

Sub-index 6:
SYNC start value.
Default value: 0.
The object is defined for TPDO only.
The SYNC start value of 0 indicates that the counter of the SYNC message is not processed for this TPDO. The SYNC start value 1 to 240 indicate that the counter of the SYNC message is processed for this TPDO. In case the counter of the SYNC message is not enabled (see object 1019$_h$) the sub-index is ignored. The SYNC message of which the counter value equals the SYNC start value is regarded as the first received SYNC message. The TPDO will be re-synchronized after setting the sub-index. It is not allowed to change the value while the PDO exists (bit 31 of sub-index 1 is set to 0). An attempt is responded with the SDO abort transfer service (abort code 0601 0000$_h$, unsupported access to an object).

1600$_h$ – 1603$_h$
Receive PDO mapping parameter (RPDO1 to RPDO4).
1A00$_h$ – 1A03$_h$
Transmit PDO mapping parameter (TPDO1 to TPDO4).

Sub-index 0 contains the number of valid object entries within the mapping record. The number of valid object entries is the number of the application objects that are received with the corresponding PDO. Up to 8 mapping records are reserved for each PDO.

Sub-index from 1$_h$ to 8$_h$ contains the information of the mapped application objects:

Application object index	Sub-index	Object length (bits)
31 16	15 8	7 0

An attempt to change the value of an object entry to any value that is not supported is responded with the SDO abort transfer service. The cause for a not supported value could be the mapping (index and sub-index) of a non-existing application object, a wrong length for the mapped application object, or a wrong length for the PDO at all. The index and sub-index may reference simple data types (objects 0002$_h$ to 0007$_h$) for the dummy mapping.
The following procedure is used for re-mapping, which may take place during the NMT state Pre-operational and during the NMT state Operational:
1. Destroy PDO by setting bit 31 (valid) to 1 of sub-index 1$_h$ of the according PDO communication parameter.
2. Disable mapping by setting sub-index 0$_h$ to 0.
3. Modify mapping by changing the values of the corresponding sub-indexes.
4. Enable mapping by setting sub-index 0$_h$ to the number of mapped objects.
5. Create PDO by setting bit 31 (valid) to 0 of sub-index 1$_h$ of the according PDO communication parameter.

When mapping is disabled (step 2) the corresponding PDO is destroyed automatically (step 1). At the same time, re-creating of the PDO (step 5) must be executed explicitly.
If during step 3 the device detects that index and sub-index of the mapped object does not exist or the object cannot be mapped, it responds with the SDO abort transfer service (abort code 0604 0041$_h$, object cannot be mapped to the PDO). If during step 4 the device detects that the PDO mapping is not valid or not possible, it responds with the SDO abort transfer service (abort codes 0604 0041$_h$, object cannot be mapped to the PDO or 0604 0042$_h$, the number and length of the objects to be mapped would exceed PDO length).
If the device receives a PDO that is having more data bytes than the number of mapped data bytes, then the device uses the first data bytes up to the mapped data length. If the device receives a PDO that is having less data bytes than the number of mapped data bytes, then the device initiates the EMCY with the error code 8210$_h$ (PDO not processed due to length error).

Default IO Remote PDO mapping.

RPDO1 (digital outputs):

Index (hex)	Sub-index (hex)	Object description	Default value (hex)
1600	0	Number of valid object entries	1 or 2
	1	Digital outputs 1 to 8	6200 01 08
	2	Digital outputs 9 to 16	6200 02 08

TPDO1 (digital inputs):

Index (hex)	Sub-index (hex)	Object description	Default value (hex)
1A00	0	Number of valid object entries	1 or 2
	1	Digital inputs 1 to 8	6000 01 08
	2	Digital inputs 9 to 16	6000 02 08

RPDO2 (digital counters reset):

Index (hex)	Sub-index (hex)	Object description	Default value (hex)
1601	0	Number of valid object entries	1
	1	Digital counters 1 to 16 reset mask	2022 00 10

TPDO2 to TPDO4 (logical counters 32 bit):

Index (hex)	Sub-index (hex)	Object description	Default value (hex)
1A01	0	Number of valid object entries	2
	1	Logical counter 1	2021 01 20
	2	Logical counter 2	2021 02 20
1A02	0	Number of valid object entries	2
	1	Logical counter 3	2021 03 20
	2	Logical counter 4	2021 04 20
1A03	0	Number of valid object entries	2
	1	Logical counter 5	2021 05 20
	2	Logical counter 6	2021 06 20

RPDO2, RPDO3 (analog outputs 16 bit):

Index (hex)	Sub-index (hex)	Object description	Default value (hex)
1601	0	Number of valid object entries	4
	1	Analog output 1	6411 01 10
	2	Analog output 2	6411 02 10
	3	Analog output 3	6411 03 10
	4	Analog output 4	6411 04 10
1602	0	Number of valid object entries	4
	1	Analog output 5	6411 05 10
	2	Analog output 6	6411 06 10
	3	Analog output 7	6411 07 10
	4	Analog output 8	6411 08 10

TPDO2, TPDO3 (analog outputs 16 bit):

Index (hex)	Sub-index (hex)	Object description	Default value (hex)
1A01	0	Number of valid object entries	4
	1	Analog input 1	6401 01 10
	2	Analog input 2	6401 02 10
	3	Analog input 3	6401 03 10
	4	Analog input 4	6401 04 10

1A02	0	Number of valid object entries	4
	1	Analog input 5	6401 05 10
	2	Analog input 6	6401 06 10
	3	Analog input 7	6401 07 10
	4	Analog input 8	6401 08 10

The number of mapped application objects depends on the IO Remote device type.

6. IO Remote device profile.

6.1 Device configuration objects.

2000$_h$
IO Remote hardware configuration.

Sub-index 1:
Number of digital inputs 8 bit: 0 to 2.

Sub-index 2:
Number of digital outputs 8 bit: 0 to 2.

Sub-index 3:
Number of analog inputs: 0 or 8.

