PC10
CANOPEN SLAVE

Features

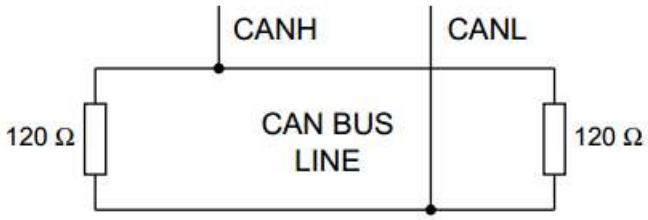
CANopen BUS
12V and 24V power supply supported
Electronic protection from short circuit
4 bypass circuits for emergency
Up to 12 power outputs with current sense
3 high current outputs
Up to 8 digital inputs
Up to 8 analog inputs
Up to 4 low side outputs
Operating temperature range: -20 to +85°C
Splash proof (IP65)
Molex connectors MX150L series

THE PRESENT MANUAL IS FOR REFERENCE ONLY AND MIGHT BE NOT UP TO DATE TO THE LATEST VERSION.PLEASE CONTACT US FOR GETTING THE MOST UPDATED FILE

Table of contents

1. How to Connect CAN bus:

J3 PIN	FUNCTION
1	CAN L
2	CAN H
3	Negative Battery (GND)
4	J3 Power Output (May be set as bus power)



Each end of the CAN bus is terminated with 120Ω resistors in compliance with the standard to minimize signal reflections on the bus. You may need to place a 120Ω resistor between CAN-L and CAN-H.

