PIXSYS

elettronica

EXPANSION MODULE MODBUS - CANOPEN

MCM260

Manuale
User manual

TABLE OF CONTENTS

1 INTRODUCTION

Thank you for choosing a Pixsys instrument. MCM260 are digital expansion modules for PLC; communication with other devices is provided by Modbus or CANopen protocol.
MCM260 series includes 5 models as described in the table below.

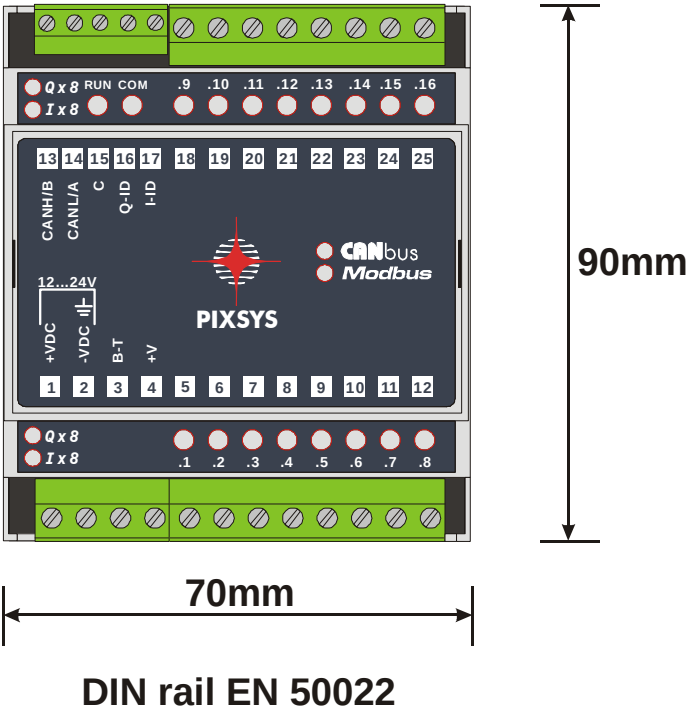
1.1 Model identification

MCM260-	☐	
Power supply Inputs / Outputs	**1AD**	**12…24Vdc** ±15% **16 Static outputs 24Vdc** 700mA MAX
Power supply Inputs / Outputs	**2AD**	**12…24Vac/dc** ±15% 50/60Hz **16 Digital inputs PNP 24Vdc** **2 Analog inputs 0...10V**
Power supply Inputs / Outputs	**3AD**	**12…24Vdc** ±15% **8 Digital inputs PNP 24Vdc** **8 Static outputs 24Vdc** 700mA MAX
Power supply Inputs / Outputs	**4AD**	**12…24Vac/dc** ±15% **8 Digital inputs PNP 24Vdc** **8 relay outputs** 5A-250V~ resistive load
Power supply Inputs / Outputs	**5AD**	**12…24Vac/dc** ±15% **4 Analog inputs** **2 Analog outputs** (0..10V or 4..20mA)

2 CHARACTERISTICS

General characteristics		
Operating conditions	0-40℃ - humidity 35..95uR%	
Material	Noryl UL 94 V-0	
Weight	250g	
Box	4 DIN rail modules	
Dimensions	70x90mm(frontal)x53mm	
Sealing	IP30 box	
Hardware characteristics		
Inputs	**MCM260-2AD, MCM260-3AD, MCM260-4AD** Digital inputs PNP 24Vdc	
	MCM260-5AD Inputs Configurable via software. **Thermocouples**: type K,S,R,J; automatic compensation of cold junction from 0℃ to 50℃. **Thermoresistance**: PT100, PT500, PT1000, NI100, PTC1K, NTC10K (β 3435K) **Linear**: 0-10V, 0-20 or 4-20mA, 0-40mV **Potentiometers**: 6KΩ, 1506KΩ	Tolerance (25℃) 0.2% ±1 unit
Outputs	**MCM260-1AD, MCM260-3AD** Static outputs: 24Vdc – 700mA max Each output can give 700mA, max consumption 4A	
	MCM260-4AD 8 relays: contacts 5A-250V~ resistive load	
	MCM260-5AD 2 linear 0-10V or 4-20mA 0-10V: resolution 7680 points. 4-20mA: resolution 6500 points	
Software characteristics		
Communication protocols	Modbus RTU / CANopen	

2.1 Mechanical dimensions and installation

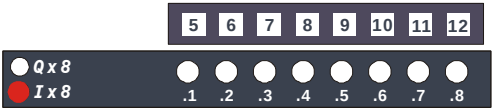
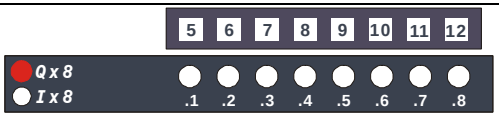
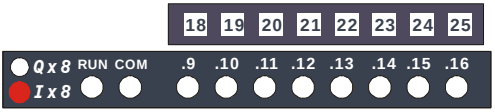
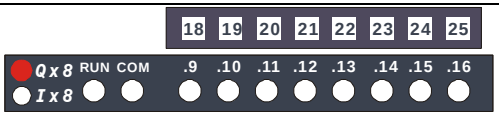


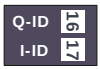
DIN rail EN 50022

2.2 Electrical connections

2.2.1 MCM260-1/2/3AD

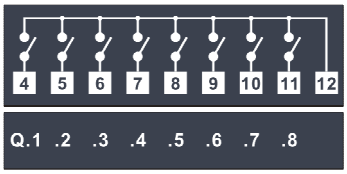
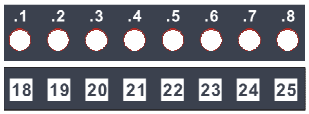
1 +VDC 2 -VDC 12...24V	Power supply 12…24Vdc (Vac for MCM260-2AD) ±15% • 1: +Vdc • 2: -Vdc
3 B-T	This pin allows to switch on logic part of device only. If +Vdc is connected to pin 3 and not to pin1, outputs won't be active.
4 +V	Common pin for digital inputs 12…24Vdc

	**MCM260-2AD, MCM260-3AD** Digital inputs PNP 24Vdc • 5: Input 1 • 6: Input 2 • 7: Input 3 • 8: Input 4 • 9: Input 5 • 10: Input 6 • 11: Input 7 • 12: Input 8
	**MCM260-1AD** Static outputs 24Vdc • 5: Output 1 • 6: Output 2 • 7: Output 3 • 8: Output 4 • 9: Output 5 • 10: Output 6 • 11: Output 7 • 12: Output 8
	**MCM260-2AD** Digital inputs PNP 24Vdc • 18: Input 9 • 19: Input 10 • 20: Input 11 • 21: Input 12 • 22: Input 13 • 23: Input 14 • 24: Input 15 • 25: Input 16
	**MCM260-1AD** Static outputs 24Vdc • 18: Output 9 • 19: Output 10 • 20: Output 11 • 21: Output 12 • 22: Output 13

	• 23: Output 14 • 24: Output 15 • 25: Output 16 **MCM260-3AD** Static outputs 24Vdc • 18: Output 1 • 19: Output 2 • 20: Output 3 • 21: Output 4 • 22: Output 5 • 23: Output 6 • 24: Output 7 • 25: Output 8
	Analogue input 0...10V 10bit (**MCM260-2AD** only)[1] • 6: Input 1 • 7: Input 2 • 2: Common inputs
	2 counter inputs for proximity (max 2KHz) • 9: Proximity 1 • 10: Proximity 2
	Encoder input (max 10KHz) • 12: A phase • 11: B phase
	Field bus: • 13: CANH / RS485+ • 14: CANL / RS485- • 15: C GND for CANbus and Modbus RTU
	Automatic addressing pins (Modbus RTU only) • 16: Output automatic addressing • 17: Input automatic addressing

[1] Analogue input linearization is fixed and not changeable, from 0 to 10000: 0V→0, 10V→10000

2.2.2 MCM260-4AD

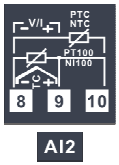
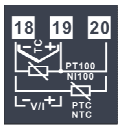
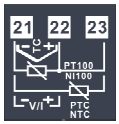
	Power supply 12…24Vac/dc ±15% • 1: +Vdc • 2: -Vdc
	Common pin for digital inputs 12…24Vdc
	Relay outputs: • 4: output 1 • 5: output 2 • 6: output 3 • 7: output 4 • 8: output 5 • 9: output 6 • 10: output 7 • 11: output 8 • 12: Common relays
	Digital inputs PNP 24Vdc • 18: input 1 • 19: input 2 • 20: input 3 • 21: input 4 • 22: input 5 • 23: input 6 • 24: input 7 • 25: input 8
	Analogue input 0...10V 10bit[2] • 19: Input 1 • 20: Input 2 • 2: Common inputs

[2] Analogue input linearization is fixed and not changeable, from 0 to 10000: 0V→0, 10V→10000

	2 counter inputs for proximity (max 2KHz) • 22: Proximity 1 • 23: Proximity 2
	Encoder input (max 10KHz) • 24: A phase • 25: B phase
	Field bus: • 13: CANH / RS485+ • 14: CANL / RS485- • 15: C GND for CANbus and Modbus RTU
	Automatic addressing pins (Modbus RTU only) • 16: Output automatic addressing • 17: Input automatic addressing

2.2.3 MCM260-5AD

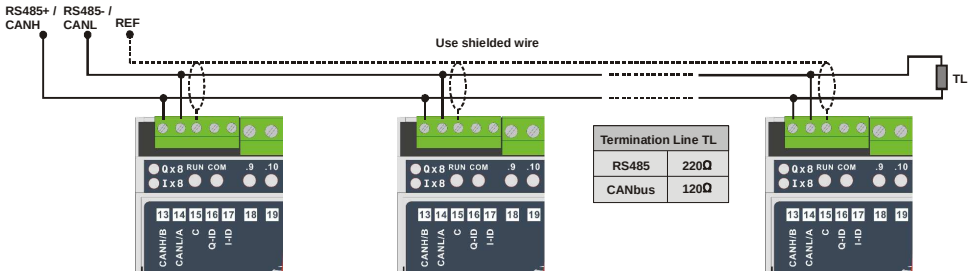
	Power supply 12…24Vac/dc ±15% • 1: +Vdc • 2: -Vdc
	Reference of common pin for digital inputs 24Vdc
	Common pin for digital inputs 12Vdc
	Analogue input AI1 • Thermocouples K, S, R, J – **Terminals 5-6.** • Pt100, Ni100 – **Terminals 5-6-7** (for the two-wire connection short-circuit terminals 6 and 7). • Thermoresistances – **Terminals 5-7.** • Linear signals – **Terminals 5-6.** Linear sensor power supply is available on terminal 4. • Potentiometers– **Terminals 5-7.**

 AI2	Analogue input AI2 • Thermocouples K, S, R, J – **Terminals 8-9.** • Pt100, Ni100 – **Terminals 8-9-10** (for the two-wire connection short-circuit terminals 9 and 10). • Thermoresistances – **Terminals 8-10.** • Linear signals – **Terminals 8-9.** Linear sensor power supply is available on terminal 4. • Potentiometers– **Terminals 8-10.**
AI3 	Analogue input AI3 • Thermocouples K,S,R,J - **Terminals 18-19.** • Pt100, Ni100 – **Terminals 18-19-20** (for the two-wire connection short-circuit terminals 19 and 20). • Thermoresistances – **Terminals 18-20.** • Linear signals – **Terminals 18-19.** Linear sensor power supply is available on terminal 4. • Potentiometers– **Terminals 18-20.**
AI4 	Analogue input AI3 • Thermocouples K,S,R,J - **Terminals 21-22.** • Pt100, Ni100 – **Terminals 21-22-23** (for the two-wire connection short-circuit terminals 22 and 23). • Thermoresistances – **Terminals 21-23.** • Linear signals – **Terminals 21-22.** Linear sensor power supply is available on terminal 4. • Potentiometers– **Terminals 21-23.**
 AO1	Analogue output AO1 • 4…20mA remove jumper S3. • 0…10V insert jumper S3.
AO2 	Analogue output AO2 • 4…20mA remove jumper S4. • 0…10V insert jumper S4.

	Field bus: • 13: CANH / RS485+ • 14: CANL / RS485- • 15: C GND for CANbus and Modbus RTU
	Automatic addressing pins (Modbus RTU only) • 16: Output automatic addressing • 17: Input automatic addressing

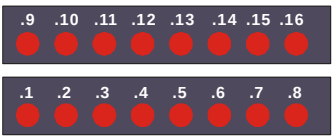
2.3 Wiring connection to network

Following diagram shows connection of MCM260s to RS485 serial line or to a CAN network.



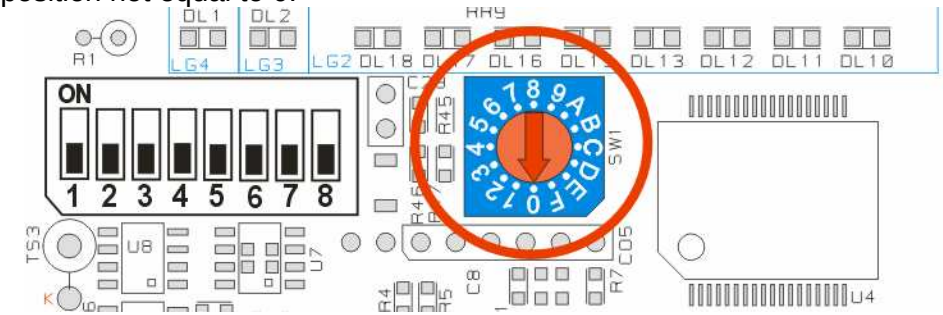
3 LED FUNCTIONS

	• LED **RUN** (green): it indicates device is on and all operative states. • LED **COM** (amber): it indicates MCM260 is communicating with other devices.

	**MCM260-1AD: .1 ..... .16** indicate output status of Out1...Out16 **MCM260-2AD: .1 ..... .16** indicate input status of In1...In16 **MCM260-3AD:** **MCM260-4AD: .1 ..... .8** indicate input status of In1...In8 **.8 ..... .16** indicate output status of Out1...Out8

4 MCM260 SETUP

To enter **SET-UP** mode, device has to be powered-on when rotary switch is in position 0; MCM260 will be allowed to work as a node of a network (Modbus or CANopen) if it has been powered-on in any rotary switch position not equal to 0.



For both selectable protocols (Modbus or CANopen), rotary switch positions between 1 and 8 are **address offset multiplier**, while positions between 9 and F are **baud-rate**. Node address will be:

Node-ID = (**N-1) x 15 + M**,

where **N** = switch position saved during SET-UP (1...8), **M** = actual switch position if device has been powered-on in a position not equal to 0.

SET-UP			
Switch position	**Type of set-up**	**Description**	
0 - 0x00	-	Waiting a rotary switch change	
1 - 0x01	Address Offset	Offset 0 – Address from 1 to 15	
2 - 0x02	Address Offset	Offset 15 - Address from 16 to 30	
3 - 0x03	Address Offset	Offset 30 - Address from 31 to 45	
4 - 0x04	Address Offset	Offset 45 - Address from 46 to 60	
5 - 0x05	Address Offset	Offset 60 - Address from 61 to 75	
6 - 0x06	Address Offset	Offset 75 - Address from 76 to 90	
7 - 0x07	Address Offset	Offset 90 - Address from 91 to 105	
8 - 0x08	Address Offset	Offset 105 - Address from 106 to 120	
		Modbus	**CANopen**
9 - 0x09	Baud rate	2400 bit/sec	50 Kbit/sec
10 - 0x0A	Baud rate	4800 bit/sec	62.5 Kbit/sec
11 - 0x0B	Baud rate	9600 bit/sec	100 Kbit/sec
12 - 0x0C	Baud rate	19200 bit/sec	125 Kbit/sec
13 - 0x0D	Baud rate	28800 bit/sec	250 Kbit/sec
14 - 0x0E	Baud rate	38400 bit/sec	500 Kbit/sec
15 - 0x0F	Baud rate	57600 bit/sec	1 Mbit/sec

5 MODBUS RTU

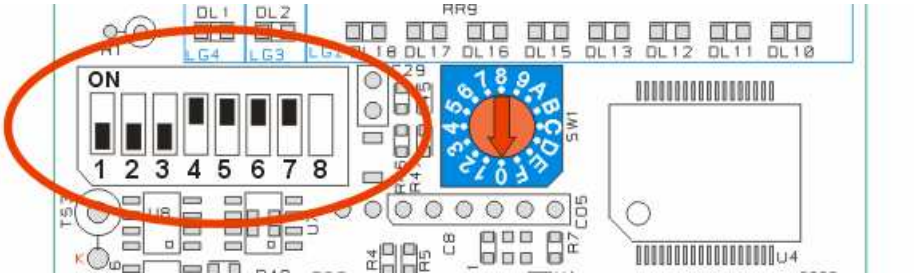
Each LED RUN blinking type indicates a particular Modbus RTU status.