Sub-index 4:
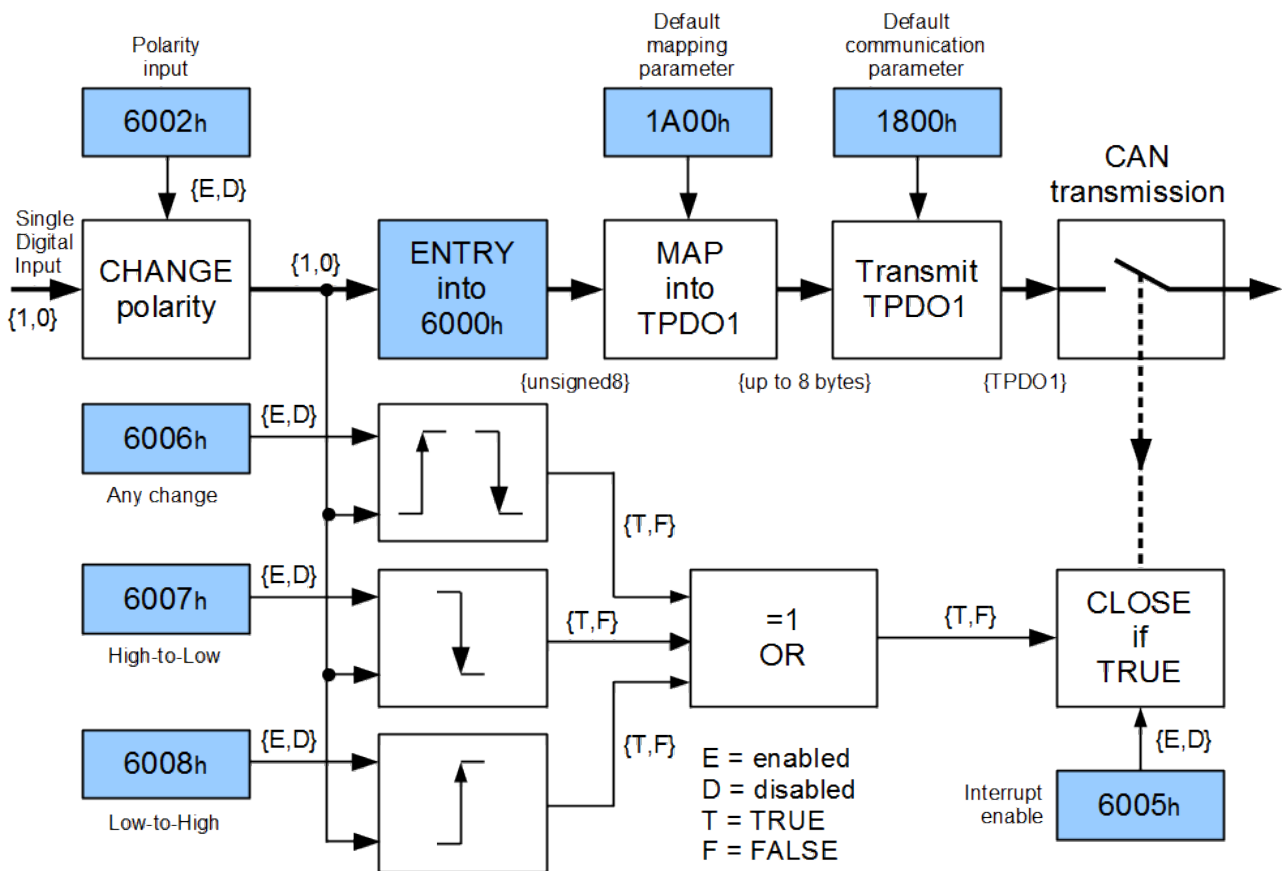Number of analog outputs: 0 or 8.

Sub-index 5, for analog I/O only:
Analog I/O type index.

Sub-index 6, for analog I/O only:
Analog I/O resolution, bit.

6.2 Digital input module.

6.2.1 Standardized CiA 401 objects.

6000$_h$
Read input 8-bit.
This object reads groups of one or two 8 input lines as 8-bit information.

6002$_h$
Polarity input 8-bit.
Default value: 0$_h$.
This object defines the polarity of a group of 8 input lines. Input polarity can be inverted individually.
1 = input inverted.
0 = input not inverted.

6005$_h$
Global interrupt enable digital input.
Default value: TRUE.
This object enables and disables globally the interrupt behavior (TPDO transmission) without changing the interrupt masks (objects 6006$_h$, 6007$_h$, 6008$_h$).
TRUE = global interrupt enabled.
FALSE = global interrupt disabled.

6006$_h$

Interrupt mask any change 8-bit.

Default value: FF$_h$.

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

1 = interrupt enabled.

0 = interrupt disabled.

6007$_h$

Interrupt mask low-to-high 8-bit.

Default value: 0$_h$.

This object determines, which input port lines activate an interrupt by positive edge detection (logical 0 to 1). Done for groups of 8 lines. The values are 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 corresponds to negative physical edge.

1 = interrupt enabled.

0 = interrupt disabled.

**6008$_h$**

Interrupt mask high-to-low 8-bit.

Default value: 0$_h$.

This object determines, which input port lines activate an interrupt by negative edge detection (logical 1 to 0). Done for groups of 8 lines. The values are 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 negative logical edge corresponds to positive physical edge.

1 = interrupt enabled.

0 = interrupt disabled.

6.3 Digital counters module.
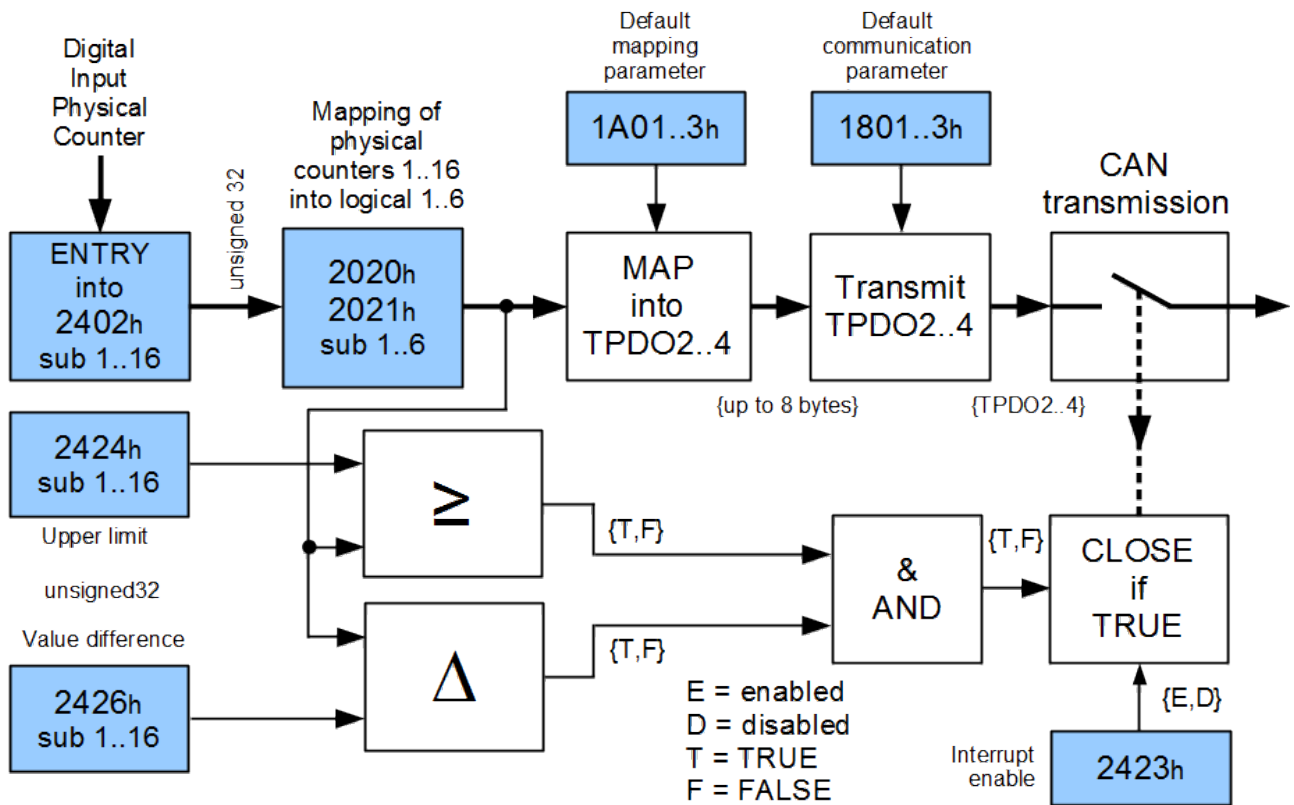
6.3.1 Manufacturer objects.

2020$_h$
Mapping of the physical and logical counters.
Default value: 1 to 6 for sub-indexes 1 to 6.
This object allows to reassign physical counters without TPDO remapping.