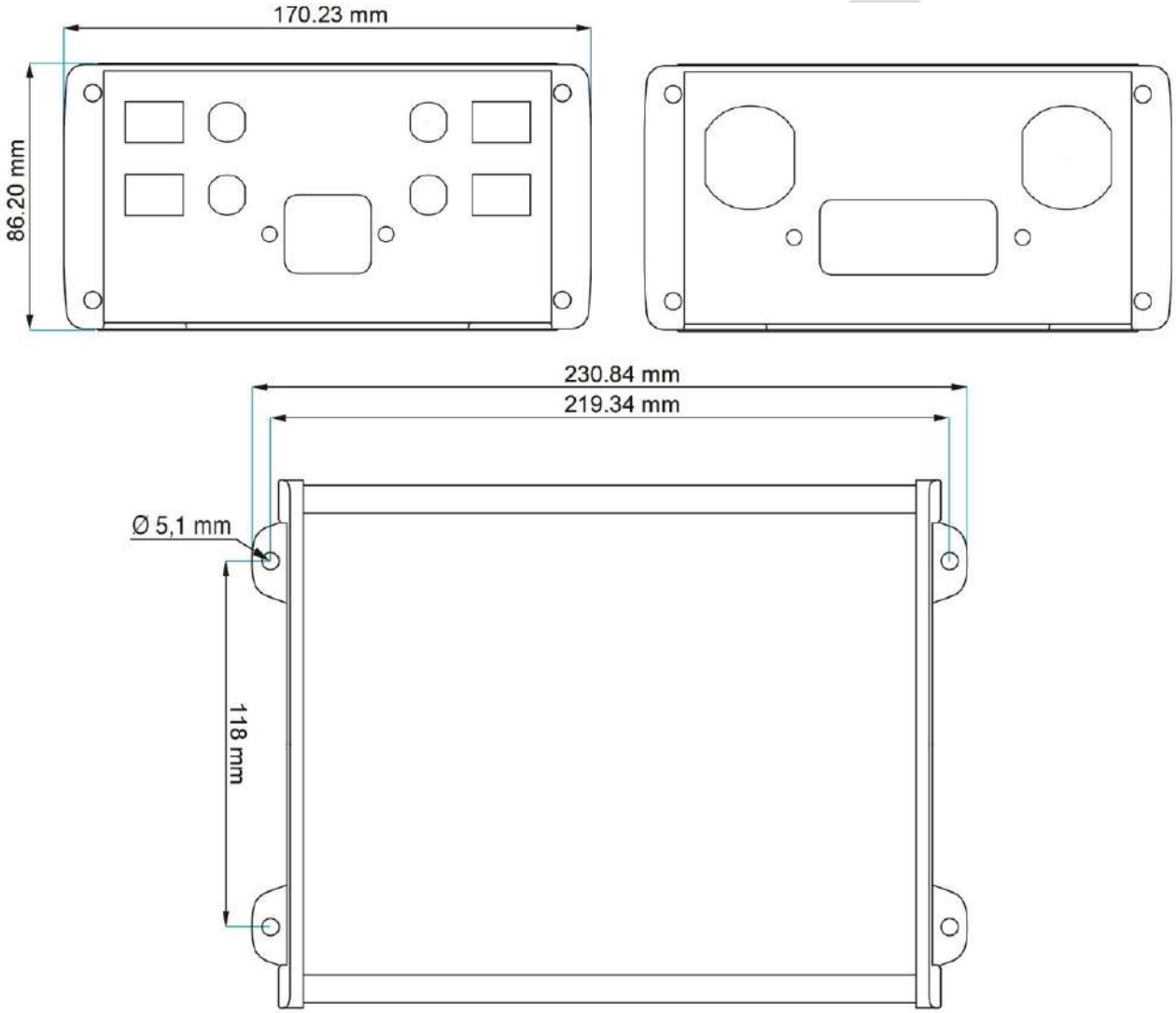
2. PC10 Technical Specifications

Electrical	Value	Unit
Supply Voltage (Battery Voltage)	8-32	V
Maximum input current	100	A
Maximum current single pin high side	20 or 15 see pin assignment table	A
Maximum current single pin low side	20 or 15 see pin assignment table	A
Maximum Digital input voltage	Battery Voltage	V
Digital input low voltage max	5,5	V
Analog input voltage	0-14	V
Inverse supply polarity protection	Not protected	N/A
Environmental	**Value**	**Unit**
Storage temperature range	-40 to +85	°C
Operating temperature range	-20 to +85	°C
Humidity	0 to 98	%

3. Mechanical Features

- Extruded aluminum housing
- End-panel power and I/O connections 2 x 5,7mm power lugs Amphenol SurLok Plus™ with sealed plug.
- 16 pin Molex MX150L connectors (Molex part # 19427-0049). Mates with 16 pin receptacle (Molex part # 19418-0030) using 14-16 gauge contact (Molex part #19420-0009)
- One 4 pin can bus connector Molex MX150L (Molex #19427-0032). Mates with 4pin receptacle (Molex part # 19418-0004) using 14-16 gauge contact (Molex #19420-0009)

Mechanical Dimension



4. Electrical Loads Connection

HIGH side switch:

The electrical device is connected between an output pin of the PowerCore and ground.

The output pin state can be HIGH (pin voltage at battery voltage) or OFF (pin floating).

The ground can be connected either to one of the ground pins of the PowerCore or directly to the battery's negative pole.

The electrical device is ON when the pin state is HIGH and the current consumption of the electrical device is read by the PowerCore.

LOW side switch:

The electrical device is connected between an output pin of the PowerCore with Low Side Drive feature and the positive battery pole.

The output pin state can be LOW (pin voltage at ground) or OFF (pin floating).

The current flowing in the low side switch is not read by the PowerCore. A 25A fuse protects the circuit, but a lower current threshold is needed, the connection to the battery's positive pole should be made through a fuse.

The electrical device is ON when the pin state is LOW.

H-Bridge connection:

The electrical device is connected between two output pins of the PowerCore.

The state of each output pin can be HIGH (pin voltage at battery voltage), LOW (pin voltage at ground) or OFF (pin floating).

This connection is common for DC motor to invert the power supply polarity to change the direction of the motor's rotation.

5. Bypasses

The bypasses are intended a backup for the system, in the unlikely case of a system failure. Every load controller is equipped with 4 ON/OFF rocker switches which, when set to ON position, bypasses the electronic circuitry, to guarantee that said channel will be able to operate even in the remote case of a general failure of the electronic board.

Bypass circuits are protected by cartridge fuses 6,3 x 32 mm.

Default configuration is no fuses in fuse holders.