Blinking name LED RUN	Blinking type
Blink_fast	Fast blinking 50msec
Blink_medium	Blinking 200msec
Blink_slow	Blinking 600msec
LED_on	LED always on
Blink_3_on	LED on for 1sec, 3 blink 150msec
Blink_1_off	Slow blink 40msec every 1.2sec
Blink_3_off	LED off for 1sec, 3 blink 150msec

STATUS	LED RUN BLINKING
Boot-up	Blink_fast
Rotary switch in moving (during set-up)	Blink_fast
Waiting for selection	Blink_1_off
Address offset multiplier saved	Blink_3_on
Baud-rate saved	LED_on
MCM260 in normal operation mode	LED_on
Off-line Signalling occurred	Blink_medium

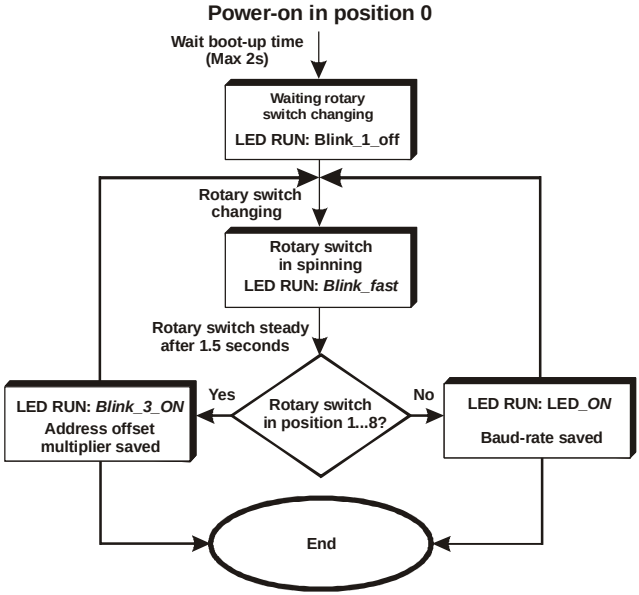
5.1 SET-UP as Modbus RTU slave node

To configure MCM260 as a node of Modbus RTU network, dip-switch has to be set as shown in the figure of the following page.



Subsequently procedure shown in the following page has to be done to configure address offset multiplier and baud-rate.
Once time procedure is ended, device is still not ready to work: it will already be in SET-UP mode until it won't be powered-off.

SET-UP procedure for Modbus RTU



5.2 Operation as Modbus RTU slave node

If dip-switch is configured as shown in figure on page 10, when MCM will be powered-on with rotary switch in position not equal to 0 (1...F), it will be ready to work as a Modbus RTU node. Its baud-rate was selected during SET-UP, its address is calculated by actual rotary switch position added to offset selected during SET-UP.

NORMAL OPERATION MODE	
Rotary switch position	**Description**
0 - 0x00	If powered-on with rotary switch in position not equal to 0, its changing involves device in STOPPED status (node not active).
1…15 - 0x00…0x0F	Defines node address: offset selected during SET-UP has to be added to this value. Example: if in SET-UP it was set to position 5, and in normal operation mode actual switch position is 1, node address will be 60 (offset, 15x(5-1)) + 1 = 61

5.3 Modbus RTU protocol characteristics

Electrical cable provided for Modbus RTU configuration is serial line RS485 optoisolated, available in pins 13 (485+), 14 (485-) and 15 (GND).

Baud-rate	Selectable by rotary switch: • 57600 bits/s • 9600 bits/s • 38400 bits/s • 4800 bits/s • 28800 bits/s • 2400 bits/s • 19200 bits/s
Format	8, N, 1 (8bit, no parity, 1 stop)
Supported functions	WORD READING (max. 20 word) (0x03, 0x04) SINGLE WORD WRITING (0x06) MULTIPLE WORD WRITING (max 20 word) (0x10)

5.4 Modbus RTU communication area

5.4.1 MCM260-1AD, MCM260-2AD, MCM260-3AD, MCM260-4AD

Modbus Address	Description	
0	**Device Type**	
	Contains the device identification code. (Default 160)	RO
1	**Firmware Version**	
	Contains the device firmware version.	RO
5	**Slave Address**	
	Contains the selected modbus slave address. (Depends on the position of the rotary-switch)	RO
998	**Analogue Input 1**	
	Contains the value in mV of the analogue input 1 (0-10000)	RO
999	**Analogue Input 2**	
	Contains the value in mV of the analogue input 2 (0-10000)	RO
1000	**Digital Inputs Status**	
	Contains the digital inputs logic state: Bit 0: Input 1 Bit 8: Input 9 Bit 1: Input 2 Bit 9: Input 10 Bit 2: Input 3 Bit 10: Input 11 Bit 3: Input 4 Bit 11: Input 12 Bit 4: Input 5 Bit 12: Input 13 Bit 5: Input 6 Bit 13: Input 14 Bit 6: Input 7 Bit 14: Input 15 Bit 7: Input 8 Bit 15: Input 16	RO

Modbus Address	Description	
1001	**Counts encoder H**	
	More significant word of the double-word which contains the encoder counts	RO
1002	**Counts encoder L**	
	Less significant word of the double-word which contains the encoder counts	RO
1003	**Counts proximity 1 H**	
	More significant word of the double-word which contains the proximity 1 counts	R/W
1004	**Counts proximity 1 L**	
	Less significant word of the double-word which contains the proximity 1 counts	R/W
1005	**Counts proximity 2 H**	
	More significant word of the double-word which contains the proximity 2 counts	R/W
1006	**Counts proximity 2 L**	
	Less significant word of the double-word which contains the proximity 2 counts	R/W
1007	**Speed encoder - 100ms**	
	Contains the number of encoder counts detected in 100ms	RO
1008	**Speed encoder - 1s**	
	Contains the number of encoder counts detected in 1s	RO
1009	**Time input On (0.5ms) proximity 1**	
	Indicates how long the proximity input 1 remained at logical value 1. The accuracy is 0.5ms. (Ex. 20=10ms).	RO
1010	**Speed proximity 1 - 100ms**	
	Contains the number of proximity 1 counts detected in 100ms	RO
1011	**Speed proximity 1 - 1s**	
	Contains the number of proximity 1 counts detected in 1s.	RO
1012	**Time Input On (0.5ms) proximity 2**	
	Indicates how long the proximity input 2 remained at logical value 1. The accuracy is 0.5ms. (Ex. 20=10ms).	RO
1013	**Speed proximity 2 - 100ms**	
	Contains the number of proximity 2 counts detected in 100ms	RO
1014	**Speed proximity 2 - 1s**	
	Contains the number of proximity 2 counts detected in 1s	RO
1015	**I-ID Input Status and terminator**	
	Contains the I-ID and terminator inputs logic state: Bit 0 : I-ID Bit 1 : terminator.	RO

Modbus Address	Description	
1100	**Digital Outputs Status**	
	Contains the digital outputs logic state (default 0): Bit 0: output 1 Bit 8: output 9 Bit 1: output 2 Bit 9: output 10 Bit 2: output 3 Bit 10: output 11 Bit 3: output 4 Bit 11: output 12 Bit 4: output 5 Bit 12: output 13 Bit 5: output 6 Bit 13: output 14 Bit 6: output 7 Bit 14: output 15 Bit 7: output 8 Bit 15: output 16	R/W
1101	**Pre-load Command encoder**	
	Determines how the encoder counter is set on 1103 and 1104 words value. (default 0) Bit 0: When this bit is set to 1, the encoder counter is immediately loaded with the preload value. This bit automatically returns to 0. Bit 1: Setting this bit to 1, the encoder counter is loaded with the preload value when the digital input 1 is set to logical value 1 (mark of zero). This bit automatically returns to 0.	R/W
1102	**Pre-load Command proximity 1, 2 and Q-ID**	
	Manages the proximity counter preload and Q-ID output status. (default 0) Bit 0: When this bit is set to 1, the proximity 1 counter is immediately set on 1105 and 1106 words value. This bit automatically returns to 0. Bit 1: When this bit is set to 1, the proximity 2 counter is immediately set on 1107 and 1108 words value. This bit automatically returns to 0. Bit 2: Q-ID output logic state.	R/W
1103	**Pre-loaded Value encoder H**	
	More significant word of the double-word which contains the encoder pre-loaded value. (Default 0)	R/W
1104	**Pre-loaded Value encoder L**	
	Less significant word of the double-word which contains the encoder pre-loaded value. (Default 0)	R/W

Modbus Address	Description	
1105	**Pre-loaded Value proximity 1 H**	
	More significant word of the double-word which contains the proximity 1 pre-loaded value. (Default 0)	R/W
1106	**Pre-loaded Value proximity 1 L**	
	Less significant word of the double-word which contains the proximity 1 pre-loaded value. (Default 0)	R/W
1107	**Pre-loaded Value proximity 2 H**	
	More significant word of the double-word which contains the proximity 2 pre-loaded value. (Default 0)	R/W
1108	**Pre-loaded Value proximity 2 L**	
	Less significant word of the double-word which contains the proximity 2 pre-loaded value. (Default 0)	R/W
1201… …1320	**Digital outputs status slave on the bus**	
	These words contain the digital outputs status of all slaves on the bus: according to address slave set (rotary-switch) device determines its reference word (ex. Slave1-word 1201… Slave 10-word 1210…) and sets the outputs by the value of the word. Used to set all digital outputs by broadcast writing on the bus. (Default 0).	R/W
2000	**Starting Digital Outputs Status**	
	Contains the digital outputs status when the device switches on. It's also digital outputs value if a communication error with Master node occurs. After off-line time (word 2002), outputs will change to value written on this word. Default configuration provides no output changes in event of communication error. (Default 0)	R/W
2001	**Encoder type selection**	
	Select the encoder type of counting: 0 → bidirectional encoder (default) 1 → unidirectional incremental encoder. 2 → unidirectional decremental encoder.	R/W
2002	**Off-line Time (0-60000)**	
	It's the maximun time (in ms) that can elapse between two Modbus messages without the slave sets outputs to word 2000 value. If 0, control is not performed. (Default 0).	R/W

Modbus Address	Description	
2003	**Digital inputs Filter (1-20)**	
	If the status of input changes, the new state is accepted only if it remains unchanged for the time set in this word. Accuracy is 0.5ms (1=0.5ms). (Default 1).	R/W
2004	**Modbus answer delay (0-250ms)**	
	Select modbus answer delay. (Default 0).	R/W
2005	**Modbus Baud-Rate**	
	Select modbus baud-rate (also set by rotary-switch): 0 → 2400 baud. 1 → 4800 baud. 2 → 9600 baud. 3 → 19200 baud. 4 → 28800 baud. 5 → 38400 baud. 6 → 57600 baud. (Default)	R/W
2006	**Address slave multiplier (0-7)**	
	Select address slave multiplier by serial (see MCM260 SET-UP). (Default 0)	R/W
2007	**Boot-up Time (0-200)**	
	Select the boot-up time. Accuracy is 10ms (1=10ms). (Default 2)	R/W
2102	**Access to Automatic Addressing Function**	
	To use automatic addressing function, Q-ID pin of a node has to be connected to I-ID pin of the following one. In the first device I-ID pin, in the last device Q-ID pin, won't be connected. To enable (disable) automatic addressing function to all connected device, value 1 (0) has to be written on this word in broadcast mode. Once address is assigned (see word below), to exit procedure write value 0 on this word, with slave address just saved, of course.	R/W
2103	**Address Slave Assignment**	
	To assign address write on this word password 1234: used address will be the one that slave will assign to itself.	R/W

5.4.2 MCM260-5AD

Modbus Address	Description	
0	**Device Type**	
	Contains the device identification code. (Default 165)	RO
1	**Firmware Version**	
	Contains the device firmware version.	RO
5	**Slave Address**	
	Contains the selected modbus slave address. (Depends on the position of the rotary-switch)	RO
900	**Cold Junction Temperature For AI1 and AI2**	
901	**Cold Junction Temperature For AI3 and AI4**	
	Contains the value of cold junction in tenths of degree	RO
1000	**Process Value AI1**	
1001	**Process Value AI2**	
1002	**Process Value AI3**	
1003	**Process Value AI4**	
	Contain process value. For temperature sensors the value is in tenths of degree; for the other sensors the value depends on the words 3005-3012.	RO
1100	**AO1 Value**	
1101	**AO2 Value**	
	Contain analogue outputs value. The current or voltage value of outputs depends on the values set on the word 4003-4006. (Defualt 0).	R/W
1102	**Q-ID Status**	
	Manages Q-ID output status. (default 0) Bit 0: Reserved Bit 1: Reserved Bit 2: Q-ID output status.	R/W
2002	**Off-line Time (0-60000)**	
	It's the maximun time (in ms) that can elapse between two Modbus messages without the slave sets outputs to word 2000 value. With 0 control is not performed. (Default 0).	R/W
2003	**Reserved**	
2004	**Delay Modbus Answer (0-250ms)**	
	Select modbus answer delay. (Default 0).	R/W

Modbus Address	Description	
2005	**Modbus Baud-Rate**	
	Select modbus baud-rate (also set by rotary-switch): 0 → 2400 baud. 1 → 4800 baud. 2 → 9600 baud. 3 → 19200 baud. 4 → 28800 baud. 5 → 38400 baud. 6 → 57600 baud. (Default)	R/W
2006	**Slave Address Multiplier (0-7)**	
	Select address slave multiplier by serial (see MCM260 SET-UP). (Default 0)	R/W
2007	**Boot-up Time (0-200)**	
	Select the boot-up time. Accuracy is 10ms (1=10ms). (Default 2)	R/W
2102	**Access to Automatic Addressing Function**	
	To use automatic addressing function, Q-ID pin of a node has to be connected to I-ID pin of the following one. In the first device I-ID pin, in the last device Q-ID pin, won't be connected. To able (disable) automatic addressing function to all connected device, value 1 (0) has to be written on this word in broadcast mode. Once address is assigned (see word below), to exit procedure write value 0 on this word, with slave address just saved, of course.	R/W
2103	**Address Slave Assignment**	
	To assign address write on this word password 1234: used address will be the one that slave will assign to itself.	R/W

Modbus Address	Description	
3001	**Analogue input configuration AI1**	
3002	**Analogue input configuration AI2**	
3003	**Analogue input configuration AI3**	
3004	**Analogue input configuration AI4**	
	These words define the type of sensor connected to analog inputs AI1…AI3. 0 → Disabled (default) 1 → Thermocouples K (range -260…1360℃) 2 → Thermocouples S (range -40…1760℃) 3 → Thermocouples R (range -40…1760℃) 4 → Thermocouples J (range -200…1200℃) 5 → PT100 (range -200…600℃) 6 → PT100 (range -200…140℃) 7 → NI100 (range -60…180℃) 8 → NTC 10K (β 3435K) (range -40…125℃) 9 → PTC1K (range -50…150℃) 10 → PT500 (range -100…600℃) 11 → PT1000 (range -100…600℃) 12 → Input 0…10V 13 → Input 0…20mA 14 → Input 4…20mA 15 → Input 0…40mV 16 → Potentiometer max. 6Kohm 17 → Potentiometer max. 150Kohm	R/W
3005	**Lower range limit AI1 only for linear input**	
3006	**Lower range limit AI2 only for linear input**	
3007	**Lower range limit AI3 only for linear input**	
3008	**Lower range limit AI4 only for linear input**	
	Minimum value of input if selected as linear Range -32768…32767. (Default 0).	R/W
3009	**Upper range limit AI1 only for linear input**	
3010	**Upper range limit AI2 only for linear input**	
3011	**Upper range limit AI3 only for linear input**	
3012	**Upper range limit AI4 only for linear input**	
	Maximum value of input if selected as linear Range -32768…32767. (Default 32767).	R/W

Modbus Address	Description/ meaning	
3013	**Offset Calibration for AI1**	
3014	**Offset Calibration for AI2**	
3015	**Offset Calibration for AI3**	
3016	**Offset Calibration for AI4**	
	Value of offset calibration for relevant input Range -1000…1000 (Default 0).	R/W
3017	**Gain Calibration for AI1**	
3018	**Gain Calibration for AI2**	
3019	**Gain Calibration for AI3**	
3020	**Gain Calibration for AI4**	
	Value of gain calibration for relevant input Range -1000…1000 (Default 0).	R/W
3021	**Filter for AI1**	
3022	**Filter for AI2**	
3023	**Filter for AI3**	
3024	**Filter for AI4**	
	Value of filter (=number of averages for the relevant input) Range 1…10 (Default 5).	R/W
3025	**Selection ℃ / ℉ for AI1**	
3026	**Selection ℃ / ℉ for AI2**	
3027	**Selection ℃ / ℉ for AI3**	
3028	**Selection ℃ / ℉ for AI4**	
	Type of degree when input is configured for temperature sensor 0 → Celsius (default) 1 → Fahrenheit	R/W

Modbus Address	Description/Meaning	
3201	**Frequency of sampling for AI1 and AI2**	
3202	**Frequency of sampling for AI3 and AI4**	
	0 → 242 Hz 1 → 123 Hz 2 → 62 Hz 3 → 50 Hz 4 → 39 Hz 5 → 33.2 Hz 6 → 19.6 Hz 7 → 16.7 Hz (Default) 8 → 12.5 Hz 9 → 10 Hz 10 → 8.33 Hz 11 → 6.25 Hz 12 → 4.17 Hz	R/W
4001	**Analogue output configuration AO1**	
4002	**Analogue output configuration AO2**	
	Type of anaolg output 0 → output 0…10V (Default) 1 → output 4…20mA	R/W
4003	**Lower limit range AO1**	
4004	**Lower limit range AO2**	
	Lower limit analog output Range -32768…32767. (Default 0).	R/W
4005	**Upper limit range AO1**	
4006	**Upper limit range AO2**	
	Upper limit analog output Range -32768…32767. (Default 32767).	R/W
4101	**Error Mode AO1**	
	Action on analog output AO1 in case of error 0 → output does not change 1 → output assumes the value selected on word 4102 (Default)	R/W

Modbus Address	Description/Meaning	
4102	**Error Value AO1**	
	Value assumed by analog output AO1 in case of error if 1 is selected on word 4101. Range -32768…32767. (Default 0).	R/W
4103	**Error Mode AO2**	
	Action on analog output AO2 in case of error 0 → output does not change 1 → output assumes the value selected on word 4104 (Default)	R/W
4104	**Error Value AO2**	
	Value assumed by analog output AO2 in case of error if 1 is selected on word 4103. Range -32768…32767. (Default 0).	R/W

6 CANOPEN

Each LED RUN blinking type indicates a specific CANopen status.