2021$_h$
Read logical counter 32-bit unsigned.
This object reads the value of the 32-bit logical counter. Only logical counters can be mapped into TPRO.

2022$_h$
Physical counters reset mask.
Bits 0 to 15 of the mask reset physical counters 1 to 16. This object can be mapped into RPDO.
1 = counter is reset.
0 = counter is not reset.

2402$_h$
Read or set physical counter 32-bit unsigned.
The number of physical counters coincides with the number of digital input bits (0, 8 or 16).
On write access the counter is being set to the downloaded value.

2421$_h$
Physical counter interrupt trigger selection.
Default value: 7$_h$.
This object determines, which events shall cause an interrupt for a specific counter. All bits set to 1 trigger the corresponding physical counter.

Bits	Value	Definition
5-7	0	Reserved.
3,4	0/1	Not used.
2	0/1	Counter changed by more then delta (see object 2426$_h$).
1	0/1	Not used for the counters.
0	0/1	Upper limit exceeded (see object 2424$_h$).

2422$_h$
Physical counter interrupt source.
This object determines, which physical counter has produced an interrupt. Bits set relate to the number of counters that have produced interrupts. The bits are reset automatically after read by SDO.
1 = interrupt produced.
0 = no interrupt produced.

2423$_h$
Counters global interrupt enable.
Default value: FALSE.
This object enables and disables globally the interrupt behavior (TPDO transmission) without changing the interrupt mask (object 2421$_h$).
TRUE = global interrupt enabled.
FALSE = global interrupt disabled.

2424$_h$
Physical counter interrupt upper limit.
Default value: 0$_h$.
If enabled (see object 2423$_h$), an interrupt is triggered when counting is equal or rises above the given value. As long as the trigger condition is met, every change of the counting data generates a new interrupt, if there is no additional trigger condition, e.g. a counter interrupt delta (object 2426$_h$).
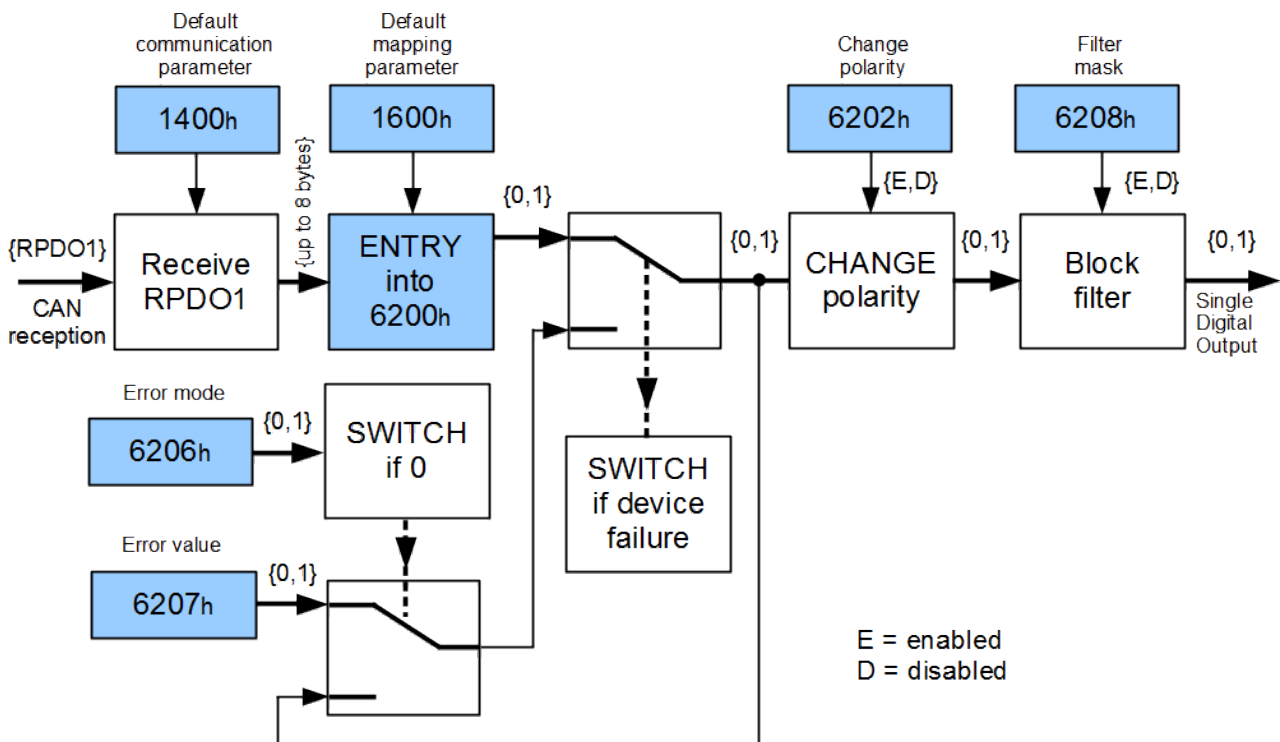
2426$_h$
Physical counter interrupt delta.
Default value: 0$_h$.
This object sets the delta value (rising above the last communicated value) for interrupt-enabled physical counters (see objects 2421$_h$, 2423$_h$).

6.4 Digital output module.

6.4.1 Standardized CiA 401 objects.

6200$_h$
Write output 8-bit.
Default value: 0$_h$.
This object sets groups of one or two 8 output lines as 8-bit information.

6202$_h$
Change polarity output 8-bit.
Default value: 0$_h$.
1 = output inverted.
0 = output not inverted.

6206$_h$
Error mode output 8-bit.
Default value: FF$_h$.
This object indicates, whether an output is set to a pre-defined error value (object 6207$_h$) in case of an internal device failure or a 'Stop Remote Node' indication.
1 = output value shall take the pre-defined condition specified in 6207$_h$ object.
0 = output value shall be kept if an error occurs.

6207$_h$

Error value output 8-bit.

Default value: 0_h.

On condition that the corresponding error mode is active, device failures set the outputs to the value configured by this object.

0 = output shall be set to '0' in case of fault, if object 6206$_h$ is enabled.

1 = output shall be set to '1' in case of fault, if object 6206$_h$ is enabled.

6208$_h$

Filter mask output 8-bit.

Default value: FF$_h$.

This object defines an additional configurable output filter mask for a group of 8 outputs.

1 = output shall be set to the received output value.

0 = don't care, the received output value is neglected for the appropriated output channel, the old output value shall be kept.

6.4.2 Manufacturer objects.

2010$_h$

Status output 8-bit.

The object value is updated regardless of the method of EMCY processing (objects 2011$_h$, 2012$_h$).

1 = output line short circuit.

0 = output O'K.

2011$_h$

Inhibit time EMCY for digital output lines short circuit.

Default value: 10000.

Minimum value: 1000.

Maximum value: 65535.

The value is given in multiples of 100 mcS.