Bypass 1	Bypass 2	Bypass 3	Bypass 4
J1-3	J1-5	J1-9	J1-16

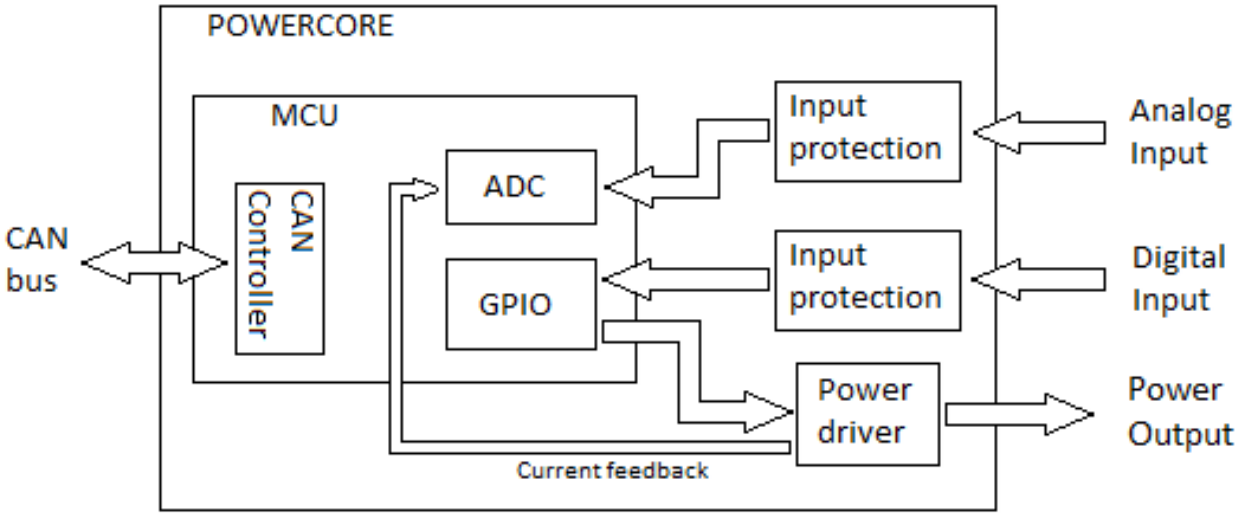
6. Pin Assignment

Connector	Pin	Function	Rating	High Current Driver	Low Side Driver	Level Sense	Analog Sense	Optional Relay Contact	Optional Diode Protection	By-Pass
J1	1	I/O	15		X		X		X	
J1	2	I/O	15		X	X	X		X	
J1	3	I/O	20	X		X	X		X	X
J1	4	I/O	15				X		X	
J1	5	I/O	15			X	X		X	X
J1	6	I/O	15			X	X		X	
J1	7	I/O	15			X	X		X	
J1	8	I/O	15			X	X		X	
J1	9	I/O	20	X	X	X			X	X
J1	10	O	15					X	X	
J1	11	O	15					X	X	
J1	12	GND	NA							
J1	13	GND	NA							
J1	14	GND	NA							
J1	15	GND	NA							
J1	16	I/O	20	X	X	X			X	X
J3	1	CAN L	NA							
J3	2	CAN H	NA							
J3	3	GND	NA							
J3	4	BUS PWR	15							