Blinking name LED RUN	Blinking type
Blink_fast	Fast blinking 50msec
Blink_medium	Blinking 200msec
Blink_slow	Blinking 600msec
LED_on	LED always on
Blink_3_on	LED on for 1sec, 3 blink 150msec
Blink_1_off	Slow blink 40msec every 1.2sec
Blink_3_off	LED off for 1sec, 3 blink 150msec

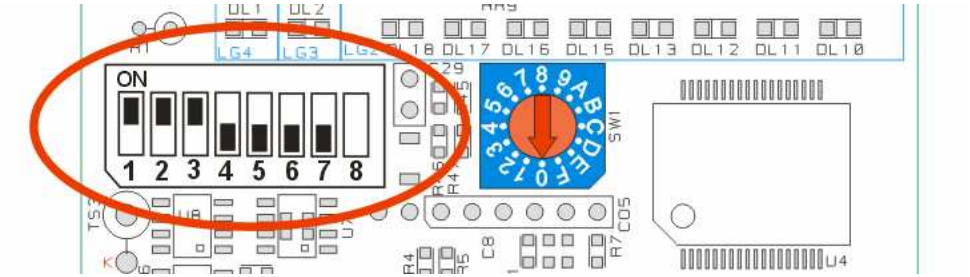
STATUS	LED RUN BLINKING
Boot-up	Blink_fast
Rotary switch in moving (during set-up)*	Blink_fast
Waiting for selection*	LED_off
Address offset multiplier saved*	Blink_3_on
Baud-rate saved*	LED_on
Pre-Operational	Blink_slow
Operational	LED_on
Stopped	LED_off
Pre-Operational with Emergency[2]	Blink_medium
Operational with Emergency[2]	Blink_3_on
Stopped with Emergency[2]	Blink_3_off

* They are not standard CANopen DS401 status, but particular phases of device configuration.

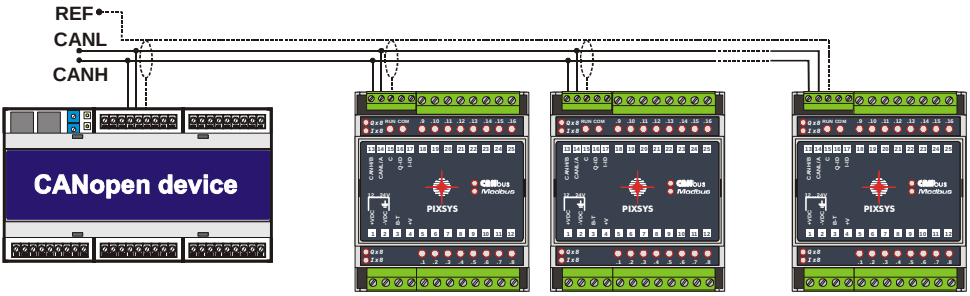
[2] For more details, see chapter 7

6.1 SET-UP as CANopen slave node

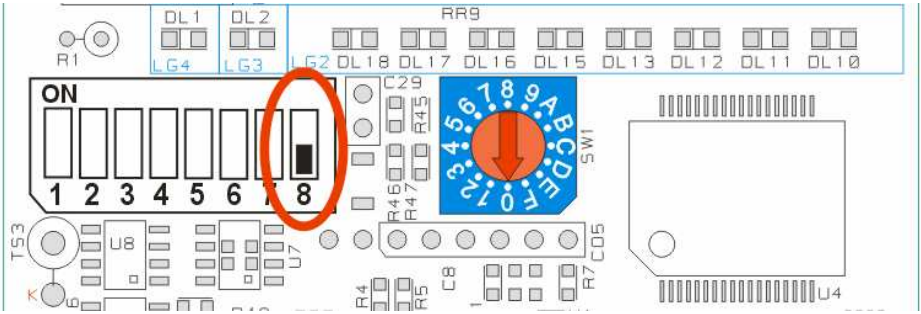
To configure MCM260 as a node of a CANopen network, dip-switch has to be set as shown in the figure below.



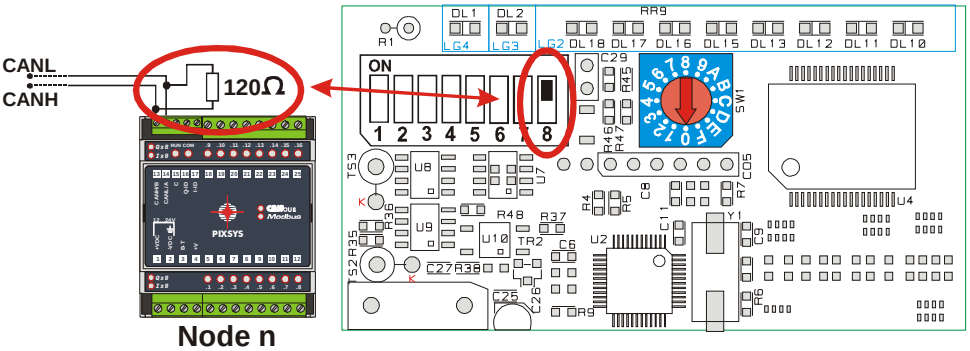
A CANopen network must have a 120Ω terminal line resistance. **If more devices have to be connected in cascade, that resistance has to be connected to the last MCM260 of network, at the end of the line only.**



If **no** terminal line resistance has to be connected, set last dip-switch as shown in figure below.



If terminal line resistance **has** to be connected (if MCM260 is the last device of the network), set last dip-switch as shown in figure below.



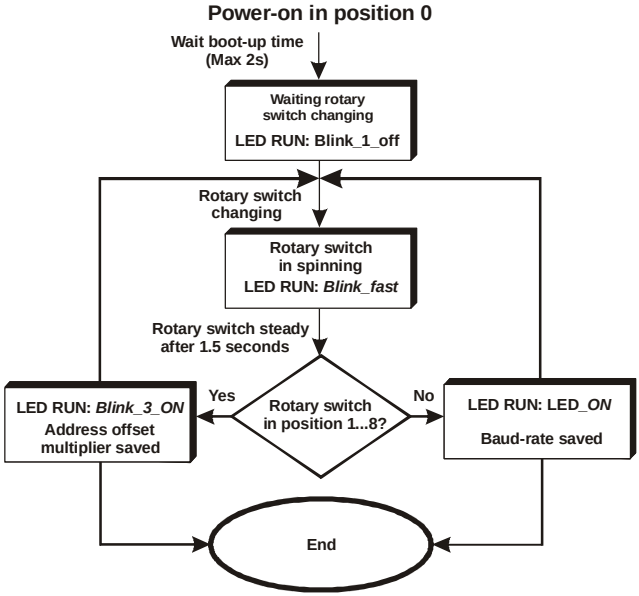
Subsequently, procedure shown in the following page has to be done to configure address offset multiplier and baud-rate.
Once time procedure is ended, device is not still ready to work: it will already be in SET-UP mode until it won't be powered-off. This isn't a standard CANopen DS401 status, this is specific device configuration.

6.2 Operation as CANopen slave node

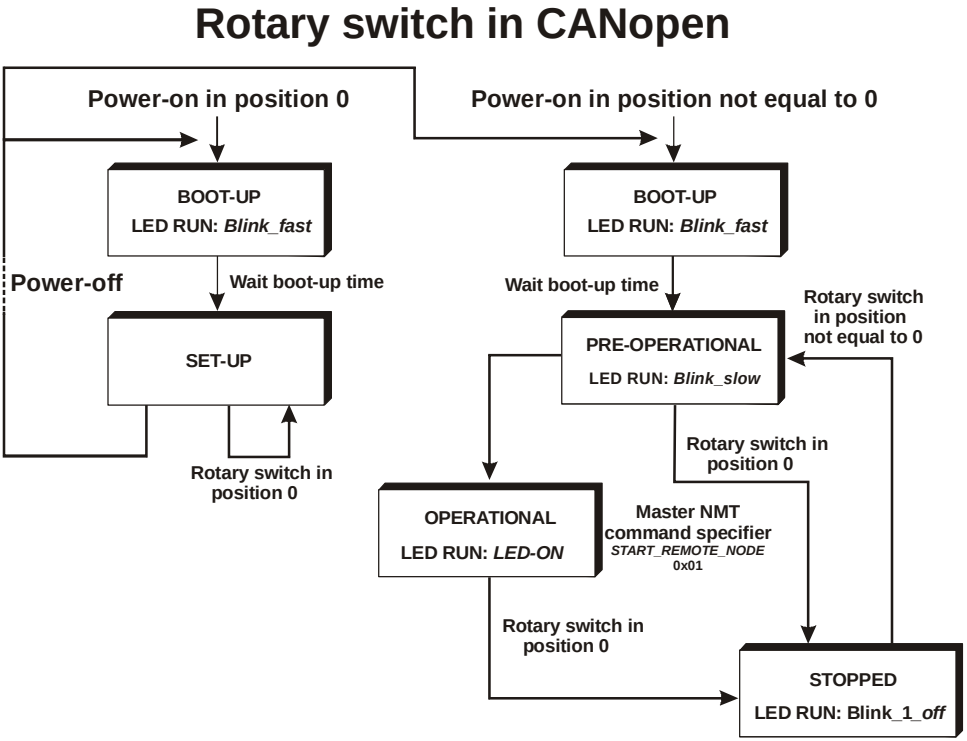
If dip-switch is configured as shown in figure in page 16, when MCM will be powered-on with rotary switch in position not equal to 0 (1...F), it will be ready to work as a CANopen slave node. Its baud-rate was selected during SET-UP, its address is calculated by actual rotary switch position added to offset selected during SET-UP.

After boot-up, device will be in Pre-Operational status automatically (LED RUN *Blink_slow* blinking). PDO transmission/reception aren't allowed, SDO transmission/reception are only allowed. To change into Operational status, an NMT[1] messages from a master is needed.
It's possible to change into Stopped status from Pre-Operational or Operational status by rounding rotary switch in position 0. Replacing switch in other positions, device will change into Pre-Operational status anyway.

SET-UP procedure for CANopen



[1] For more details about CANopen, see chapter 7



6.3 Object Dictionary

The following object dictionary table is divided in 3 parts. The first one is **Communication Profile Area**, where all parameters necessary for communication like setting identifier and PDO configuration are described. The second one is **Manufacturer Specific Parameter Area**, where all special manufacturer features are described. The third one is **Standard Device Profile Area**, where digital/Analogue input/output transmission type is described.

COMMUNICATION PROFILE AREA						
Index	**Sub-index**	**Name**	**Type**	**Default Value**	**Comment**	**R/W**
0x1000	0	Device type	32bit unsigned	0x00020191	MCM260-1AD	CONST
				0x00050191	MCM260-2AD	CONST
				0x00030191	MCM260-3AD	CONST
0x1001	0	Error register (all MCM260 series)	8bit unsigned	-	Emergency messages	R
0x1003	0	Pre-defined Error Field (all MCM260 series)	Array 8bit unsigned	-	Number of entries	R/W
	1		Array 32bit unsigned	-	Standard error field (always last error	R
	...		...	-		...
	10		Array 32bit unsigned	-	Standard error field (always first error	R
0x1005	0	COB-ID SYNC message (all MCM260 series)	32bit unsigned	0x00000080	COB-ID for SYNC message	R
0x1006	0	Communication Cycle Period (all MCM260 series)	32bit unsigned	0	Time between 2 SYNC messages	R/W
0x1008	0	Manufacturer Device Name (all MCM260 series)	String	M260	-	CONST
0x1009	0	Manufacturer Hardware Version (all MCM260 series)	String	Actual hardware version	-	CONST
0x100A	0	Manufacturer Software Version (all MCM260 series)	String	Actual software version	-	CONST
0x100B	0	Node ID (all MCM260 series)	8bit unsigned	0	Node address	R
0x100C	0	Guard Time (all MCM260 series)	16bit unsigned	0	Time between 2 Guard time interrogation	R/W
0x100D	0	Life Time Factor (all MCM260 series)	8bit unsigned	0	If its value is 0, no Node Guarding is monitored	R/W
0x1010	0	Store Parameters (all MCM260 series)	Array 8bit unsigned	1	Number of entries	R
	1		Array 32bit unsigned	1	Write string “save” to store user setting parameters	R/W

0x1011	0	Restore default Parameter (all MCM260 series)	Array 8bit unsigned	2	Number of entries	R
	1		Array 32bit unsigned	1	Write string “load” to restore default setting parameters	R/W
	2		Array 32bit unsigned	1	Write string “load” to restore default Pixsys parameters	
0x1014	0	COB-ID Emergency Object (all MCM260 series)	32bit unsigned	0x80 + module-ID	-	R
0x1015	0	Inhibit time Emergency Object (all MCM260 series)	16bit unsigned	0	Time have to be elapsed before sending another Emergency	R/W
0x1017	0	Producer Heartbeat Time (all MCM260 series)	16bit unsigned	0	Time between two heartbeat messages	R/W
0x1018	0	Identity Object (all MCM260 series)	Record 8bit unsigned	4	Number of entries	R
	1		Record 32bit unsigned	PIX	Manufacturer ID	R
	2		Record 32bit unsigned	260	Device description	R
	3		Record 32bit unsigned	-	Revision number	R
	4		Record 32bit unsigned	-	Serial number	R
0x1029	0	Error Behaviour (all MCM260 series)	Array 8bit unsigned	1	Number of entries	R
	1		Array 8bit unsigned	0	Communicatio n error	R/W
0x1400	0	Receive PDO communication parameter (except MCM260-2AD)	Record 8bit unsigned	2	Number of entries	R
	1		Record 32bit unsigned	0x1400 0x200 + module-ID	COB-ID used by PDO	R/W
	2		Record 32bit unsigned	255	Transmission type	R/W
0x1600	0	Receive PDO mapping parameter (except MCM260-2AD)	Record 8bit unsigned	-	Number of entries	R/W
	1		Record 32bit unsigned	-	PDO mapping object$_1$	R/W

	2		Record 32bit unsigned	-	PDO mapping object$_2$	R/W
0x1800	0	Transmit PDO communication parameter (except MCM260-1AD)	Record 8bit unsigned	5	Number of entries	R
	1		Record 32bit unsigned	0x1800 0x180 + module-ID	COB-ID used by PDO	R/W
	2		Record 8bit unsigned	255	Transmission type	R/W
	3		Record 16bit unsigned	0	Inhibit time	R/W
	5		Record 16bit unsigned	0	Event timer	R/W
0x1A00	0	Transmit PDO mapping parameter (except MCM260-1AD)	Record 8bit unsigned	-	Number of entries	R/W
	1...8		Record 32bit unsigned	-	PDO mapping object$_i$	R/W

MANUFACTURER SPECIFIC PARAMETER AREA MCM260-1AD, MCM260-2AD, MCM260-3AD, MCM260-4AD						
Index	**Sub-index**	**Name**	**Type**	**Default Value**	**Comment**	**R/W**
0x2000	0	Device specifications (all MCM260 series)	Array 16bit signed	10	Number of entries	R
	1		Array 16bit signed	6 (1Mbps)	Baud rate MCM260	R
	2		Array 16bit signed	0	Node-ID offset multiplier	R/W
	3		Array 16bit signed	50	Boot-up time	R/W
	4		Array 16bit signed	0x7F (Pre-Operational)	CANopen status after boot-up	R/W
	5		Array 16bit signed	1	Filter digital inputs	R/W
	6		Array 16bit signed	0	Pre-load Pixsys parameters	R/W
	7...10		Reserved			R/W
0x2100	0	Encoder counter (MCM260-2AD, MCM260-3AD)	Array 8bit signed	1	Number of entries	R
	1		Array 32bit signed	0	Encoder counter	R
0x2101	0	Preset encoder (MCM260-2AD, MCM260-3AD)	Record 8bit signed	1	Number of entries	R
	1		Record 32bit signed	0	Preset encoder	R/W
0x2102	0	Encoder command preset	Array 8bit unsigned	1	Number of entries	R