Special value: 0.

When the device detects short circuit at digital output lines, it tries to reset this condition and initiates the EMCY with the error code 2320$_h$ (short circuit at outputs). In case the short circuit is gone, bit 1 (current) of the error register is cleared (see object 1001$_h$)

If the object value is 0, EMCY is initiated with any change of the output lines status (object 2010$_h$). When the inhibit time is other, then zero, emergency messages are transmitted regularly. The EMCY traffic with persistent short circuit shall be determined by both objects 2011$_h$ and 1015$_h$. Manufacturer-specific error code of the R2DIO device EMCY message is filled with the output lines short circuit status. Byte 3 of the EMCY contains lines 1 to 8 status, byte 4 - lines 9 to 16 status.

2012$_h$

Short circuit emergency mask output 8-bit.

Default value: FF$_h$.

1 = EMCY is transmitted in the CAN network.

0 = short circuit is logged in the pre-defined error field (object 1003$_h$) without producing EMCY.

6.4.3 Error mode behavior.

The device transits to error mode in the following cases:
- 'Stop Remote Node' indication;
- A serious device failure is detected, which is processed with the error behavior object 1029_h.

Digital outputs are returned to regular maintenance mode with 'Start Remote Node' or 'Reset Node' indication.

In the error mode an attempt to write objects 6200_h, 6202_h, 6206_h, 6207_h or 6208_h is refused and responded with the SDO abort transfer service (abort code 0800 0022_h, data cannot be transferred or stored to the application because of the present device state). If the device is in the NMT state Operational, EMCY is also initiated with the error code $FF80_h$ (the device is in the error mode). If object 6200_h is updated with RPDO, it is discarded. When exiting the error mode, error register (object 1001_h) is reset.

6.5 Analog input module.
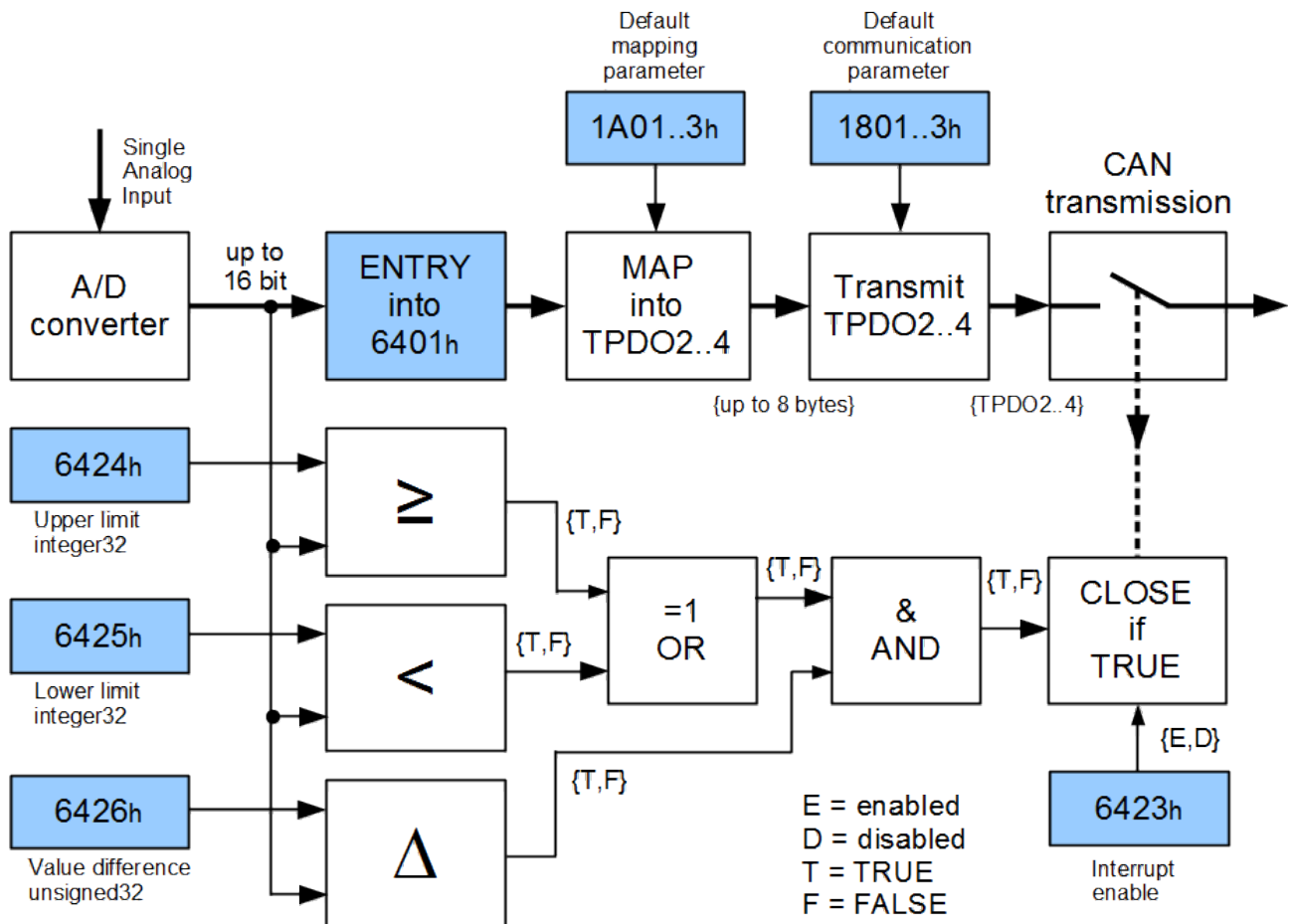
6.5.1 Standardized CiA 401 objects.

6401$_h$

Read analog input 16-bit.

This object reads the value of the 16-bit input channel. The value is left adjusted.

6403$_h$

Read analog input float.

This object reads the float value of the input channel. The float value is calculated by the following formula:

(Float value) = (integer value) * (input scale) + (offset value)

or

(Object 6403$_h$) = (object 6401$_h$) * (object 642F$_h$) + (object 642E$_h$)

6421$_h$

Analog input interrupt trigger selection.

Default value: 7$_h$.

This object determines, which events shall cause an interrupt for a specific channel. All bits set to 1 trigger the corresponding analog input.

Bits	Value	Definition
5-7	0	Reserved.

3,4	0/1	Not used.
2	0/1	Input changed by more then delta (see object 6426_h).
1	0/1	Input below lower limit (see object 6425_h).
0	0/1	Upper limit exceeded (see object 6424_h).

6422_h
Analog input interrupt source.
This object determines, which channel has produced an interrupt. Bits set relate to the number of any channels that have produced interrupts. The bits are reset automatically after read by SDO.
1 = interrupt produced.
0 = no interrupt produced.

6423_h
Analog input global interrupt enable.
Default value: FALSE.
This object enables and disables globally the interrupt behavior (TPDO transmission) without changing the interrupt mask (object 6421_h).
TRUE = global interrupt enabled.
FALSE = global interrupt disabled.

6424_h
Analog input interrupt upper limit integer.
Default value: 0_h.
If enabled (see object 6423_h), an interrupt is triggered when the analog input is equal or rises above the given value. 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 (object 6426_h).