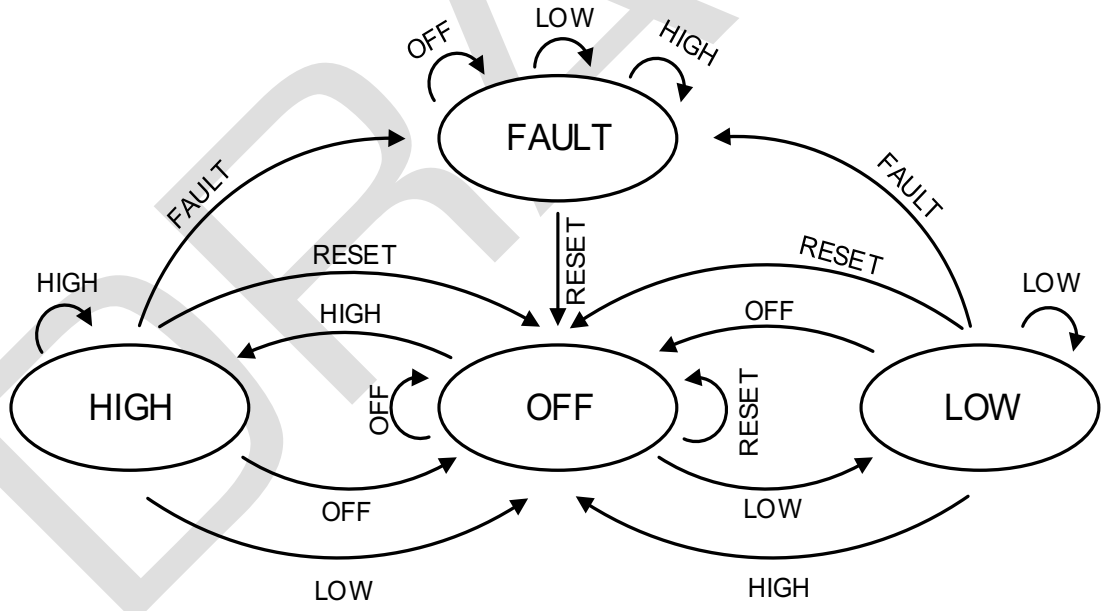
Function Summary

- **O:** Output high side pin
- **I/O:** Input / output high side pin
- **GND:** Internally connected to Negative terminal of the batteries
- **BUS PWR:** Power supply for the bus
- **CAN-L, CAN-H:** CAN bus signals.

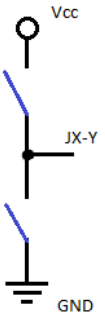
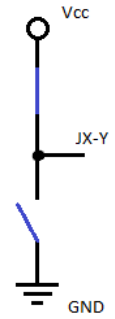
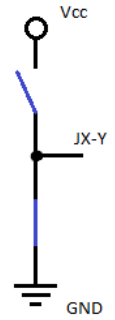
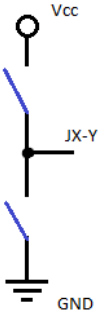
7. Hardware Block Diagram



8. Output State Machine Diagram



The output pin can be set to one of the 4 states available:

Output state	Pin voltage	Equivalent Circuit
OFF	Floating	
HIGH	Battery Voltage	
LOW	Ground Voltage	
FAULT	Floating	

To avoid cross conduction, the system does not allow to change the state of an output pin from HIGH to LOW and from LOW to HIGH. The pins must be always set to OFF for at least 300 milliseconds before the polarity can be changed.

As a measure of protection when using inductive loads, longer delays are added so a current peak during the transient period does not active the system's protection against overcurrent.

Each output circuit is protected against short circuits and overcurrent. If a failure event occurs, the pin goes into the FAULT state. To exit the FAULT state, a reset command is needed to return the pin to its initial OFF state.

The LOW state is applicable only on the output pins that have a low side driver (refer to the pin assignment table).

9. CANopen Messages Structure

All the data type used are unsigned integer and the syntax is specified in the following table:

octet number	1.	2.	3.	4.	5.	6.	7.	8.
UNSIGNED8	b7..b0							
UNSIGNED16	b7..b0	b15..b8						
UNSIGNED24	b7..b0	b15..b8	b23..b16					
UNSIGNED32	b7..b0	b15..b8	b23..b16	b31..b24				
UNSIGNED40	b7..b0	b15..b8	b23..b16	b31..b24	b39..b32			
UNSIGNED48	b7..b0	b15..b8	b23..b16	b31..b24	b39..b32	b47..b40		
UNSIGNED56	b7..b0	b15..b8	b23..b16	b31..b24	b39..b32	b47..b40	b55..b48	
UNSIGNED64	b7..b0	b15..b8	b23..b16	b31..b24	b39..b32	b47..b40	b55..b48	b63..b56

NMT MESSAGES

The Network Management messages follow a master-slave structure. Through NMT services, CANopen devices are initialized, started, reset, or stopped. All CANopen devices are regarded as NMT slaves.
NMT messages have CAN-ID always equal to 00h.

00h	1-byte command specifier	1-byte NODE-ID	6 bytes not used

PDO MESSAGES

PDO are fast telegram messages that can simply manage the most important functions. All PDOs have an equivalent SDO message. There are no answers for this type of messages. PDO messages have identifiers from 180h to 57Fh.