	1	(MCM260-2AD, MCM260-3AD)	Array 8bit unsigned	0	0→No preset is loaded 1→Preset value loaded simultaneously 2→Preset value loaded when DI=1 (zero inpulse)	R/W
0x2103	0	Encoder type (MCM260-2AD, MCM260-3AD)	Array 8bit signed	1	Number of entries	R
	1		Array 8bit signed	0	1→Mono-directional 2→Bi-directional	R/W
0x2104	0	Encoder speed (MCM260-2AD, MCM260-3AD)	Record 8bit signed	1	Number of entries	R
	1		Record 16bit signed	0	Encoder speed (100µsec)	R
0x2105	0	Encoder speed (MCM260-2AD, MCM260-3AD)	Record 8bit signed	1	Number of entries	R
	1		Record 16bit signed	0	Encoder speed (1sec)	
0x2200	0	Proximity counter (MCM260-2AD, MCM260-3AD)	Record 8bit signed	2	Number of entries	R
	1		Record 32bit signed	0	Proximity counter 1	R/W
	2		Record 32bit signed	0	Proximity counter 2	R/W
0x2201	0	Preset proximity (MCM260-2AD, MCM260-3AD)	Record 8bit signed	2	Number of entries	R
	1		Record 32bit signed	0	Preset proximity 1	R/W
	2		Record 32bit signed	0	Preset proximity 2	R/W
0x2202	0	Proximity command preset (MCM260-2AD, MCM260-3AD)	Array 8bit signed	2	Number of entries	R
	1		Array 8bit unsigned	0	Proximity 1 command preset	R/W
	2		Array 8bit unsigned	0	Proximity 2 command preset	R/W
0x2203	0	Timer ON proximity (MCM260-2AD, MCM260-3AD)	Record 8bit signed	2	Number of entries	R
	1		Record 16bit unsigned	0	Proximity 1 Timer ON	R
	2		Record 16bit unsigned	0	Proximity 2 Timer ON	R
0x2204	0	Proximity speed (MCM260-2AD, MCM260-3AD)	Record 8bit signed	1	Number of entries	R
	1		Record 16bit unsigned	0	Proximity 1 speed (100µsec)	R
	2		Record 16bit unsigned	0	Proximity 2 speed (100µsec)	R

0x2205	0	Proximity speed (MCM260-2AD, MCM260-3AD)	Record 8bit signed	1	Number of entries	R
	1		Record 16bit unsigned	0	Proximity 1 speed (1sec)	R
	2		Record 16bit unsigned	0	Proximity 2 speed (1sec)	R R

MANUFACTURER SPECIFIC PARAMETER AREA MCM260-5AD						
Index	**Sub-index**	**Name**	**Type**	**Default value**	**Comment**	**R/W**
0x2000	0	Device specifications	Array 16bit signed	10	Number of entries	R
	1		Array 16bit signed	6 (1Mbps)	Baud rate MCM260	R
	2		Array 16bit signed	0	Node-ID offset multiplier	R/W
	3		Array 16bit signed	50	Boot-up time	R/W
	4		Array 16bit signed	0x7F (Pre-Operational)	CANopen status after boot-up	R/W
	5		Array 16bit signed	0	Pre-load Pixsys parameters	R/W
	6...10		Reserved			R/W
0x4000 0x4001 0x4002 0x4003	0	AI1 parameters AI2 parameters AI3 parameters AI4 parameters	Array 16bit signed	20	Number of entries	R
	1		Array 16bit signed	0	Sensor type	R/W
	2		Array 16bit signed	0	Lower range limit only for linear input	R/W
	3		Array 16bit signed	0x7FFF	Upper range limit only for linear input	R/W
	4		Array 16bit signed	0	Offset calibration	R/W
	5		Array 16bit signed	0	Gain calibration	R/W
	6		Array 16bit signed	5	Filter	R/W
	7		Array 16bit signed	0	Degree type	R/W
	8...20		Reserved			R/W

0x4100	0	Input parameters	Array 16bit signed	10	Number of entries	R
	1		Array 16bit signed	7	Frequency of sampling for AI1 and AI2	R/W
	2		Array 16bit signed	7	Frequency of sampling for AI3 and AI4	R/W
	3...10		Reserved			R/W
0x5000 0x5001	0	AO1 parameters AO2 parameters	Array 16bit signed	10	Number of entries	R
	1		Array 16bit signed	0	Analogue output type	R/W
	2		Array 16bit signed	0	Lower limit range of analogue output	R/W
	3		Array 16bit signed	0x7FFF	Upper limit range of analogue output	R/W
	4...10		Reserved			R/W

STANDARD DEVICE PROFILE AREA						
Index	Sub-index	Name	Type	Default Value	Comment	R/W
0x6000	0	Digital input (MCM260-2AD, MCM260-3AD)	Array 8bit unsigned	-	Number of entries	R
	1		Array 8bit unsigned	-	1st input block	R
	2		Array 8bit unsigned	-	2nd input block (MCM260-2AD only)	R
0x6005	0	Global Interrupt Enable Digital 8bit (MCM260-2AD, MCM260-3AD)	8bit signed	1	Allows digital input transmission by PDO	R/W
0x6006	0	Interrupt Mask Any Change 8bit (MCM260-2AD, MCM260-3AD)	Array 8bit unsigned	-	Number of entries	R
	1		Array 8bit unsigned	255	Transmission channels 1...8 released in a event of a change	R/W
	2		Array 8bit unsigned	255	Transmission channels 9...16 released in a event of a change (MCM260-2AD only)	R/W

0x6007	**0**	**Interrupt Mask Low-to-High 8bit (MCM260-1AD, MCM260-3AD)**	**Array 8bit unsigned**	**-**	**Number of entries**	**R**
	1		**Array 8bit unsigned**	**0**	**Transmission channels 1...8 released in a event of a positive transition**	**R/W**
	2		**Array 8bit unsigned**	**0**	**Transmission channels 9...16 released in a event of a positive transition (MCM260-1AD only)**	**R/W**
0x6008	**0**	**Interrupt Mask High-to-Low 8bit (MCM260-1AD, MCM260-3AD)**	**Array 8bit unsigned**	**-**	**Number of entries**	**R**
	1		**Array 8bit unsigned**	**0**	**Transmission channels 1...8 released in a event of a negative transition**	**R/W**
	2		**Array 8bit unsigned**	**0**	**Transmission channels 9...16 released in a event of a negative transition (MCM260-1AD only)**	**R/W**
0x6200	**0**	**Digital Output (MCM260-1AD, MCM260-3AD)**	**Array 8bit unsigned**	**-**	**Number of entries**	**R**
	1		**Array 8bit unsigned**	**0**	**1st output block**	**R/W**
	2		**Array 8bit unsigned**	**0**	**2nd output block (MCM260-1AD only)**	**R/W**
0x6206	**0**	**Error Mode Output 8bit (MCM260-1AD, MCM260-3AD)**	**Array 8bit unsigned**	**-**	**Number of entries**	**R**
	1		**Array 8bit unsigned**	**255**	**1st output block (Output channels 1...8 change to a pre-defined error status in a event of an error)**	**R/W**
	2		**Array 8bit unsigned**	**255**	**2nd output block (MCM260-1AD only)**	**R/W**
0x6207	**0**	**Error Value Output 8bit (MCM260-1AD, MCM260-3AD)**	**Array 8bit unsigned**	**-**	**Number of entries**	**R**
	1		**Array 8bit unsigned**	**0**	**1st output block (Pre-defined error status in a event of an error)**	**R/W**
	2		**Array 8bit unsigned**	**0**	**2nd output block (MCM260-1AD only)**	**R/W**

0x6401	**0**	**Analogue Input 16bit (MCM260-2AD, MCM260-5AD)**	**Array 8bit unsigned**	**2(-2AD) 4(-5AD)**	**Number of Analogue input channels**	**R**
	1		**Array 16bit unsigned**	**-**	**1st channel**	**R**
	2		**Array 16bit unsigned**	**-**	**2nd channel**	**R**
	3		**Array 16bit unsigned**	**-**	**3rd channel**	**R**
	4		**Array 16bit unsigned**	**-**	**4th channel**	**R**
0x6411	**0**	**Analogue Output 16bit (MCM260-5AD)**	**Array 8bit unsigned**	**2**	**Number of Analogue output channels**	**R**
	1		**Array 16bit unsigned**	**-**	**1st channel**	**R/W**
	2		**Array 16bit unsigned**	**-**	**2nd channel**	**R/W**
0x6421	**0**	**Analogue Input Interrupt Trigger Selection (MCM260-2AD, MCM260-5AD)**	**Array 8bit unsigned**	**2(-2AD) 4(-5AD)**	**Number of Analogue input channels**	**R**
	1		**Array 8bit unsigned**	**7**	**Trigger 1st channel (transmission disabled)**	**R/W**
	2		**Array 8bit unsigned**	**7**	**Trigger 2nd channel (tx disabled)**	**R/W**
	3		**Array 8bit unsigned**	**7**	**Trigger 3rd channel (tx disabled)**	**R/W**
	4		**Array 8bit unsigned**	**7**	**Trigger 4th channel (tx disabled)**	**R/W**
0x6423	**0**	**Analogue Input Global Interrupt Enable (MCM260-2AD, MCM260-5AD)**	**Boolean**	**0**	**Enable/Disable Analogue input transmission**	**R/W**
0x6424	**0**	**Analogue Input Interrupt Upper Limit Integer (MCM260-2AD only)**	**Array 8bit unsigned**	**2(-2AD) 4(-5AD)**	**Number of Analogue input channels**	**R**
	1		**Array 16bit unsigned**	**0**	**Upper limit 1st ch. (Analogue input transmission monitored by threshold value)**	**R/W**
	2		**Array 16bit unsigned**	**0**	**Upper limit 2nd ch.**	**R/W**
	3		**Array 16bit unsigned**	**0**	**Upper limit 3rd ch.**	**R/W**
	4		**Array 16bit unsigned**	**0**	**Upper limit 4th ch.**	**R/W**

0x6425	0	Analogue Input Interrupt Lower Limit Integer (MCM260-2AD, MCM260-5AD)	Array 8bit unsigned	2(-2AD) 4(-5AD)	Number of Analogue input channels	R
	1		Array 16bit unsigned	0	Lower limit 1st channel (Analogue input transmission monitored by threshold value)	R/W
	2		Array 16bit unsigned	0	Lower limit 2nd channel	R/W
	3		Array 16bit unsigned	0	Lower limit 3rd channel	R/W
	4		Array 16bit unsigned	0	Lower limit 4th channel	R/W
0x6426	0	Analogue Input Interrupt Delta Unsigned (MCM260-2AD, MCM260-5AD)	Array 8bit unsigned	2(-2AD) 4(-5AD)	Number of Analogue input channels	R
	1		Array 16bit unsigned	0	Delta 1st channel (actual transmission conditioned by previously sent value ±Delta)	R/W
	2		Array 16bit unsigned	0	Delta 2nd channel	R/W
	3		Array 16bit unsigned	0	Delta 3rd channel	R/W
	4		Array 16bit unsigned	0	Delta 4th channel	R/W
0x6427	0	Analogue Input Interrupt Negative Delta Unsigned (MCM260-2AD, MCM260-5AD)	Array 8bit unsigned	2(-2AD) 4(-5AD)	Number of Analogue input channels	R
	1		Array 16bit unsigned	0	Delta 1st channel (actual transmission conditioned by previously sent value -Delta)	R/W
	2		Array 16bit unsigned	0	Delta 2nd channel	R/W
	3		Array 16bit unsigned	0	Delta 3rd channel	R/W
	4		Array 16bit unsigned	0	Delta 4th channel	R/W

0x6428	**0**	**Analogue Input Interrupt Positive Delta Unsigned (MCM260-2AD, MCM260-5AD)**	**Array 8bit unsigned**	**2(-2AD) 4(-5AD)**	**Number of Analogue input channels**	**R**
	1		**Array 16bit unsigned**	**0**	**Delta 1st channel (actual transmission conditioned by previously sent value + Delta)**	**R/W**
	2		**Array 16bit unsigned**	**0**	**Delta 2nd channel**	**R/W**
	3		**Array 16bit unsigned**	**0**	**Delta 3rd channel**	**R/W**
	4		**Array 16bit unsigned**	**0**	**Delta 4th channel**	**R/W**
0x6443	**0**	**Analogue Output Error Mode (MCM260-5AD)**	**Array 8bit unsigned**	**2**	**Number of Analogue output channels**	**R**
	1		**Array 16bit unsigned**	**1**	**Error Mode 1st channel**	**R/W**
	2		**Array 16bit unsigned**	**1**	**Error Mode 2nd channel**	**R/W**
0x6444	**0**	**Analogue Output Error Value Integer (MCM260-5AD)**	**Array 8bit unsigned**	**2**	**Number of Analogue output channels**	**R**
	1		**Array 16bit unsigned**	**1**	**Error Value 1st channel**	**R/W**
	2		**Array 16bit unsigned**	**1**	**Error Value 2nd channel**	**R/W**
0x67FE	**0**	**Error Behaviour (all MCM260 series)**	**Array 8bit unsigned**	**1**	**Number of entries**	**R**
	1		**Array 8bit unsigned**	**0**	**Communication error (see object 0x1029)**	**R/W**

6.4 EDS files

EDS files of the various models are available for download at http://download.pixsys.com/

7 CANOPEN IN DETAILS

CAN (Controller Area Network) is a Multimaster bus system. Messages are sent to the bus with a certain priority, defined by COB ID (Communication Object Identifier).
CANopen is a networking concept defined as an application layer by DS 301 CIA specification (CAN in automation).

CANopen is built on top of CAL (CAN Application Layer, an high layer communication protocol for CAN-based network). CAL defines 4 application layer service elements:

1. **CMS** (CAN-based Message Specification): it defines a set of objects (Variable, Event, Domain) to specify how CAN interface can access to the network node features.
2. **NMT** (Network Management): it defines all typical services of a master-slave concept network as initialisation, start and stop node, detection of failures.
3. **DBT** (Distributor): it defines a dynamic distribution of CAN identifiers to the nodes of the network, called **COB-ID** (Communication Object Identifier)
4. **LMT** (Layer Management): it offers the possibility to change parameters as NMT address of a node, bit-timing and baud rate of CAN interface.

CMS defines 8 priority levels, 220 **COB-ID** each. Others identifiers are reserved for NMT, DBT and LMT.

CAN Application Layer (CAL)	
COB-ID	**Description**
0	NMT start/stop services
1...220	CMS object priority 0
221...440	CMS object priority 1
441...660	CMS object priority 2
661...880	CMS object priority 3
881...1100	CMS object priority 4
1101...1320	CMS object priority 5
1321...1540	CMS object priority 6
1541...1760	CMS object priority 7
1761...2015	NMT Node Guarding
2016...2031	NMT, LMT, DBT services

CAL doesn't define contents of the **CMS** objects, it defines how but not what. **CANopen** provides an implementation of a distributed control system using **CAL** services and protocols.

7.1 Object Dictionary

The object dictionary is the most important point of a CANopen device where all configuration information and data are stored. It is an ordered group of objects, where each one is addressed using a 16bit index. Organization of dictionary is based on tables and contains three areas of CANopen objects:

Communication Profile Area (Index 0x1000...0x1FFF): this profile contains all parameters relevant for CANopen communication and it is common for all CANopen devices.

Manufacturer Specific Profile Area (Index 0x2000...0x5FFF): in this profile, each manufacturer can implement its own company specific objects.

Standardized Device Profile Area (Index 0x6000...0x9FFF): this profile contains all objects which are assisted by a certain device profile. The bus coupler assists the device profile DS-401 (Device Profile for Generic I/O Modules)

In the object dictionary, a logical addressing scheme is used for the access to communication and device parameters, data and functions. Each entry into the dictionary is identified by a 16 bit index which indicates the row address of the table. A maximum of 65536 entries are permitted.
If an object is composed of several components, the components are identified by means of an 8 bit sub-index. The sub-index indicates the individual column address of the table allowing a maximum of 256 entries.
If index only consists of simple variables (8bit unsigned, 16bit unsigned, ecc.), sub-index is always zero.
For more objects, as array, records, ecc. sub-index 0 indicates the maximum number of the following sub-indexes.