6425_h
Analog input interrupt lower limit integer.
Default value: 0_h.
If enabled (see object 6423_h), an interrupt is triggered when the analog input falls below the given value. 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 (object 6426_h).

6426_h
Analog input interrupt delta unsigned.
Default value: 0_h.
This object sets the delta value (rising above or falling below the last communicated value) for interrupt-enabled analog inputs (see objects 6421_h, 6423_h).

$642E_h$
Analog input offset float.
Default value: 0.0
This object sets the offset in float format for input data (object 6403_h):
(Object 6403_h) = (object 6401_h) * (object $642F_h$) + (object $642E_h$)

642F$_h$
Analog input pre-scaling float.
Default value: 1.0
This object sets the pre-scaling in float format for input data (object 6403$_h$):
(Object 6403$_h$) = (object 6401$_h$) * (object 642F$_h$) + (object 642E$_h$)

6.5.2 Manufacturer objects.

2001$_h$
Analog inputs range index.
Equal for all analog inputs.

Value (hex)	Analog input range
0	0..+10 V
1	0..+5 V
2	0..+2,5 V
8	4 – 20 mA
9	4 – 20 mA
A	4 – 20 mA

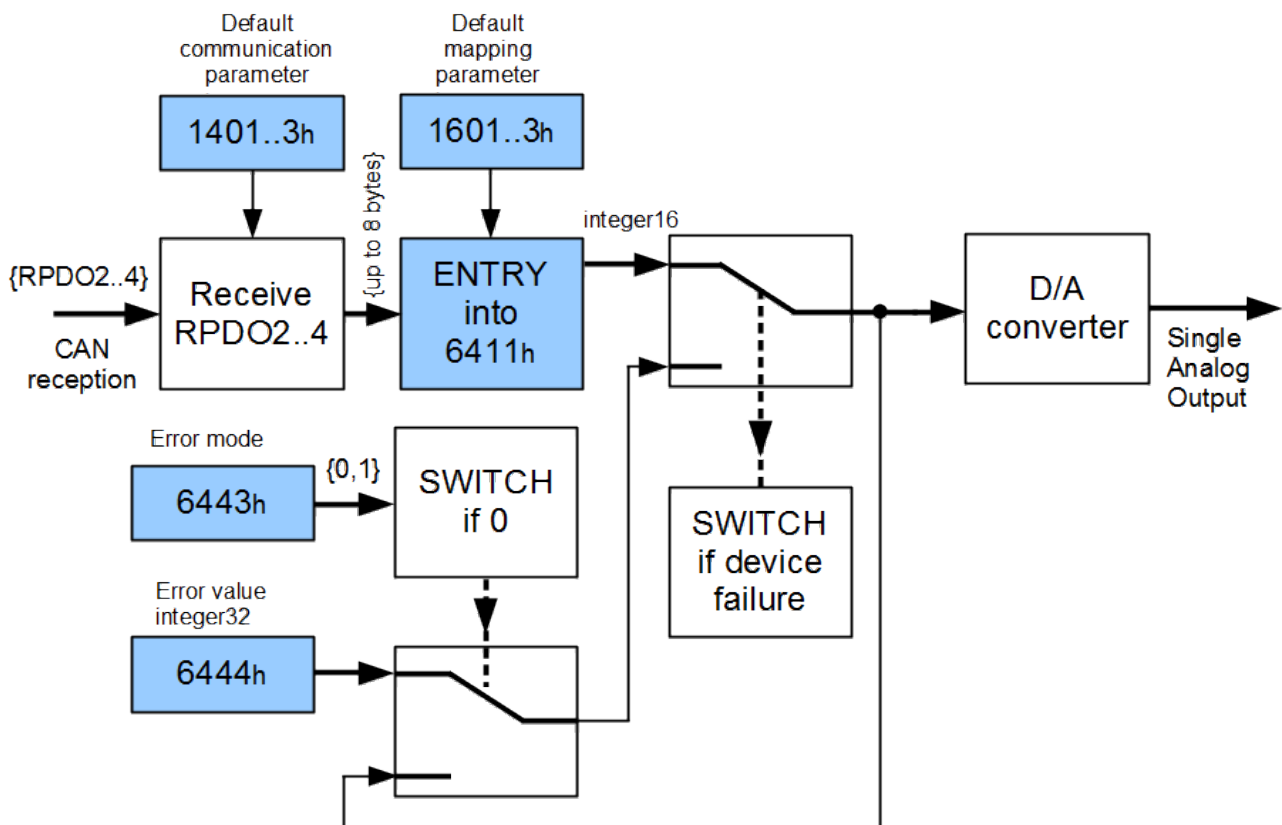**2002$_h$**
Analog inputs rate (SpS) index.
Equal for all analog inputs.

Value (hex)	Rate Samples per Second
0	2,5
1	5
2	10
3	25
4	30
5	50
6	100
7	500

6.6 Analog output module.

6.6.1 Standardized CiA 401 objects.

6411$_h$
Write analog output 16-bit.
This object writes the value to the 16-bit output channel. The value shall be always left adjusted.

6443$_h$
Analog output error mode.
Default value: 1.
This object defines, whether an output is set to a pre-defined error value (object 6444$_h$) in case of an internal device failure or a 'Stop remote node' indication.
1 = reverts to error value integer (object 6444$_h$).
0 = actual value rest.

6444$_h$
Analog output error value integer.
Default value: 0.
On condition that the corresponding error mode is active (object 6443$_h$), device failures set the outputs to the value configured by this object.

6.6.2 Manufacturer objects.

2001$_h$
Analog outputs range index.
Determined for each analog output.

Value (hex)	Analog output range
0	0..+5 V
1	0..+10 V
2	-5..+5 V
3	-10..+10 V
4	4 – 20 mA

2018$_h$
Analog outputs status.
The object value is updated regardless of the method of EMCY processing (objects 2019$_h$, 201A$_h$, 201B$_h$).
1 = output line short circuit (voltage output) or load dump (current output).
0 = output O'K.

2019$_h$
Inhibit time EMCY for the analog outputs short circuit or load dump.
Default value: 10000.
Minimum value: 1000.
Maximum value: 65535.
The value is given in multiples of 100 mcS.
Special value: 0.
When the device detects short circuit (voltage outputs) or load dump (current outputs), it initiates EMCY with the error code 2320$_h$ (short circuit at outputs) or 2330$_h$ (load dump at outputs). In case the error condition is gone, bit 1 (current) of the error register is cleared (see object 1001$_h$)
If the object value is 0, EMCY is initiated with any change of the analog outputs status (object 2018$_h$). When the inhibit time is other, then zero, Emergency messages are transmitted regularly. The EMCY traffic with persistent analog outputs error shall be determined by both objects 2019$_h$ and 1015$_h$. Byte 3 of the manufacturer-specific error code contains analog outputs 1 to 8 status.

201A$_h$
Short circuit emergency mask for the analog outputs.
Default value: FF$_h$.
1 = EMCY is transmitted in the CAN network.
0 = short circuit is logged in the pre-defined error field (object 1003$_h$) without producing EMCY.

201B$_h$
Load dump emergency mask for the analog outputs.
Default value: FF$_h$.
1 = EMCY is transmitted in the CAN network.
0 = load dump is logged in the pre-defined error field (object 1003$_h$) without producing EMCY.

2030$_h$
Slew rate update clock index.
Default value: 0.
Objects 2030$_h$ and 2031$_h$ define the rate of change of the analog output value.

Slew rate control is not supported for the devices, that have manufacturer status register value 00000001$_h$ (see object 1002$_h$).

Object value (hex)	Update clock frequency (Hz)
0	65000
1	32500
2	16250
3	8125
4	4060
5	2030
6	1015
7	500
8	250
9	125
A	64
B	32
C	16
D	8
E	4
F	0.5

2031$_h$
Slew rate step size index.
Default value: 0.
Objects 2030$_h$ and 2031$_h$ define the rate of change of the analog output value.
Slew rate control is not supported for the devices, that have manufacturer status register value 00000001$_h$ (see object 1002$_h$).

Object value (hex)	Step size (LSBs)
0	1
1	2
2	4
3	16
4	32
5	64
6	128
7	256

6.6.3 Error mode behavior.

The device transits to error mode in the following cases:
• 'Stop Remote Node' indication;
• A serious device failure is detected, which is processed with the error behavior object 1029$_h$.
Analog outputs are returned to regular maintenance mode with 'Start Remote Node' or 'Reset Node' indication.
In the error mode an attempt to write objects 6411$_h$, 6443$_h$, and 6444$_h$ is refused and responded with the SDO abort transfer service (abort code 0800 0022$_h$, data cannot be transferred or stored to the application because of the present device state). If the device is in the NMT state Operational, EMCY is also initiated with the error code FF80$_h$ (the device is in the error mode). If object 6411$_h$ is

updated with RPDO, it is discarded. When exiting the error mode, error register (object 1001_h) is reset.

7. IO Remote indicator specification.

IO Remote indication complies with CiA 303 part 3 v. 1.4. Bicolor Red/Green status LED is used. Red LED indication is of higher priority then the Green one.

7.1 Error LED (red).

Error LED	Description
Off	No error. Red LED if turned OFF when the device receives any NMT command.
Blinking	General configuration error.
Single flash	At least one of the error counters of the CAN controller has reached or exceeded the warning level (too many error frames).
Double flash	A guard event has occurred.
Triple flash	The sync message has not been received within the configured communication cycle period time out (object 1006_h).
Quadruple flash	An expected PDO has not been received before the event-timer elapsed.
On	Bus off. The device is logically disconnected from the CAN data link layer.

7.2 Run LED (green).

Run LED	Description
Blinking	The device is in state PRE-OPERATIONAL.
Single flash	The device is in state STOPPED.
On	The device is in state OPERATIONAL.

Both LEDs are off when CAN Node-ID or CAN bit rate index is invalid (the device does not start). If the device receives invalid NMT command, both LEDs turn off, while IO Remote NMT state is not changed.

8. CANopen error codes.

8.1 SDO abort codes.

Аборт код	Описание
0503 0000$_h$	Toggle bit not alternated.
0504 0000$_h$	SDO protocol timed out.
0504 0001$_h$	Client/server command specifier not valid or unknown.
0504 0002$_h$	Invalid block size (block mode only).
0504 0003$_h$	Invalid sequence number (block mode only).
0504 0004$_h$	CRC error (block mode only).
0504 0005$_h$	Out of memory.
0601 0000$_h$	Unsupported access to an object.
0601 0001$_h$	Attempt to read a write only object.
0601 0002$_h$	Attempt to write a read only object.
0602 0000$_h$	Object does not exist in the object dictionary.
0604 0041$_h$	Object cannot be mapped to the PDO.
0604 0042$_h$	The number and length of the objects to be mapped would exceed PDO length.
0604 0043$_h$	General parameter incompatibility reason.
0604 0047$_h$	General internal incompatibility in the device.
0606 0000$_h$	Access failed due to a hardware error.
0607 0010$_h$	Data type does not match, length of service parameter does not match.
0607 0012$_h$	Data type does not match, length of service parameter too high.
0607 0013$_h$	Data type does not match, length of service parameter too low.
0609 0011$_h$	Sub-index does not exist.
0609 0030$_h$	Invalid value for parameter (download only).
0609 0031$_h$	Value of parameter written too high (download only).
0609 0032$_h$	Value of parameter written too low (download only).
0609 0036$_h$	Maximum value is less than minimum value.
060A 0023$_h$	Resource not available: SDO connection.
0800 0000$_h$	General error.
0800 0020$_h$	Data cannot be transferred or stored to the application.
0800 0021$_h$	Data cannot be transferred or stored to the application because of local control.
0800 0022$_h$	Data cannot be transferred or stored to the application because of the present device state.
0800 0023$_h$	Object dictionary dynamic generation fails or no object dictionary is present.
0800 0024$_h$	No data available.

8.2 Emergency error codes.

Error code (hex)	Description
0000	Error reset or no error.
1000	Generic error.
2000	Current – generic error.
2100	Current, CANopen device input side – generic.
2200	Current inside the CANopen device – generic.
2300	Current, CANopen device output side – generic.
2320	Short circuit at outputs.
2330	Load dump at outputs.
3000	Voltage – generic error.
3100	Mains voltage – generic.
3200	Voltage inside the CANopen device – generic.
3300	Output voltage – generic.
4000	Temperature – generic error.
4100	Ambient temperature – generic.
4200	Device temperature – generic.
5000	CANopen device hardware – generic error.
6000	CANopen device software – generic error.
6100	Internal software – generic.
6180	CANopen cache overflow.
6190	CANopen timer initialization error.
6191	CANopen timer overlap.
61A0	Non-volatile memory data error.
61A1	Non-volatile memory operations error.
6200	User software – generic.
6300	Data set – generic.
7000	Additional modules – generic error.
8000	Monitoring – generic error.
8100	Communication – generic.
8110	CAN overrun (objects lost).
8120	CAN in error passive mode.
8130	Life guard error or heartbeat error.
8140	Recovered from bus off.
8150	CAN-ID collision.
8180	CAN controller event «hardware overrun».

8181	CAN controller event «software overrun».
8182	CAN controller event «error warning limit».
8183	CAN controller event «write timeout».
8200	Protocol error - generic.
8210	PDO not processed due to length error.
8220	PDO length exceeded.
8230	DAM MPDO not processed, destination object not available.
8240	Unexpected SYNC data length.
8250	RPDO timeout.
9000	External error – generic error.
F000	Additional functions – generic error.
FF00	Device specific – generic error.
FF80	The device is in the error mode.

Color marked are manufacturer-specific error codes.

Errors 6180_h, 6190_h, $61A0_h$ $61A1_h$ are listed in the pre-defined error field (object 1003_h), but do not initiate EMCY, because the Emergency service can not be executed.