Data is coded in the following sub-indexes:

- object name describing the object function
- a data type attribute
- an access attribute: read only, write only or read/write

CANopen object dictionary structure	
Index (Exadecimal)	**Object**
0x0000	**Not used**
0x0001- 0x001F	**Static data types**
0x0020 - 0x003F	**Complex data types**
0x0040 - 0x005F	**Manufacturer specific data types**
0x0060 - 0x007F	**Profile specific static data types**
0x0080 - 0x009F	**Profile specific complex data types**
0x00A0 - 0x0FFF	**Reserved**
0x1000 - 0x1FFF	**Communication Profile (DS-301)**
0x2000 - 0x5FFF	**Manufacturer specific parameters**
0x6000 - 0x9FFF	**Parameters from standardized device profiles**
0xA000 - 0xFFFF	**Reserved**

7.1.1 CANopen communication model

CANopen defines 4 message types:

1. **Administrative message**: Layer management, network management and identifier distribution services (initialisation, configuration and supervision network). Services and protocols are according to LMT, NMT and DBT elements.

2. **Service Data Object (SDO)**: it provides client access to objects of object dictionary of the device (server) using index and subindex. A replay is generated for every CAN message: one SDO requires 2 CAN-identifiers. SDO request and reply message always contains 8 bytes.

3. **Process Data Object (PDO)**: it provides transfer real-time data. Data transfer is limited from 1 to 8 bytes, and it's content is defined by its CAN-identifier only. Each PDO is described by 2 object in the object dictionary:
   - **PDO Communication Parameter**: it contains COB-ID used, transmission type, inhibit time and time period
   - **PDO Mapping Parameter**: it contains a list of entries of object dictionary mapped in PDO. It's configurable using SDO messages if "variable PDO mapping" is supported by devices.

   There are 2 types of PDO transmission:
   1. **Synchronous**: it's regulated by receipt of a SYNC object (**acyclic**, means not periodically, or **cyclic**, means that transmission is periodically triggered every 1,2,...,240 by SYNC messages).
   2. **Asynchronous**: transmission is triggered by a remote transmission request from another device, or it's triggered by configuration of an object specific event specified in the device profile (input change of value, or a timer event...)

   Inhibit time for a PDO defines minimum time between two consecutive PDO transmission. It's a part of PDO Communication Parameter and it's defined as an unsigned 16bit integer (unit is 100μsec).
   Event time period for a PDO defines where PDO transmission is periodically triggered when a specific time has elapsed. It's defined as an unsigned 16bit integer (unit is millisecond).
   PDO transmit data without overhead and messages aren't confirmed: one PDO requires one CAN-identifier (no more than 8 bytes can be transferred with 1 PDO).

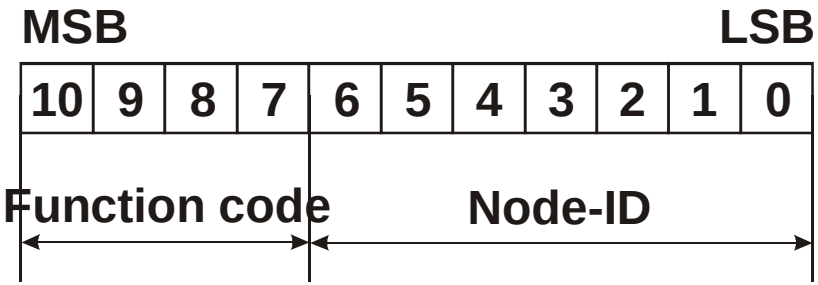
4. **Predefined Messages** or **Special Function Objects**. It's a list of pre-defined and important messages:
   Synchronization (SYNC): it regulates input/output transmission and update through PDO synchronization. It is in the highest priority COB-ID and no data bytes are transferred to ensure message as short as possible.
   Time Stamp: It provides application devices a common time frame reference.

Emergency: The event is triggered by device internal errors.
Node/Life Guarding: NMT master monitors nodes status (node guarding). Nodes optionally monitor NMT master status (life guarding): it starts on the NMT slave after it has received the first node guarding message from NMT master. It detects errors in the network interfaces of devices: a remote transmission request from NMT master to a particular node triggers a reply containing node status.
Boot-up: NMT slave send this message when it has transitioned from **Initialising** to **Pre-Operational** status.

Finally, SDO is typically used for device configuration in a CANopen network, while PDO is used for fast data transfer. All CANopen devices should have at least one PDO, all other communication objects are optional.

7.1.2 CANopen Pre-defined Connection Set

When a device has to reply a master request, a default CAN-identifier scheme is used. It's based on a 11bit frame, where the first 7bit (LSB) are used for **Node-ID** (range 1...127, defined by specific manufacturer configuration), and the last 4bit (MSB) are used for **Function Code**.



Pre-defined connection set defines 4 Rx PDOs, 4 TX PDOs, 1 SDO, 1 Emergency Object and 1 Node-Error-Control Identifier[1]. It also support broadcasting of non-confirmed NMT Module Control Services, SYNC and

[1] **MCM260** series has only 1 TX PDO and 1 RX PDO

Time Stamp objects. Complete CAN-identifier scheme is shown below:

Broadcast objects of CANopen Pre-defined Connection Set			
Object	**Function Code (bit 7...10)**	**COB-ID**	**Communication parameters**
NMT Module Control	0000	0x000	-
SYNC	0001	0x080	0x1005, 0x1006, 0x1007
Time Stamp	0010	0x100	0x1012, 0x1013

Peer-to-Peer objects of CANopen Pre-defined Connection Set			
Object	**Function Code (bit 7...10)**	**COB-ID**	**Communication parameters**
Emergency	0000	0x81 – 0xFF	0x1024, 0x1015
PDO1 (transmit)	0011	0x181 – 0x1FF	0x1800
PDO1 (receive)	0100	0x201 – 0x27F	0x1400
PDO2 (transmit)	0101	0x281 – 0x2FF	0x1801
PDO2 (receive)	0110	0x301 – 0x37F	0x1401
PDO3 (transmit)	0111	0x381 – 0x3FF	0x1802
PDO3 (receive)	1000	0x401 – 0x47F	0x1402
PDO4 (transmit)	1001	0x481 – 0x4FF	0x1803
PDO4 (receive)	1010	0x501 – 0x57F	0x1403
SDO (transmit/receive)	1011	0x581 – 0x5FF	0x1200
SDO (receive/client)	1100	0x601 – 0x67F	0x1200
NMT Error Control	1110	0x701 – 0x77F	0x1016, 0x1017

All peer-to-peer identifiers are different so only one master device can communicate to each slave node (up to 127 nodes). Two slaves aren't able to communicate because they don't know each other's node-ID, master only knows them.

7.1.3 CANopen identifier distribution

COB-ID's allocation can be made in 3 ways:

- Pre-defined Connection Set: it's mode shown in the previous section. Allocation is default setting, no configuration is needed.

- PDO identifiers (COB-IDs) can be modified after powering-up device, when it's in **Pre-Operationa**l status (see next section). In this status, it's possible to write new values to Object Dictionary allocation by SDO.
- Using DBT (Distributor, a CAL service): nodes are initially identified by their configured node-ID. Node-ID of slave nodes may be configured by internal dip-switches or LMT (Layer Management, a CAL service).
  When network initialises and boots, master provides a connection with each connected slave with a 'telegram' (NMT service). Once this connection has been established, DBT provides allocation of CAN identifiers for communication of SDOs and PDOs to the node.

7.1.4 CANopen boot-up process

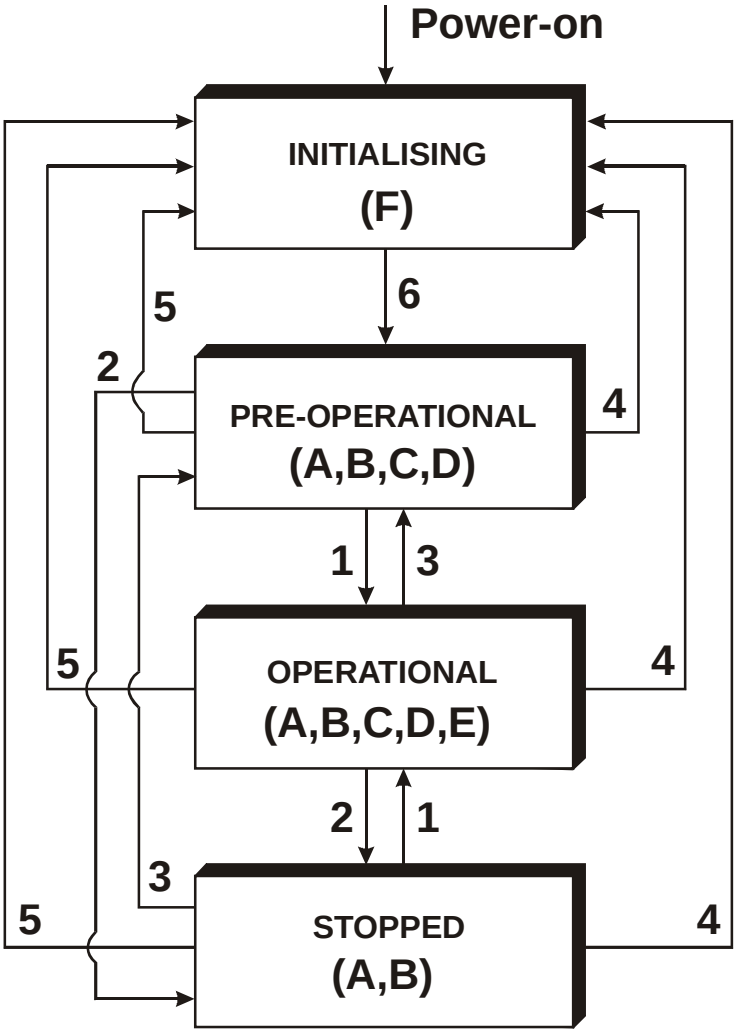
Networks initialisation supports two boot-up process: **Minimum boot-up** and **Extended boot-up**. The first one is a pre-required process for a CANopen device, the second one is optional, but necessary if COB-IDs have to be allocated by DBT services[1].
State-transition diagram in next page shows a CANopen minimum boot-up node.

Letters in brackets show which communication object types are allowed in the different status:
A = NMT
B = Node Guard
C = SDO
D = Emergency
E = PDO
F = Boot-up

Numbers show Status transitions and NMT command specifier:
1 = Start Remote Node (0x01)
2 = Stop Remote Node (0x02)
3 = Enter Pre-Operational status (0x80)

[1] **MCM260** series only provides **Minimum boot-up**

4 = Reset Node (0x81)
5 = Reset Communication (0x82)
6 = Device initialisation is finished, device changes to Pre-Operational status automatically and boot-up message is send



NMT services allow changing status nodes at any time. NMT message consists of CAN-header (COB-ID = 0) and 2 data bytes. One byte

contains the requested service (NMT command specifier) and other byte contains Node-ID (0 for broadcasting mode).
A CANopen network can only have one NMT master, which brings NMT messages and controls initialisation process.
CANopen device supporting only minimum boot-up enter Pre-Operational status automatically after ending device initialisation. In this status setting device parameters and COB-ID allocation are possible by SDO only.

A distinction is needed between device configuration and CANopen device initialisation (**boot-up**). **MCM260** configuration is allowed when device is powered-on with rotary switch in position 0 (Node-ID offset multiplier, baud-rate); **MCM260** CANopen initialisation (**boot-up**) is always provided by itself when it is powered-on (for each position of rotary switch).
MCM260 enter Pre-Operational status after having finished its initialisation. It's possible to force Stopped status by changing rotary switch in 0 position (from Pre-Operational or Operational status).

7.1.5 Communication Profile: Initialisation

In the most common cases, a default configuration is assigned to Object dictionary, if there aren't other configuration saved by user.
All object are initialised with default DS301 values (CANopen Application Layer and Communication Profile). Rx/Tx PDO have default values.

Rx PDO: it contains the first 2x8 digital outputs. If there aren't digital outputs, subindex 0 is 0 (default value) and PDO isn't used with its default value

Index	Subindex	Description	Default value
0x1600	**0**	Number of objects	**0**: no digital output block **1…2**: 1…2 digital output blocks
	1	1st mapped digital output block	**0x6200 01 08**
	2	2nd mapped digital output block	**0x6200 02 08**

Tx PDO: it contains the first 2x8 digital inputs and 2 Analogue inputs. If there aren't digital inputs, subindex 0 is 0 (default value) and PDO isn't used with its default value

Index	Subindex	Description	Default value
0x1A00	**0**	Number of objects	**0**: no digital input block **1…8**: 1…8 digital input blocks
	1	1st mapped digital input block	**0x6000 01 08**
	2	2nd mapped digital input block	**0x6000 02 08**
	3	1st mapped Analogue input	**0x6401 01 10**
	4	2nd mapped Analogue input	**0x6401 02 10**
	5...8	...	**...**

7.2 Communication Profile Area

The following table shows all the supported communication profile objects:

N.B. **ALL** = All devices
O_IO = Output devices (**MCM260-1AD**), Input/Output devices (**MCM260-3AD, MCM260-4AD**), analogue device (**MCM260-5AD**)
I_IO = Input devices (**MCM260-2AD**), Input/Output devices (**MCM260-3AD, MCM260-4AD**), analogue device (**MCM260-5AD**)

Index	Name	Type	Device Type	R/W
0x1000	Device type	32bit unsigned	**ALL**	**CONST**
0x1001	Error register	8bit unsigned	**ALL**	**R**
0x1003	Pre-defined Error Field	Array 32bit unsigned	**ALL**	**R/W**
0x1005	COB-ID SYNC message	32bit unsigned	**ALL**	**R**

0x1006	Communication Cycle Period	32bit unsigned	**ALL**	**R/W**
0x1008	Manufacturer Device Name	String	**ALL**	**CONST**
0x1009	Manufacturer Hardware Version	String	**ALL**	**CONST**
0x100A	Manufacturer Software Version	String	**ALL**	**CONST**
0x100B	Node ID	8bit unsigned	**ALL**	**R**
0x100C	Guard Time	16bit unsigned	**ALL**	**R/W**
0x100D	Life Time Factor	8bit unsigned	**ALL**	**R/W**
0x1010	Store Parameters	Array 32bit unsigned	**ALL**	**R/W**
0x1011	Restore default Parameter	Array 32bit unsigned	**ALL**	**R/W**
0x1014	COB-ID Emergency Object	32bit unsigned	**ALL**	**R**
0x1015	Inhibit time Emergency Object	16bit unsigned	**ALL**	**R/W**
0x1017	Producer Heartbeat Time	16bit unsigned	**ALL**	**R/W**
0x1018	Identity Object	Record 32bit unsigned	**ALL**	**R**
0x1029	Error Behaviour	Array 8bit unsigned	**ALL**	**R/W**
0x1400	Receive PDO communication parameter	Record 32bit unsigned	**O_IO**	**R/W**
0x1600	Receive PDO mapping parameter	Record 32bit unsigned	**O_IO**	**R/W**
0x1800	Transmit PDO communication parameter	Record 32bit unsigned	**I_IO**	**R/W**

0x1A00	Transmit PDO mapping parameter	Record 32bit unsigned	**I_IO**	**R/W**

7.2.1 Device Type

This object indicates device type profile:

Index	Subindex	Name	Type	Default	R/W
0x1000	0	Device type	32bit unsigned	-	CONST

Structure:

Bit 24…31 MSB	Bit 16…23	Bit 8...15	Bit 0...7 LSB
0x00	**0000b_{19} b_{18} b_{17} b_{16}**	**0x01**	**0x91**

b_{16}	**0**	If no **digital input** is connected
	1	If at least one **digital input** is connected
b_{17}	**0**	If no **digital output** is connected
	1	If at least one **digital output** is connected
b_{18}	**0**	If no **Analogue input** is connected
	1	If at least one **Analogue input** is connected
b_{19}	**0**	If no **Analogue output** is connected
	1	If at least one **Analogue output** is connected

Example: For MCM260 series:

Device Name	Attribute	Device Type Profile
MCM260-1AD	16 Digital Output	**0x00020191**
MCM260-2AD	16 Digital Input + 2 Analogue Input 0...10V	**0x00050191**
MCM260-3AD **MCM260-4AD**	8 Digital Input + 8 Digital Output	**0x00030191**
MCM260-5AD	4 Analogue Input + 2 Analogue output	**0x000C0191**

Least significant word (LSW) is always **0x0191** = **401_{dec}** CAN DS standard.

7.2.2 Error Register

This object contains internal errors and it's a subset of emergency message

Index	Subindex	Name	Type	Default	R/W
0x1001	**0**	Error register	8bit unsigned	**-**	R

Structure:

Bit number	Meaning
0	General error
1	Current
2	Voltage
3	Temperature
4	Communication
5	Device profile specific
6	Reserved
7	Manufacturer specific

If an error occurs, bit 0 is always set 1.

7.2.3 Pre-defined Error Field

This object contains information about last 10 errors occurred. New errors will be entered in subindex 1, and error in subindex 10 will be lost.

Index	Subindex	Name	Type	Default	R/W
0x1003	**0**	Number of entries	Array 8bit unsigned	**-**	R/W
	1	Standard error field (always last error)	Array 32bit unsigned	**-**	R
	...	...	...	**-**	...