9. Generic pre-defined connection set.

9.1 Broadcast objects.

CAN-ID	Communication object	Object index
0	NMT	---
128 (80$_h$)	SYNC	1005$_h$, 1006$_h$
256 (100$_h$)	TIME	1012$_h$, 1013$_h$

9.2 Peer-to-peer objects.

CAN-ID	Communication object	Object index
129 (81$_h$) − 255 (FF$_h$)	EMCY for CAN node-IDs 1 to 127	1014$_h$, 1015$_h$
385 (181$_h$) − 511 (1FF$_h$)	TPDO 1 for CAN node-IDs 1 to 127	1800$_h$
513 (201$_h$) − 639 (27F$_h$)	RPDO 1 for CAN node-IDs 1 to 127	1400$_h$
641 (281$_h$) − 767 (2FF$_h$)	TPDO 2 for CAN node-IDs 1 to 127	1801$_h$
769 (301$_h$) − 895 (37F$_h$)	RPDO 2 for CAN node-IDs 1 to 127	1401$_h$
897 (381$_h$) − 1023 (3FF$_h$)	TPDO 3 for CAN node-IDs 1 to 127	1802$_h$
1025 (401$_h$) − 1151 (47F$_h$)	RPDO 3 for CAN node-IDs 1 to 127	1402$_h$
1153 (481$_h$) − 1279 (4FF$_h$)	TPDO 4 for CAN node-IDs 1 to 127	1803$_h$
1281 (501$_h$) − 1407 (57F$_h$)	RPDO 4 for CAN node-IDs 1 to 127	1403$_h$
1409 (581$_h$) − 1535 (5FF$_h$)	SDO server->client for CAN node-IDs 1 to 127	1200$_h$
1537 (601$_h$) − 1663 (67F$_h$)	SDO client->server for CAN node-IDs 1 to 127	1200$_h$
1793 (701$_h$) − 1919 (77F$_h$)	NMT error control for CAN node-IDs 1 to 127	1016$_h$, 1017$_h$

9.3 Other objects.

CAN-ID	Communication object
2020 (7E4$_h$)	LSS slave device messages
2021 (7E5$_h$)	LSS master device messages

9.4 Restricted CAN-IDs.

Such a restricted CAN-ID shall not be used as a CAN-ID by any configurable communication object, neither for SYNC, TIME, EMCY, PDO and SDO.

CAN-ID	Communication object
0	NMT
1 (001$_h$) − 127 (07F$_h$)	reserved
257 (101$_h$) − 384 (180$_h$)	reserved
1409 (581$_h$) − 1535 (5FF$_h$)	default SDO server->client (tx)
1537 (601$_h$) − 1663 (67F$_h$)	default SDO client->server (rx)
1760 (6E0$_h$) − 1791 (6FF$_h$)	reserved

1793 (701$_h$) – 1919 (77F$_h$)	NMT error control
1920 (780$_h$) – 2047 (7FF$_h$)	Reserved

10. Tools for IO Remote module family.

IO Remote configuration and testing tools are fulfilled as DLL plug-ins for the CANwise program.

10.1 DCF based configuration.

Plug-in file: CANopenDCF.dll
Plug-in name: Device Configuration Manager
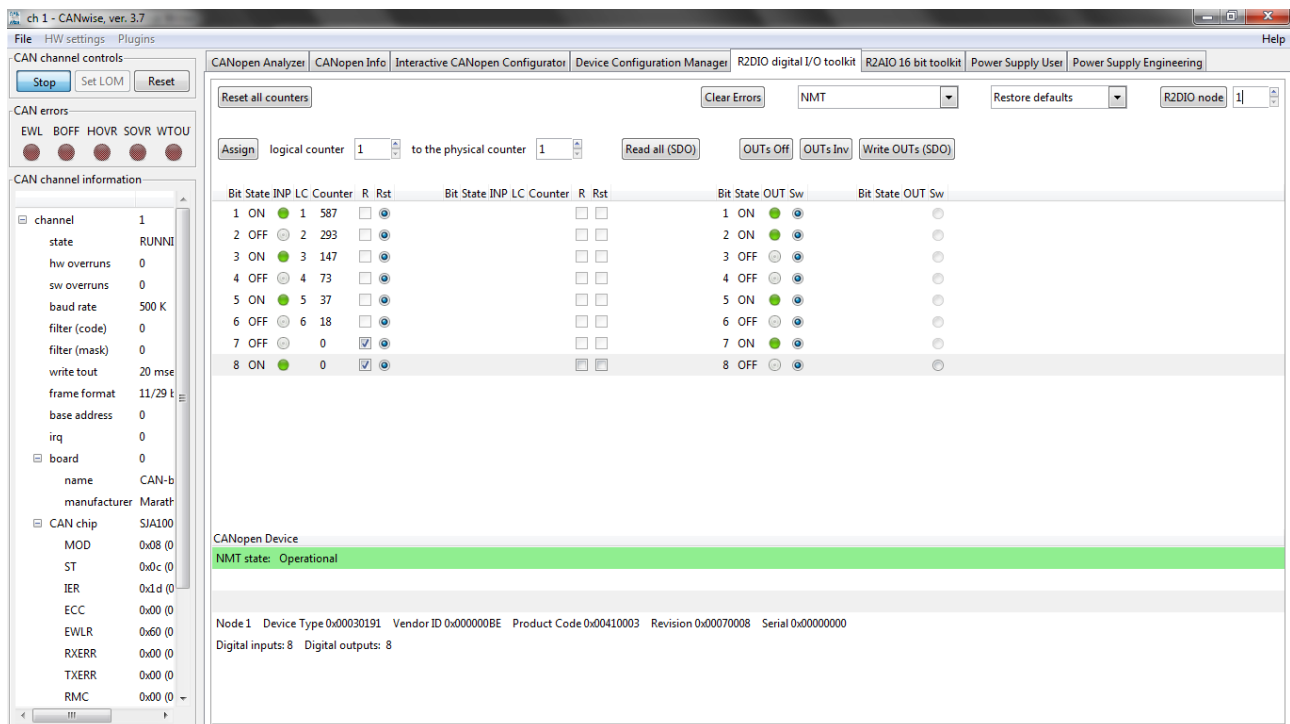Reference guide: CANopen_DCF.pdf

10.2 Digital I/O and counters tool.

Plug-in file: CANopenR2DIO.dll
Plug-in name: R2DIO digital I/O toolkit
Version 1.8.x

Digital I/O tool presents two application windows and a number of control buttons:

The top window allows to manage the device digital inputs, counters and digital outputs. The bottom one is used to describe the device type and display its status.

To start working with CANwise:
· set CAN network bitrate if needed;
· run CANwise, clicking 'Start' button;
Then specify CAN node-ID and click 'R2DIO node' button. Upon successful detection of the R2DIO device its configuration is checked and heartbeat protocol activated.

Each digital input can be in the state OFF or state ON. For digital outputs three states are possible: OFF, ON and short circuit (CS). Short circuit state can only occur when the output is in the ON state.

Control buttons.

Button	Description
Reset all counters	Resets all counters to zero value. CANopen PDO protocol is used.
Clear Errors	Clears error message lines in the bottom window.
NMT	Pop-down menu. Allows to choose and send NMT command in the CAN network. All commands are addressed to the specified R2DIO node.
Restore defaults	Pop-down menu. ALL – restores all default parameters; Communication – restores communication default parameters; Application – restores application default parameters; Pre-defined CAN-ID – restores pre-defined connection set CAN-IDs; NODE ID – restores CAN node-ID selection with hardware switch; BIT RATE – restores CAN bit rate index selection with hardware switch.
R2DIO node	Start operations with the device. The R2DIO node-ID is selected with the spin button.
Assign	Mapping of the physical and logical counters.
Read all (SDO)	Uploads from the device and shows: all digital inputs; all digital counters; physical and logical counters mapping table; all digital outputs; digital outputs state (OFF, ON, short circuit). CANopen SDO protocol is used.
OUTs Off	Writes OFF value to all digital outputs. CANopen PDO protocol is used.
OUTs Inv	Writes inverted polarity value to all digital outputs. CANopen PDO protocol is used.
Write OUTs (SDO)	Writes all digital outputs. CANopen SDO protocol is used.