	10	Standard error field (first error)	Array 32bit unsigned	**-**	R

Structure:

Bit 16…31 MSW	Bit 0...15 LSW
Additional info	Error code

Additional info are first 2 bytes of additional code of Emergency telegram. Error code is error code in the Emergency telegram.

7.2.4 COB-ID SYNC message

This object contains COB-ID for synchronization message.

Index	Subindex	Name	Type	Default	R/W
0x1005	**0**	COB-ID SYNC	32bit unsigned	**0x00000080**	R

Structure:

Bit 11…31 MSW	Bit 0...10
0 (reserved)	**COB-ID**

7.2.5 Communication Cycle Period

This object contains maximum time (µsec) between 2 SYNC messages (resolution 2msec). If value is 0, no SYNC monitoring is realized.

Index	Subindex	Name	Type	Default	R/W
0x1006	**0**	Communication Cycle Period	32bit unsigned	**0**	R/W

7.2.6 Manufacturer Device Name

Index	Subindex	Name	Type	Default	R/W
0x1008	**0**	Manufacturer Device Name	String	**M260**	CONST

7.2.7 Manufacturer Hardware Version

Index	Subindex	Name	Type	Default	R/W

0x1009	0	Manufacturer Hardware Version	String	**Actual hardware version**	CONST

7.2.8 Manufacturer Software Version

Index	Subindex	Name	Type	Default	R/W
0x100A	**0**	Manufacturer Software Version	String	Actual software version	CONST

7.2.9 Node ID

Index	Subindex	Name	Type	Default	R/W
0x100B	**0**	Node ID	8bit unsigned	**0**	R

7.2.10 Guard Time

This object defines Guarding Time (time between 2 interrogation, in msec)

Index	Subindex	Name	Type	Default	R/W
0x100C	**0**	Guard Time	16bit unsigned	**0**	R/W

7.2.11 Life Time Factor

This object is a part of Node Guarding Protocol. If it's equal to 0, no monitoring is realized.

Index	Subindex	Name	Type	Default	R/W
0x100D	**0**	Life Time Factor	8bit unsigned	0	R/W

7.2.12 Store Parameters

This object stores user's setting parameters permanently, if signature "save" (ASCII 0x65766173) is written into subindex 1

Index	Subindex	Name	Type	Default	R/W
0x1010	**0**	Number of entries (1)	Array 8bit unsigned	**1**	R

	1	Store all parameters	Array 32bit unsigned	**1 (string "save" to store)**	R/W

7.2.13 Restore Default Parameters

This object allows to reset user stored parameters and load default values. If signature "load" (ASCII 0x64616F6C) is written on Sub-index 1, standard default parameters are loaded each power on (until next "save" command is written). This is the same behaviour for Pixsys default parameters on Sub-index 2.

Index	Subindex	Name	Type	Default	R/W
0x1011	**0**	Number of entries (1)	Array 8bit unsigned	**2**	R
	1	Load standard default parameters	Array 32bit unsigned	**1 (string "load" for standard default)**	R/W
	2	Load Pixsys default parameters	Array 32bit unsigned	**1 (string "load" for Pixsys default)**	R/W

7.2.14 COB-ID Emergency Object

Index	Subindex	Name	Type	Default	R/W
0x1014	**0**	COB-ID EMCY	32bit unsigned	**0x80+module-ID**	R

Structure:

Bit 31	Bit 11...30	Bit 0...10
0(valid) / **1**(invalid)	**0** Reserved	**COB-ID**

7.2.15 Inhibit Time Emergency Object

This object indicates time have to be elapsed before sending another Emergency (minutes).

Index	Subindex	Name	Type	Default	R/W
0x1015	**0**	Inhibit Time EMCY	16bit unsigned	**0**	R/W

7.2.16 Producer Heartbeat Time

This object contains time between two Heartbeat messages (milliseconds). If it's equal to 0, no Heartbeat is sent.

Index	Subindex	Name	Type	Default	R/W
0x1017	**0**	Producer Heartbeat Time	16bit unsigned	**0**	R/W

7.2.17 Identity Object

This object specifies device and manufacturer.

Index	Subindex	Name	Type	Default	R/W
0x1018	**0**	Number of entries	Record 8bit unsigned	**4**	R
	1	Manufacturer ID	Record 32bit unsigned	**PIX**	R
	2	Device description	Record 32bit unsigned	**260**	R
	3	Revision number	Record 32bit unsigned	**-**	R
	4	Serial number	Record 32bit unsigned	**-**	R

7.2.18 Error Behaviour

This object specifies into which states modules changes in case of communication error.

Index	Subindex	Name	Type	Default	R/W
0x1029	**0**	Number of entries (1)	Array 8bit unsigned	**1**	R
	1	Communication error	Array 8bit unsigned	**0**	R/W

Structure:

Communication error	Action
0	Change into status PRE-OPERATIONAL (only if actual status were OPERATIONAL)
1	No status change
2	Change into status STOPPED

7.2.19 Receive PDO Communication Parameter

This object set communication parameters of Rx PDO supported. Default COB-ID of PDO is set by DS301 standard.

Index	Subindex	Name	Type	Default	R/W
0x1400	**0**	Number of entries	Record 8bit unsigned	**2**	R
	1	COB-ID used by PDO1	Record 32bit unsigned	**0x1400 0x200 + Module-ID**	R/W
	2	Transmission type	Record 8bit unsigned	**255**	R/W

COB-ID Structure:

Bit 31	Bit 30	Bit 29...11	Bit 0...10
0(valid) / **1**(invalid)	**0**(RTR allowed) / **1**(RTR not allowed)	**0** Reserved	**COB-ID**

Digital and Analogue inputs are transmitted as Change Of Value (COV). Type of transmission depending upon set transmission type and it's explained in the following table (RTR = Remote Transmission Request received):

Trans-mission type	PDO transmission						
	cyclic	acyclic	synchro -nous	asynchro -nous	RTR only	TxPDO (inputs)	RxPDO (outputs)
0		X	X			If COV is transmitted with each SYNC	Set outputs after each SYNC as requested by last PDO received
1...240	X		X			Transmission with each i SYNC (i = 1...240)	Set outputs after each SYNC as requested by last PDO received
241..251	Reserved						

252			X		X	Data is read again with a SYNC, but not sent, request via RTR	Not supported
253				X	X	Request via RTR	COV
254				X		COV	COV
255				X		COV	COV

7.2.20 Receive PDO Mapping Parameter

This object defines data transmitted by PDO. Subindex 0 contains number of objects valid for PDO.

Index	Subindex	Name	Type	Default	R/W
0x1600	**0**	Number of entries	Record 8bit unsigned	**-**	R/W
	1...2	PDO mapping object$_i$ (i=1,2)	Record 32bit unsigned	**-**	R/W

Object$_i$ structure:

Bit 16...31	Bit 8...15	Bit 0...7
Index	Subindex	Object size

Index: object index to be transmitted
Subindex: object subindex to be transmitted
Object size: object size in bits (no more than 2 bytes can be transmitted in a PDO, so sum of valid object lengths have not to exceed 16

7.2.21 Transmit PDO Communication Parameter

This object set communication parameters of Tx PDO supported. Default COB-ID of PDO is set by DS301 standard.

Index	Subindex	Name	Type	Default	R/W

0x1800	**0**	Number of entries	Record 8bit unsigned	**5**	R
	1	COB-ID used by PDO1	Record 32bit unsigned	**0x1800 0x180 + Module-ID**	R/W
	2	Transmission type	Record 8bit unsigned	**255**	R/W
	3	Inhibit Time	Record 16bit unsigned	**0**	R/W
	5	Event Timer	Record 16bit unsigned	**0**	R/W

COB-ID Structure:

Bit 31	Bit 30	Bit 29...11	Bit 0...10
0(valid) / **1**(invalid)	**0**(RTR allowed) / **1**(RTR not allowed)	**0** Reserved	**COB-ID**

Digital and Analogue inputs are transmitted as Change Of Value (COV). Type of transmission depending upon set transmission type and it's explained in the following table:

Trans-mission type	PDO transmission						
	cyclic	acyclic	synchro -nous	asynchro -nous	RTR only	TxPDO (inputs)	RxPDO (outputs)
0		X	X			If COV is transmitted with each SYNC	Set outputs after each SYNC as requested by last PDO received
1...240	X		X			Transmission with each i SYNC (i = 1...240)	Set outputs after each SYNC as requested by last PDO received
241..251	Reserved						

252			X		X	Data is read again with a SYNC, but not sent, request via RTR	Not supported
253				X	X	Request via RTR	COV
254				X		COV	COV
255				X		COV	COV

Inhibit Time is minimum time between two consecutive PDOs with same COB-ID (unit time 100μsec).
Event Timer defines time after the elapsed of a sent PDO, even if no change of data has occured (millisecond). It can be used only for transmission types 254 and 255.

7.2.22 Transmit PDO Mapping

This object defines data transmitted by PDO. Subindex 0 contains number of objects valid for PDO.

Index	Subindex	Name	Type	Default	R/W
0x1A00	**0**	Number of entries	Record 8bit unsigned	**-**	R/W
	1...8	PDO mapping $object_i$ (i=1...8)	Record 32bit unsigned	**-**	R/W

$Object_i$ structure:

Bit 16...31	Bit 8...15	Bit 0...7
Index	Subindex	Object size

Index: object index to be transmitted
Subindex: object subindex to be transmitted
Object size: object size in bits (no more than 8 bytes can be transmitted in

a PDO, so sum of valid object lengths have not to exceed 64

7.3 Manufacturer Specific Parameter Area MCM260-1/2/3/4AD

The following table shows all Pixsys specific parameters objects supported:

N.B. **ALL** = All devices
I_IO = Input devices (**MCM260-2AD**), Input/Output devices (**MCM260-3AD**)

Index	Name	Type	Device Type	R/W
0x2000	Device specification	Array 16bit signed	**ALL**	**R/W**
0x2100	Encoder counter	Array 32bit signed	**I_IO**	**R/W**
0x2101	Preset encoder	Record 32bit signed	**I_IO**	**R/W**
0x2102	Encoder Command Preset	Array 8bit unsigned	**I_IO**	**R/W**
0x2103	Encoder type	Array 8bit unsigned	**I_IO**	**R/W**
0x2104	Encoder speed (0.1msec)	Record 16bit signed	**I_IO**	**R**
0x2105	Encoder speed (1sec)	Record 16bit signed	**I_IO**	**R**
0x2200	Proximity counter	Record 32bit unsigned	**I_IO**	**R/W**
0x2201	Preset proximity	Array 32bit unsigned	**I_IO**	**R/W**
0x2202	Proximity Command Preset	Array 8bit unsigned	**I_IO**	**R/W**
0x2203	Timer ON proximity	Record 16bit unsigned	**I_IO**	**R**
0x2204	Proximity speed (0.1msec)	Record 16bit unsigned	**I_IO**	**R**
0x2205	Proximity speed (1sec)	Record 16bit unsigned	**I_IO**	**R**

7.3.1 Device Specification

This object defines most important device configuration parameters: baud rate, node-ID offset multiplier (read only), boot-up time, device status after boot-up, filters of digital inputs and Pixsys pre-load parameters.

Index	Subindex	Name	Type	Default	R/W
0x2000	**0**	Number of entries	Array 16bit signed	**10**	R
	1	Baud rate MCM260	Array 16bit signed	**6 (1Mbps)**	R
	2	Node-ID offset multiplier	Array 16bit signed	**0**	R/W
	3	Boot-up time	Array 16bit signed	**50 (1=10msec)**	R/W
	4	CANopen state after boot-up	Array 16bit signed	**0x7F (Pre-Operational)**	R/W
	5	Filter digital inputs	Array 16bit signed	**1**	R/W
	6	Pre-load Pixsys parameters	Array 16bit signed	**0**	R/W
	7...10	Reserved			R/W

Baud rate specifications: this is a read-only object. It can be only modified by rotary switch in start-up mode (when device is powered-on with rotary switch in position 0)

Index	Subindex	Name	Type	Values	Baud rate
0x2000	**1**	Baud rate MCM260	Array 16bit signed	**0**	**50 Kbit/sec**
				1	**62.5 Kbit/sec**

				2	**100 Kbit/sec**
				3	**125 Kbit/sec**
				4	**250 Kbit/sec**
				5	**500 Kbit/sec**
				6	**1 Mbit/sec**

Node-ID offset multiplier specifications : this object defines node-ID offset multiplier of device. If Index 0x2000 Subindex1 = 3, node-ID offset of device will be 3x15 = 45. This value has to be added to number specified by rotary switch position (when device has be powered-on in a position not equal to 0). If rotary switch position in Operational mode is 8, total node-ID CANopen address will be 45+8 = 53.
It's possible to connect up to 120 modules in a MCM260 CANopen network (offset multiplier up to 105, rotary switch up to 0xF = 15, 105+15 = 120).
Node-ID = (**N**-1) x 15 + **M**,
where **N** = 0x2000 Subindex 2, **M** = rotary switch position (not 0)

Index	Subindex	Name	Type	Values	Offset
0x2000	**2**	Node-ID offset multiplier	Array 16bit signed	**0**	Offset 0 Index 1...15
				1	Offset 1 Index 16...30
				2	Offset 2 Index 31...45
				3	Offset 3 Index 46...60
				4	Offset 4 Index 61...75
				5	Offset 5 Index 76...90
				6	Offset 6 Index 91...105
				7	Offset 7 Index 106...120

Boot-up time specifications: this object defines how long is boot-up time (unit is 10msec)

Index	Subindex	Name	Type	Values	Time
0x2000	**3**	Boot-up Time	Array 16bit signed	**10...100**	100msec...1sec

CANopen state after boot-up specifications: CANopen standard defines that once boot-up is finished, device changes automatically into Pre-Operational status. This is default configuration (0x7F), but it's possible to change into other status:

Index	Subindex	Name	Type	Values	Status
0x2000	**4**	CANopen state after boot-up	Array 16bit signed	**0**	**Boot-up**
				4	**Stopped**
				5	**Operational**
				0x7F	**Pre-Operational**

Filter digital input specifications: this object defines average number of circular filter for digital input. Filter is timed, its unit is 500µsec.

Index	Subindex	Name	Type	Values	Comment
0x2000	**5**	Filter digital inputs	Array 16bit signed	**1...20**	1...20 averages

Pre-load Pixsys parameters specifications: this object defines if device has to pre-load default CANopen parameters (0x2000, Subindex6 = 0) or default Pixsys parameters (0x2000, Subindex6 = 0x100)

Index	Subindex	Name	Type	Values	Coment

0x2000	**6**	Pre-load Pixsys parameters	Array 16bit signed	**0**	Pre-load CANopen standard parameters
				0x100	Pre-load Pixsys parameters

7.3.2 Encoder counter

This object contains countings number of mono/bidirectional encoder eventually connected to MCM260 (phase A connected to DI8, phase B connected to DI7)

Index	Subindex	Name	Type	Default	R/W
0x2100	**0**	Number of entries	Array 8bit signed	**1**	R
	1	Encoder counter	Array 32bit signed	**0**	R

7.3.3 Preset encoder

This object contains preset value of encoder eventually connected to MCM260. This is the value which has to be loaded to encoder counter object (0x2100, Subindex1) in the event of a preset command (see paragraph 7.3.4)

Index	Subindex	Name	Type	Default	R/W
0x2101	**0**	Number of entries	Record 8bit signed	**1**	R
	1	Preset encoder	Record 32bit signed	**0**	R/W

7.3.4 Encoder command preset

This object contains command to load preset value (0x2101, Subindex1) into encoder counter object (0x2100, Subindex1). There are two way to load preset encoder value:

- Preset value is loaded on encoder counter object when value 1 is

written on 0x2102, Subindex1 (at the same time)

- Preset value is loaded on encoder counter object when zero impulse (DI1) is active if value 1 is written on 0x2102, Subindex1

In both cases, 0x2102 is reset to 0 after preset value has been loaded into encoder counter object.

Index	Subindex	Name	Type	Default	R/W
0x2102	**0**	Number of entries	Array 8bit unsigned	**1**	R
	1	Encoder command preset	Array 8bit unsigned	**0**	R/W

Structure:

Index	Subindex	Name	Type	Values	Comment
0x2102	**1**	Encoder command preset	Array 8bit unsigned	**0**	No preset is loaded
				1	Preset value loaded simultaneously
				2	Preset value loaded when DI1=1 (zero impulse)

7.3.5 Encoder type

This object defines which type of encoder is connected: 1 means bi-directional encoder, 2 means mono-directional encoder.

Index	Subindex	Name	Type	Default	R/W
0x2103	**0**	Number of entries	Array 8bit signed	**1**	R
	1	Encoder type	Array 8bit signed	**0**	R/W

7.3.6 Encoder speed

These two objects, 0x2104 and 0x2105 give information about encoder speed. The first one, 0x2104, contains speed encoder in 100μsec untis, the second one, 0x2105, in second units.