In normal mode PDO protocol is used to read digital inputs and write digital outputs. PDO service is executed if the device is in the Operational NMT state. 'Read all (SDO)' and 'Write OUTs (SDO)' buttons use SDO service, which is executed if the device is in the Operational or Pre-operational NMT state.

Digital inputs, counters and digital outputs table:

Column	Description
Digital inputs and counters table. Repeats for inputs 1 to 8 and 9 to16.	
Bit	Digital input and physical counter number.
State	Digital input state ON/OFF.
INP	Mnemonic digital input display.
LC	Logical counter number.
Counter	Physical (logical) counter value.

|	
R	Read physical counter value, using SDO protocol. Active for the physical counters, which are not mapped to the logical ones.
Rst	Resets physical counter to zero value. CANopen PDO protocol is used.
Digital outputs table. Repeats for outputs 1 to 8 and 9 to16.	
Bit	Digital output number.
State	Digital output state ON/OFF.
OUT	Mnemonic digital output display, including short circuit state.
Sw	Digital output switch ON/OFF. CANopen PDO protocol is used.

The device type and status window.

Line	Description
1	The device NMT state.
2	Incoming errors (Emergency, invalid PDO length etc.).
3	Outgoing errors (CAN frame transmit error etc.).
4	Product information (device type, vendor-ID, product code etc.).
5	The number of digital inputs and digital outputs.

10.3 Analog I/O tool.

Plug-in file: CANopenR2AIO_16.dll
Plug-in name: R2AIO 16 bit toolkit
Version 1.8.x

Analog I/O tool presents two application windows and a number of control buttons:
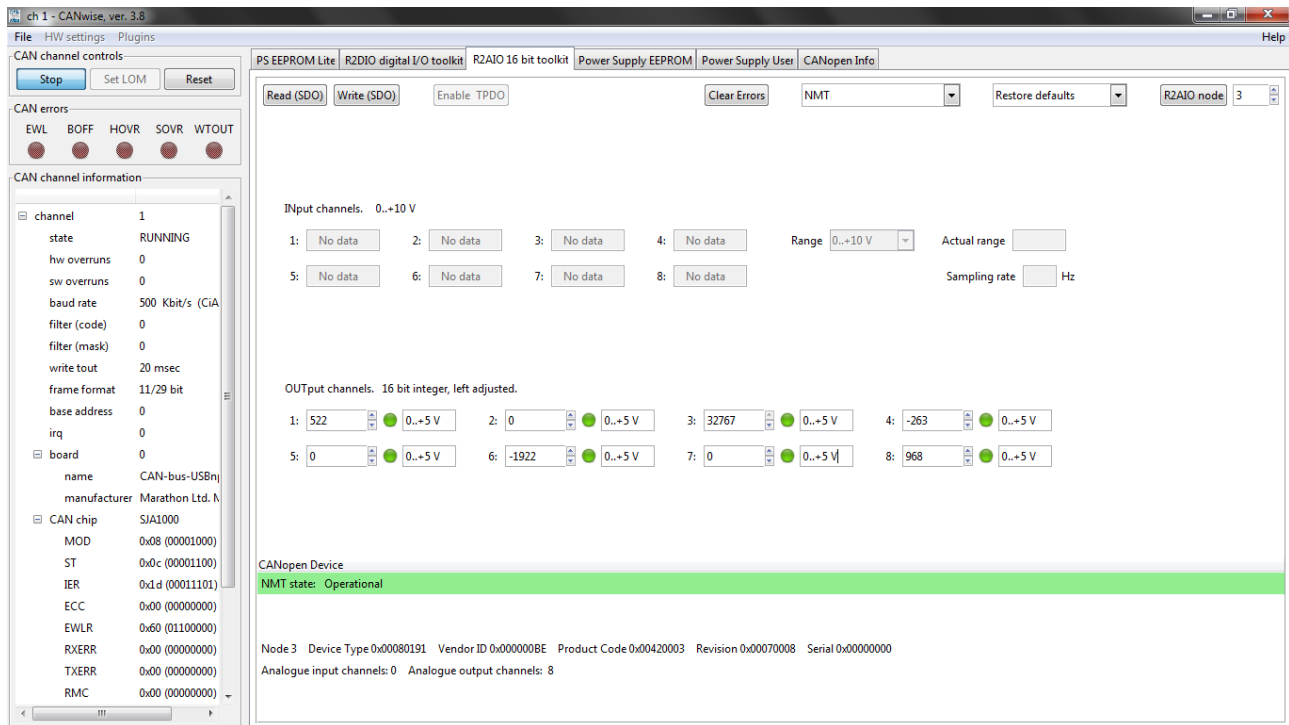
The top window shows the device analog inputs or allows to control the analog outputs. The bottom one is used to describe the device type and display its status.

To start working with CANwise:
•   set CAN network bitrate if needed;
•   run CANwise, clicking 'Start' button;
Then specify CAN node-ID and click 'R2AIO node' button. Upon successful detection of the R2AIO device its configuration is checked and heartbeat protocol activated.

The value of analog outputs shall be left adjusted. Analog output status is displayed to the right of its spin button entry field. Green mnemonic indicates normal state of outputs, red one – short circuit at voltage outputs or load dump at current outputs.

Control buttons.

Button	Description
Read (SDO)	Uploads from the device and shows: all analog inputs; all analog outputs; analog outputs state (O'K, short circuit, load dump). CANopen SDO protocol is used.
Write (SDO)	Writes all analog outputs. CANopen SDO protocol is used.

Enable TPDO / Disable TPDO	Analog input global interrupt enable control button (object 6423$_h$).
Clear Errors	Clears error message lines in the bottom window.
NMT	Pop-down menu. Allows to choose and send NMT command in the CAN network. All commands are addressed to the specified R2AIO node.
Restore defaults	Pop-down menu. ALL – restores all default parameters; Communication – restores communication default parameters; Application – restores application default parameters; Pre-defined CAN-ID – restores pre-defined connection set CAN-IDs; NODE ID – restores CAN node-ID selection with hardware switch; BIT RATE – restores CAN bit rate index selection with hardware switch.
R2AIO node	Start operations with the device. The R2AIO CAN node-ID is selected with the spin button.
Range	Pop-down menu. Used to drive the numeric values of analog inputs to one of the measurement ranges. Actual measurement range is read from the device and shown in the 'Actual range' field.

In normal mode PDO protocol is used to read analog inputs and write analog outputs. PDO service is executed if the device is in the Operational NMT state. 'Read (SDO)' and 'Write (SDO)' buttons use SDO service, which is executed if the device is in the Operational or Pre-operational NMT state.

The device type and status window.

Button	Description
1	The device NMT state.
2	Incoming errors (Emergency, invalid PDO length etc.).
3	Outgoing errors (CAN frame transmit error etc.).
4	Product information (device type, vendor-ID, product code etc.).
5	The number of analog inputs or outputs.