Index	Subindex	Name	Type	Default	R/W
0x2104	**0**	Number of entries	Record 8bit signed	**1**	R
	1	Encoder speed (100µsec)	Record 16bit signed	**0**	R

Index	Subindex	Name	Type	Default	R/W
0x2105	**0**	Number of entries	Record 8bit signed	**1**	R
	1	Encoder speed (1sec)	Record 16bit signed	**0**	R

7.3.7 Proximity counter

This object contains countings number of proximities eventually connected to MCM260 (proximity 1 to DI5, proximity 2 to DI6).

Index	Subindex	Name	Type	Default	R/W
0x2200	**0**	Number of entries	Record 8bit signed	**2**	R
	1	Proximity 1 counter	Record 32bit unsigned	**0**	R/W
	2	Proximity 2 counter	Record 32bit unsigned	**0**	R/W

7.3.8 Preset proximity

This object contains preset value of proximities eventually connected to MCM260. This is the value which has to be loaded to proximity counter object (0x2200, Sub-index 1 and 2) in the event of a preset command (see paragraph 7.3.7)

Index	Subindex	Name	Type	Default	R/W
0x2201	**0**	Number of entries	Record 8bit signed	**2**	R
	1	Preset Proximity 1	Record 32bit unsigned	**0**	R/W
	2	Preset Proximity 2	Record 32bit unsigned	**0**	R/W

7.3.9 Proximity command preset

This object contains command to load preset value (0x2201, Subindex1 and 2) into proximity counter objects (0x2200, Sub-index 1 and 2).
Preset value is loaded on proximity counter objects when value 1 is written on 0x2202, Sub-index 1 or 2 (at the same time). This object is always reset to 0 after preset value has been loaded into proximity counter object.

Index	Subindex	Name	Type	Default	R/W
0x2202	**0**	Number of entries	Array 8bit signed	**2**	R
	1	Proximity 1 command preset	Array 8bit unsigned	**0**	R/W
	2	Proximity 2 command preset	Array 8bit unsigned	**0**	R/W

7.3.10 Timer ON proximity

This object shows last active time of proximities (time ON, unit is 500µsec). It's a read-only object.

Index	Subindex	Name	Type	Default	R/W
0x2203	**0**	Number of entries	Record 8bit signed	**2**	R
	1	Timer ON Proximity 1	Record 16bit unsigned	**0**	R

	2	Timer ON Proximity 2	Record 16bit unsigned	**0**	R

7.3.11 Proximity speed

These two objects, 0x2204 and 0x2205, give information about proximity speed. The first one, 0x2104, contains speed proximity in 100μsec units, the second one, 0x2105, in second units.

Index	Subindex	Name	Type	Default	R/W
0x2104	**0**	Number of entries	Record 8bit signed	**1**	R
	1	Proximity 1 speed (100μsec)	Record 16bit unsigned	**0**	R
	2	Proximity 2 speed (100μsec)	Record 16bit unsigned	**0**	R

Index	Subindex	Name	Type	Default	R/W
0x2105	**0**	Number of entries	Record 8bit signed	**1**	R
	1	Proximity 1 speed (1sec)	Record 16bit unsigned	**0**	R
	2	Proximity 2 speed (1sec)	Record 16bit unsigned	**0**	R

7.4 Manufacturer Specific Parameter Area – MCM260-5AD

The following table shows all Pixsys specific parameters objects supported:

Index	Nome	Tipo	R/W

0x2000	Device specification	Array 16bit signed	**R/W**
0x4000	AI1 parameters	Array 16bit signed	**R/W**
0x4001	AI2 parameters	Array 16bit signed	**R/W**
0x4002	AI3 parameters	Array 16bit signed	**R/W**
0x4003	AI4 parameters	Array 16bit signed	**R/W**
0x4100	Analogue inputs parameters	Array 16bit signed	**R/W**
0x5000	AO1 parameters	Array 16bit signed	**R/W**
0x5001	AO2 parameters	Array 16bit signed	**R/W**

7.4.1 Device Specification

This object defines most important device configuration parameters: baud rate, node-ID offset multiplier (read only), boot-up time, device status after boot-up, filters of digital inputs and Pixsys pre-load parameters.

Index	Subindex	Name	Type	Default	R/W
0x2000	**0**	Number of entries	Array 16bit signed	**10**	R
	1	Baud rate MCM260	Array 16bit signed	**6 (1Mbps)**	R
	2	Node-ID offset multiplier	Array 16bit signed	**0**	R/W
	3	Boot-up time	Array 16bit signed	**50 (1=10msec)**	R/W
	4	CANopen state after boot-up	Array 16bit signed	**0x7F (Pre-Operational)**	R/W
	5	Pre-load Pixsys parameters	Array 16bit signed	**0**	R/W
	6...10	Reserved			R/W

Baud rate specifications: this is a read-only object. It can be only modified by rotary switch in start-up mode (when device is powered-on with rotary switch in position 0)

Index	Subindex	Name	Type	Values	Baud rate
0x2000	**1**	Baud rate MCM260	Array 16bit signed	**0**	**50 Kbit/sec**
				1	**62.5 Kbit/sec**
				2	**100 Kbit/sec**
				3	**125 Kbit/sec**

				4	**250 Kbit/sec**
				5	**500 Kbit/sec**
				6	**1 Mbit/sec**

Node-ID offset multiplier specifications : this object defines node-ID offset multiplier of device. If Index 0x2000 Subindex1 = 3, node-ID offset of device will be 3x15 = 45. This value has to be added to number specified by rotary switch position (when device has be powered-on in a position not equal to 0). If rotary switch position in Operational mode is 8, total node-ID CANopen address will be 45+8 = 53.
It's possible to connect up to 120 modules in a MCM260 CANopen network (offset multiplier up to 105, rotary switch up to 0xF = 15, 105+15 = 120).
Node-ID = (**N**-1) x 15 + **M**,
where **N** = 0x2000 Subindex 2, **M** = rotary switch position (not 0)

Index	Subindex	Name	Type	Values	Offset
0x2000	**2**	Node-ID offset multiplier	Array 16bit signed	**0**	Offset 0 Index 1...15
				1	Offset 1 Index 16...30
				2	Offset 2 Index 31...45
				3	Offset 3 Index 46...60
				4	Offset 4 Index 61...75
				5	Offset 5 Index 76...90
				6	Offset 6 Index 91...105
				7	Offset 7 Index 106...120

Boot-up time specifications: this object defines how long is boot-up time (unit is 10msec)

Index	Subindex	Name	Type	Values	Time
0x2000	**3**	Boot-up	Array	**10...100**	100msec...1sec

		Time	16bit signed		

CANopen state after boot-up specifications: CANopen standard defines that once boot-up is finished, device changes automatically into Pre-Operational status. This is default configuration (0x7F), but it's possible to change into other status:

Index	Subindex	Name	Type	Values	Status
0x2000	**4**	CANopen state after boot-up	Array 16bit signed	**0**	**Boot-up**
				4	**Stopped**
				5	**Operational**
				0x7F	**Pre-Operational**

Pre-load Pixsys parameters specifications: this object defines if device has to pre-load default CANopen parameters (0x2000, Subindex6 = 0) or default Pixsys parameters (0x2000, Subindex6 = 0x100)

Index	Subindex	Name	Type	Values	Comment
0x2000	**5**	Pre-load Pixsys parameters	Array 16bit signed	**0**	Pre-load CANopen standard parameters
				0x100	Pre-load Pixsys parameters

7.4.2 AI1, AI2, AI3, AI4 parameters

This object contains the parameters for each analog input (Index 0x4000 for AI1, Index 0x4001 for AI2, Index 4002 for AI3 ,Index 4003 for AI4I).

Index	Subindex	Name	Type	Default	R/W
0x4000 0x4001	**0**	Number of Sub-index	Array 16 bit signed	**20**	R/W
0x4002	**1**	Selection of sensor	Array 16 bit signed	**0**	R/W

0x4003	**2**	Lower limit for linear signals	Array 16 bit signed	**0**	R/W
	3	Upper limit for linear signals	Array 16 bit signed	**0x7FFF**	R/W
	4	Offset calibration	Array 16 bit signed	**0**	R/W
	5	Gain calibration	Array 16 bit signed	**0**	R/W
	6	Software filter	Array 16 bit signed	**5**	R/W
	7	Select type of degrees	Array 16 bit signed	**0**	R/W
	8…20	Reserved	Array 16 bit signed	**0**	R/W

Specifiche selezione sensore: Questo oggetto definisce il tipo di sensore da collegare all'ingresso.

Index	Subindex	Name	Type	Value	Baud rate
0x4000 0x4001 0x4002 0x4003	**1**	Selection of sensor	Array 16bit signed	**0**	**Desabled**
				1	**thermocouple K (-260…1360℃)**
				2	**thermocouple S (-40…1760℃)**
				3	**thermocouple R (-40…1760℃)**
				4	**thermocouple J (-200…1200℃)**
				5	**PT100 (-200…600℃)**
				6	**PT100 (-200…140℃)**
				7	**NI100 (-60…180℃)**
				8	**NTC 10K (β3435K) (-40…125℃)**
				9	**PTC 1K (-50…150℃)**

				10	**PTC500 (-100…600℃)**
				11	**PTC1000 (-100…600℃)**
				12	**Input 0...10V**
				13	**Input 0...20mA**
				14	**Input 4…20mA**
				15	**Input 0…40mV**
				16	**Potentiometer max. 6KOhm**
				17	**Potentiometer max. 150KOhm**

Specifications for lower limit of linear signals: this object defines the minimum value of input if it is selected for linear signals.

Index	Subindex	Name	Type	Range
0x4000 **0x4001** **0x4002** **0x4003**	**2**	Lower limit for linear signals	Array 16bit signed	**-32768...32767 unit**

Specifications for upper limit of linear signals: this object defines the maximum value of input if it is selected for linear signals.

Index	Subindex	Name	Type	Range
0x4000 **0x4001** **0x4002** **0x4003**	**3**	Upper limit for linear signals	Array 16bit signed	**-32768...32767 unit**

Offset calibration of input : this number is added or deducted to/from value of input reading (process value)

Index	Subindex	Name	Type	Range
0x4000 **0x4001**	**4**	Offset calibration	Array 16bit signed	**-1000...1000 unit**

0x4002 **0x4003**				

Gain calibration of input: this number is multiplied for value of input reading (process value)

Index	Subindex	Name	Type	Range
0x4000 **0x4001** **0x4002** **0x4003**	**5**	Offset calibration	Array 16bit signed	**-1000...1000 unit**

Software filter:
Number of means to calculate the value of input (to stabilize the value)

Index	Subindex	Name	Type	Range
0x4000 **0x4001** **0x4002** **0x4003**	**6**	Software filter	Array 16bit signed	**1...10 means**

Type of degrees: type of degrees for temperature sensors

Index	Subindex	Name	Type	Values	Comments
0x4000 **0x4001** **0x4002** **0x4003**	**7**	Select type of degrees	Array 16bit signed	**0**	Celsius
				1	Fahrenheit

7.4.3 Analogue Inputs Parameters

This object contains the parameters which are common to analog inputs

Index	Subindex	Name	Type	Default	R/W
0x4100	**0**	Numer of Sub-index	Array 16bit signed	**2**	R
	1	Sampling frequency AI1 and AI2	Array 16bit signed	**7**	R/W

	2	Sampling frequency AI3 and AI4	Array 16bit signed	**7**	R/W

Sampling frequency for analog inputs: (subindex 1 for AI1 and AI2; subindex 2 for AI3 and AI4)

Index	Subindex	Name	Type	Value	Baud rate
0x4100	**1** **2**	Sampling frequency	Array 16bit signed	**0**	**242 HZ**
				1	**123 HZ**
				2	**62 HZ**
				3	**50 HZ**
				4	**39 HZ**
				5	**33.2 HZ**
				6	**19.6 HZ**
				7	**16.7 HZ**
				8	**12.5 HZ**
				9	**10 HZ**
				10	**8.33 HZ**
				11	**6.25 HZ**
				12	**4.17 HZ**

7.4.4 AO1 AO2 Parameters

Configuration parameters for each analog output (Index 0x5000 for AO1, Index 0x5001 for AO2).

Index	Subindex	Name	Type	Default	R/W
0x5000 **0x5001**	**0**	Number of Sub-index	Array 16 bit signed	**10**	R/W
	1	Select type of output	Array 16 bit signed	**0**	R/W

	2	Lower limit	Array 16 bit signed	**0**	R/W
	3	Upper limit	Array 16 bit signed	**0x7FFF**	R/W
	4…10	Reserved	Array 16 bit signed	**0**	R/W

Specifications for type of analog output:

Index	Subindex	Name	Type	Values	Comments
0x5000 0x5001	**1**	Type of output	Array 16bit signed	**0**	0…10V
				1	4…20mA

Lower limit analog output

Index	Subindex	Name	Type	Range
0x5000 0x5001	**2**	Lower limit	Array 16bit signed	**-32768...32767 unit**

Upper limit analog output

Index	Subindex	Name	Type	Range
0x5000 0x5001	**3**	Uper limit	Array 16bit signed	**-32768...32767 unit**

7.5 Standard Device Profile Area

The following table shows all the supported standard device profile objects:

N.B. **ALL** = All devices
O_IO = Output devices (**MCM260-1AD**), Input/Output devices (**MCM260-3AD, MCM260-4AD**)
I_IO = Input devices (**MCM260-2AD**), Input/Output devices (**MCM260-3AD, MCM260-4AD**)
IN_AN = Input devices (**MCM260-2AD**), Analogue Input/Output devices (**MCM260-5AD**)

AN = Analogue Input/Output devices (**MCM260-5AD**)

Index	Name	Type	Device Type	R/W
0x6000	Digital Input	Array 8bit unsigned	**I_IO**	**R**
0x6005	Global Interrupt Enable Digital 8bit	8bit unsigned	**I_IO**	**R/W**
0x6006	Interrupt Mask Any Change 8bit	Array 8bit unsigned	**I_IO**	**R/W**
0x6007	Interrupt Mask Low-to-High 8bit	Array 8bit unsigned	**I_IO**	**R/w**
0x6008	Interrupt Mask High-to-Low 8bit	Array 8bit unsigned	**I_IO**	**R/W**
0x6200	Digital Output	Array 8bit unsigned	**O_IO**	**R/W**
0x6206	Digital Output Error Mode	Array 8bit unsigned	**O_IO**	**R/W**
0x6207	Digital Output Error Value	Array 8bit unsigned	**O_IO**	**R/W**
0x6401	Read Analogue input 16bit	Array 16bit unsigned	**IN_AN**	**R**
0x6411	Analogue output 16bit	Array 16bit unsigned	**AN**	**R/W**
0x6421	Analogue input Trigger Selection	Array 8bit unsigned	**IN_AN**	**R/W**
0x6423	Analogue input Global Interrupt Selection	Boolean	**IN_AN**	**R/W**
0x6424	Analogue input Interrupt Upper Limit Integer	Array 16bit unsigned	**IN_AN**	**R/W**
0x6425	Analogue input Interrupt Lower Limit Integer	Array 16bit unsigned	**IN_AN**	**R/W**
0x6426	Analogue input Interrupt Delta Unsigned	Array 16bit unsigned	**IN_AN**	**R/W**
0x6427	Analogue input Negative Delta	Array 16bit unsigned	**IN_AN**	**R/W**

	Unsigned			

0x6428	Analogue input Positive Delta Unsigned	Array 16bit unsigned	**IN_AN**	**R/W**
0x6443	Analogue output Error Mode	Array 16bit unsigned	**AN**	**R/W**
0x6444	Analogue output Error Value	Array 16bit unsigned	**AN**	**R/W**
0x67FE	Error Behaviour	Array 8bit unsigned	**ALL**	**R/W**

7.5.1 Digital Input

This object contains data of digital input modules. Subindex 1 the first 8 digital input channels, subindex 2 the second block.

Index	Subindex	Name	Type	Default	R/W
0x6000	**0**	Number of entries	Array 8bit unsigned	-	R
	1	1st input block	Array 8bit unsigned	-	R
	2	2nd input block	Array 8bit unsigned	-	R

7.5.2 Global Interrupt Enable Digital 8bit

This object allows digital input transmission by PDO. If value is 1, transmission is generally released and it's regulated by objects 0x6006, 0x6007, 0x6008 and type of PDO transmission. If value is 0, digital input isn't transmitted.

Index	Subindex	Name	Type	Default	R/W
0x6005	**0**	Global Interrupt Enable Digital 8bit	8bit unsigned	**1**	R/W

7.5.3 Interrupt Mask Any Change 8bit

This object is used to define digital input channel, which will send its data in a event of a change (Global Interrupt has to be enabled, 0x6005=1).

Index	Subindex	Name	Type	Default	R/W
0x6006	**0**	Number of entries	Array 8bit unsigned	**2**	R
	1	1st input block	Array 8bit unsigned	**255**	R/W
	2	2nd input block	Array 8bit unsigned	**255**	R/W

$\mathbf{b_i}$	**0**	Transmission channel$_i$ blocked in a event of a change
	1	Transmission channel$_i$ released in a event of a change

Example: if Subindex 0 = 1, Subindex 1 = 57 = 0x41 = 00111001_2 means that channel 1, 4, 5 and 6 will transmit their data in the event of a change.

7.5.4 Interrupt Mask Low-to-High 8bit

This object defines which digital input channel will send its data in a event of a positive transition (Global Interrupt has to be enabled, 0x6005=1).

Index	Subindex	Name	Type	Default	R/W
0x6007	**0**	Number of entries	Array 8bit unsigned	**-**	R
	1	1st input block	Array 8bit unsigned	**0**	R/W
	2	2nd input block	Array 8bit unsigned	**0**	R/W

$\mathbf{b_i}$	**0**	Transmission channel$_i$ blocked with a positive transition
	1	Transmission channel$_i$ released with a positive transition

Example: if 0x6006, Subindex 0 = 1, Subindex 1 = 57 = 0x41 = 00111001_2

0x6007, Subindex 0 = 1, Subindex 1 = 11 = 0xB = 00001011_2 means that channel 1, 4, 5 and 6 will transmit their data in the event of a change, while channel 2 will only transmit with a positive transition.

7.5.5 Interrupt Mask High-to-Low 8bit

This object defines which digital input channel will send its data in a event of a negative transition (Global Interrupt has to be enabled, 0x6005=1).

Index	Subindex	Name	Type	Default	R/W
0x6008	**0**	Number of entries	Array 8bit unsigned	**-**	R
	1	1st input block	Array 8bit unsigned	**0**	R/W
	2	2nd input block	Array 8bit unsigned	**0**	R/W

b_i	**0**	Transmission channel$_i$ blocked with a negative transition
	1	Transmission channel$_i$ released with a negative transition

Example: if 0x6006, Subindex 0 = 1, Subindex 1 = 57 = 0x41 = 00111001_2
0x6007, Subindex 0 = 1, Subindex 1 = 11 = 0xB = 00001011_2 means that channel 1, 4, 5 and 6 will transmit their data in the event of a change, while channel 2 will only transmit with a negative transition.

7.5.6 Digital Output

This object contains data of digital output modules. Subindex 1 contains the first 8 digital output channels, subindex 2 the second block.

Index	Subindex	Name	Type	Default	R/W
0x6200	**0**	Number of entries	Array 8bit unsigned	**-**	R
	1	1st output block	Array 8bit unsigned	**0**	R/W
	2	2nd output block	Array 8bit unsigned	**0**	R/W

7.5.7 Error Mode Output 8bit

This object defines if output change to a pre-defined error status in a event of an error or not. If error is eliminated, outputs are maintained in their pre-defined error status.

Index	Subindex	Name	Type	Default	R/W
0x6206	**0**	Number of entries	Array 8bit unsigned	**-**	R
	1	1st output block	Array 8bit unsigned	**255**	R/W
	2	2nd output block	Array 8bit unsigned	**255**	R/W

b_i	**0**	Output $channel_i$ doesn't change in a event of an error
	1	Output $channel_i$ change to a pre-defined error

7.5.8 Error Value Output 8bit

This objects defines values outputs have to change to in a event of an error (corresponding bit in Error Mode Output has to be enabled, 0x6206).

Index	Subindex	Name	Type	Default	R/W
0x6207	**0**	Number of entries	Array 8bit unsigned	**-**	R
	1	1st output block	Array 8bit unsigned	**0**	R/W
	2	2nd output block	Array 8bit unsigned	**0**	R/W

b_i	**0**	Output $channel_i$ change to 0 in case of an error
	1	Output $channel_i$ change to 1 in case of an error

Example: if 0x6206, Subindex 0 = 1, Subindex 1 = 57 = 0x41 = 00111001_2 0x62607, Subindex 0 = 1, Subindex 1 = 11 = 0xB = 00001011_2 means

that channels 1 and 4 are set to 1, channels 5 and 6 are set to 0, while all other output doesn't change in an event of an error.

7.5.9 Analogue Input 16bit

This object contains process data of Analogue input channels (MCM260-2AD, MCM260-5AD).

Index	Subindex	Name	Type	Default	R/W
0x6401	**0**	Number of Analogue input channels	Array 8bit unsigned	**2 (-2AD)** **4 (-5AD)**	R
	1	1st channel	Array 16bit unsigned	**-**	R
	2	2nd channel	Array 16bit unsigned	**-**	R
	3	3rd channel	Array 16bit unsigned	**-**	R
	4	4th channel	Array 16bit unsigned	**-**	R

7.5.10 Analogue Output 16bit

This object contains the value of Analogue 16 bit output channels (MCM260-5AD).

Index	Subindex	Name	Type	Default	R/W
0x6401	**0**	Number of Analogue output channels	Array 8bit unsigned	**2**	R
	1	1st channel	Array 16bit unsigned	**0**	R/W
	2	2nd channel	Array 16bit unsigned	**0**	R/W

7.5.11 Analogue Input Interrupt Trigger Selection

This object defines condition of transmission: when 1 is entered in object

0x6423, then transmission is released.

Index	Subindex	Name	Type	Default	R/W
0x6421	**0**	Number of Analogue input channels	Array 8bit unsigned	2 (-2AD) 4 (-5AD)	**R**
	1	Trigger 1st channel	Array 8bit unsigned	7	**R/W**
	2	Trigger 2nd channel	Array 8bit unsigned	7	**R/W**
	3	Trigger 3rd channel	Array 8bit unsigned	7	**R/W**
	4	Trigger 4th channel	Array 8bit unsigned	7	**R/W**

Sub-index structure:

Bit	Transmission conditions	Subindex configuration
0	Threshold value exceeded	**0x6424**
1	Threshold value fallen short	**0x6425**
2	Change of input value exceeding delta value for last transmission	**0x6426**
3	Reduction of input value by more than delta value for last transmission	**0x6427**
4	Increase of input value by more than delta value for last transmission	**0x6428**
5...7	Reserved	**-**

7.5.12 Analogue Input Global Interrupt Enable

This object is used to control Analogue input transmission by PDO. If its value is 1, transmission is released and it only depends on object 0x6421 and PDO transmission type. If value is 0, Analogue input transmission is not allowed.

Index	Subindex	Name	Type	Default	R/W
0x6423	**0**	Global Interrupt Enable Analogue input 16bit	Boolean	**0**	R/W

7.5.13 Analogue Input Interrupt Upper Limit Integer

This object allows a threshold value monitoring for Analogue input

transmission. If it's configured in object 0x6423, transmission will take place if input value is ≥ threshold value when a trigger condition is set.

Index	Subindex	Name	Type	Default	R/W
0x6424	**0**	Number of Analogue input channels	Array 8bit unsigned	**2 (-2AD) 4 (-5AD)**	R
	1	Upper limit 1st channel	Array 16bit unsigned	**0**	R/W
	2	Upper limit 2nd channel	Array 16bit unsigned	**0**	R/W
	3	Upper limit 3rd channel	Array 16bit unsigned	**0**	R/W
	4	Upper limit 4th channel	Array 16bit unsigned	**0**	R/W

7.5.14 Analogue Input Interrupt Lower Limit Integer

This object allows a threshold value monitoring for Analogue input transmission. If it's configured in object 0x6423, transmission will take place if input value is ≤ threshold value when a trigger condition is set.

Index	Subindex	Name	Type	Default	R/W
0x6425	**0**	Number of Analogue input channels	Array 8bit unsigned	**2 (-2AD) 4 (-5AD)**	R
	1	Lower limit 1st channel	Array 16bit unsigned	**0**	R/W
	2	Lower limit 2nd channel	Array 16bit unsigned	**0**	R/W
	3	Lower limit 3rd channel	Array 16bit unsigned	**0**	R/W
	4	Lower limit 4th channel	Array 16bit unsigned	**0**	R/W

7.5.15 Analogue Input Interrupt Delta Unsigned

If this object is allowed, it conditions actual Analogue input transmission with previously sent value. New value is transmitted only if it's larger than previously sent value + Delta, or if it's smaller than previously sent value – Delta.

Index	Subindex	Name	Type	Default	R/W
0x6426	**0**	Number of Analogue input channels	Array 8bit unsigned	**2 (-2AD)** **4 (-5AD)**	R
	1	Delta 1st channel	Array 16bit unsigned	**0**	R/W
	2	Delta 2nd channel	Array 16bit unsigned	**0**	R/W
	3	Delta 3rd channel	Array 16bit unsigned	**0**	R/W
	4	Delta 4th channel	Array 16bit unsigned	**0**	R/W

7.5.16 Analogue Input Interrupt Negative Delta Unsigned

If this object is allowed, it conditions actual Analogue input transmission with previously sent value. New value is transmitted only it's smaller than previously sent value – Delta.

Index	Subindex	Name	Type	Default	R/W
0x6427	**0**	Number of Analogue input channels	Array 8bit unsigned	**2 (-2AD)** **4 (-5AD)**	R
	1	Delta 1st channel	Array 16bit unsigned	**0**	R/W
	2	Delta 2nd channel	Array 16bit unsigned	**0**	R/W
	3	Delta 3rd channel	Array 16bit unsigned	**0**	R/W
	4	Delta 4th channel	Array 16bit unsigned	**0**	R/W

7.5.17 Analogue Input Interrupt Positive Delta Unsigned

If this object is allowed, it conditions actual Analogue input transmission with previously sent value. New value is transmitted only if it's larger than previously sent value + Delta.

Index	Subindex	Name	Type	Default	R/W
0x6428	**0**	Number of Analogue input channels	Array 8bit unsigned	**2 (-2AD)** **4 (-5AD)**	R
	1	Delta 1st channel	Array 16bit unsigned	**0**	R/W
	2	Delta 2nd channel	Array 16bit unsigned	**0**	R/W
	3	Delta 3rd channel	Array 16bit unsigned	**0**	R/W
	4	Delta 4th channel	Array 16bit unsigned	**0**	R/W

7.5.18 Analogue Output Error Mode

This object defines if the output must assume a pre-selected state in case of error (see object 0x6444). If error is solved, the outputs will keep the pre-selected state.

Index	Subindex	Name	Type	Default	R/W
0x6443	**0**	Number of analogue output	Array 8bit unsigned	**2**	R
	1	Error Mode 1st output	Array 8bit unsigned	**1**	R/W
	2	Error Mode 2nd output	Array 8bit unsigned	**1**	R/W

b_i	**0**	Output state remains unchanged
	1	Output state changes in case of error

7.5.19 Analogue Output Error Value Integer

Value assumed by analog output in case of error. For this purpose the

object 0x6443 must be set to 1.

Index	Subindex	Name	Type	Default	R/W
0x6444	**0**	Number of analogue output	Array 8bit unsigned	**2**	R
	1	Error Value 1st output	Array 16bit signed	**0**	R/W
	2	Error Value 2nd output	Array 16bit signed	**0**	R/W

7.5.20 Error Behaviour

This object has the same functionality of Error Behaviour 0x1029.

Index	Subindex	Name	Type	Default	R/W
0x67FE	**0**	Number of entries (1)	Array 8bit unsigned	**1**	R
	1	Communication error	Array 8bit unsigned	**0**	R/W

Structure:

Communication error	Action
0	Change into status PRE-OPERATIONAL (only if actual status was OPERATIONAL)
1	No status change
2	Change into status STOPPED

7.6 PDO Transmission

Data transmission with PDO is only allowed in Operational status. When device changes its status into Operational, TX PDO is transmitted once with transmission type 254 and 255.
In order to prevent CAN bus overflow, default value for object 0x6423 is false, so Analogue changes aren't transmitted. To prevent overflow with 0x6423=true, a long Inhibit Time can be selected, or properly values for Threshold and Delta (0x6421...0x6428) can be set.

7.6.1 PDO Mapping

If stored customer specific configuration isn't used, object dictionary is assigned with default configuration according to standard device profile DS401 (see paragraph 6.1.5).
If device is in Pre-Operational status, its mapping can be modified via SDO.

7.7 SYNC Monitoring

In Operational status, if communication cycle period isn't equal to 0, monitoring is released with the first SYNC message.
If SYNC message isn't received within monitoring time (communication cycle period), a blink code is provided and status doesn't change. An emergency message (Error Code:0x8100, Error Register: 0x81, Additional Code 00 04 00 00 00) is sent. Failure of SYNC message will be displayed even if master provides a status change.
LEDs return to their normal operating status only after new SYNC message receipt in Operational status, and another emergency message is sent to show SYNC monitoring works correctly again (Error Code:0x0000, Error Register: 0x81, Additional Code 00 04 00 00 00).

7.8 Node Guarding

Node Guarding starts when the first remote transmit request message (RTR) is received on the COB-ID for Node Guarding (0x700+ Module-ID).
If device doesn't receive corresponding message, Node Guarding isn't monitored. Default configuration provides Node Guarding is deactivated (Guard Time 0x100C=0, Life Time Factor 0x100D=0).
NMT master polls other devices at regular intervals, triggered by Guard Time 0x100C, and reply message contains device internal status.
In a event of an RTR request with Guard Time not set, Node Guarding isn't monitored, anyway device replies with its internal status.

Status codes:

CODE	STATUS
127	Pre-Operational
5	Operational
4	Stopped

If Node Guarding message isn't received within Life Time, a blink code is provided. An emergency message (Error Code:0x8130, Error Register: 0x11, Additional Code 00 04 00 00 00) is sent and device changes to pre-defined status according to object 0x67FE.
As soon as Node Guarding is restored, another emergency message is sent (Error Code:0x0000, Error Register: 0x11, Additional Code 00 04 00 00 00), and device status doesn't change.
N.B. It's only possible to use Node Guarding protocol or Heartbeat protocol.

7.9 Heartbeat Monitoring

Heartbeat generator cyclically provides a message (triggered by object 0x1017). During this time it transmits device status. Monitoring start when the first Heartbeat message occurs.
If corresponding Heartbeat message isn't received within time configured in object 0x1016, a blink code is provided. An emergency message (Error Code:0x8130, Error Register: 0x11, Additional Code 00 05 JJ 00 00, where JJ is the node number which has triggered EMCY) is sent and device changes to pre-defined status according to object 0x67FE.
As soon as Heartbeat protocol is restored, another emergency message is sent (Error Code:0x0000, Error Register: 0x11, Additional Code 00 05 JJ 00 00) to display Heartbeat works correctly again, and device status doesn't change.
Heartbeat protocol is always used if (and only if) producer time is configured (0x1017).

7.10 Emergency

There are four type of event which provides emergency messages:

- Critical error situation occurred / overcome in the device
- Important information has to be communicated to other devices
- Restore from an error
- Power-on with loaded settings equal to default settings (when setting haven ‘t yet been saved or when saved settings were discarded by device)

Structure of emergency object are shown in the table below:

Error Code	Error Register	Additional Code	Meaning
0x0000	**0x00**	**00 00 00 00 00**	Pre-defined Error Field 0x1003 Subindex0 set to 0 or all error are cleared
0x5000	**0x81**	**00 01 00 00 00**	Changed hardware configuration after power-on or reset node / communication
0x5000	**0x81**	**00 02 00 00 00**	Flash errors An error has occurred when configuration has been saved in flash memory
0x5000	**0x81**	**00 03 AA BB CC**	Programmed configuration doesn’t coincide with actual one AA: physical module where error has occurred BB: logic module where error has occurred CC: Cause of error
0x5000	**0x81**	**00 09 00 00 00**	Queue overflow for emergency messages
0x8100	**0x81**	**00 04 00 00 00**	Time between two SYNC is longer than Communication Cycle Period
0x8110	**0x11**	**00 01 00 00 00**	Internal receive buffer overflow Status changes as defined in object 0x67FE
0x8110	**0x11**	**00 02 00 00 00**	Internal transmit buffer overflow Status changes as define din object 0x67FE
0x8120	**0x11**	**00 03 00 00 00**	CAN Controller in Error Passive Mode
0x8130	**0x11**	**00 04 00 00 00**	Time between two Node Guarding telegrams is greater than Guard Time x Life Time Factor

0x8130	**0x11**	**00 05 DD 00 00**	Time between two Heartbeat telegrams is greater than configured one DD: node has tripped time overflow
0x8210	**0x81**	**00 05 EE FF GG**	PDO was sent with a number of bytes smaller than configured one in communication profile PDO data is discarded EE: configured value FF: actual value, number of bytes sent GG: number of PDO
0x8220	**0x81**	**00 06 HH II JJ**	PDO was sent with a number of bytes larger than configured one in communication profile Only the first n data is used (n = total length configured in object dictionary) HH: configured value II: actual value, number of bytes sent JJ: number of PDO
0xFF00	**0x81**	**00 06 KK 00 00**	Module bus error Status is changed to Stopped PP: Module position
0xFF00	**0x81**	**LL 07 MM NN PP**	Diagnosis message LL: diagnosis byte MM: Module position NN: Error status and channel number PP: Number of actual module error

8 NOTE / UPDATES

PIXSYS
Via Tagliamento, 18
30030 Mellaredo di Pianiga (VE)
www.pixsys.net
e-mail: sales@pixsys.net - support@pixsys.net

Software Rev. 1.10

2300.10.078-RevE 200709