Identifier	8-byte data

SDO MESSAGES

SDO are more complex messages that completely manage all the functions of the PowerCore. SDO messages have identifiers from 580h to 67Fh and always expect an answer or an acknowledge reply.

Identifier	Command byte	2-byte index	1-byte sub index	4-byte data

Identifier: The messages to the PowerCore shall have 600h+current CAN ID identifier.
The messages from the PowerCore have 580h+ current CAN ID identifier.

Command byte:

40h:	request to read a register	60h:	write acknowledge
43h:	response with 4-byte data	23h:	request to write 4-byte data
4Fh:	response with 1-byte data	2Fh:	request to write 1-byte data
80h:	error response		

Every answer has index and sub index echo.
The error responses have the byte data containing the abort codes.

Abort codes implemented:

0602 0000h: Object does not exist in the object dictionary
0609 0011h: Sub-index does not exist
0609 0030h: Invalid value for parameter
0601 0002h: Attempt to write a read only object

10. CAN Messages for Managing Output Pins

Output state	Direction	Can message
HIGH	Write	PDO 300h + node-ID
	Read/Write	SDO Object 2001h
	Read/Write	SDO Object 2008h sub-index 02h
LOW	Read/Write	SDO Object 2001h
	Read/Write	SDO Object 2008h sub-index 03h
OFF	Write	PDO 200h + node-ID
	Read/Write	SDO Object 2001h
	Read/Write	SDO Object 2008h sub-index 05h
FAULT	Read when occur	PDO 180h + node-ID
	Read	SDO Object 2001h
	Read	SDO Object 2008h sub-index 06h
RESET	Write	SDO Object 2001h
	Write	SDO Object 2008h sub-index 07h
PWM	Write	SDO Object 2008h sub-index 04h

11. CAN bus Default Settings

Setting	Default State or Level	How to Change
Baud rate	125 kbit/s	Object 2010h
CANopen Node ID	15h	Object 2011h
CANopen Node State	Pre-operational	NMT Message Start CANopen node Object 2013h Device active on startup
J3-4 CAN bus power	ON	Object 2001h and 2014h
J3-4 CAN bus current threshold	5A	Object 2002h
JX-Y output current threshold	10A	Object 2002h
Periodic Message Transmission	Disable	Object 2012h
Heartbeat Message	Disable	Object 1017h
Boot-up service	Active	Object 2015h

NMT MESSAGES

The Network Management Messages follow a master-slave structure. Through NMT services, CANopen devices are initialized, started, reset, or stopped. All CANopen devices are regarded as NMT slaves.
NMT messages have CAN-ID always equal to 00h.

12. Start CANopen Node

Identifier	00h	
Byte 0	01h	Start CANopen node
Byte 1	XXh	PowerCore CAN ID 00h: start all the devices 15h: start the PowerCore with CAN ID = 15h.
Byte 2, 7	00h	Not used

Example:

Direction	**Identifier**	**Format**	**Message**
To PowerCore	0	Std	01 15

13. Enter Pre-operational

Identifier	00h	
Byte 0	80h	Enter pre-operational
Byte 1	XXh	PowerCore CAN ID 00h: start all the devices 15h: start the PowerCore with CAN ID = 15h.
Byte 2, 7	00h	Not used

Example:

Direction	**Identifier**	**Format**	**Message**
To PowerCore	0	Std	80 15

14. Reset CANopen Node

Identifier	00h	
Byte 0	81h	Reset CANopen node
Byte 1	XXh	PowerCore CAN ID 00h: start all the devices 15h: start the PowerCore with CAN ID = 15h.
Byte 2, 7	00h	Not used

Example:

Direction	**Identifier**	**Format**	**Message**
To PowerCore	0	Std	81 15

15. Stop CANopen Node

Identifier	00h	
Byte 0	XXh	02h: Stop CANopen node
		00h: Stop CANopen node (old SW compatibility)
Byte 1	YYh	PowerCore CAN ID 00h: Stop all the devices 15h: Stop the PowerCore with CAN ID = 15h.
Byte 2, 7	00h	Not used

Example:

Direction	**Identifier**	**Format**	**Message**
To PowerCore	0	Std	02 15

16. Boot-up Service

This service is used to signal that a NMT slave has entered the NMT Pre-operational state.

Identifier	**700h + current CAN ID**	**Default 715h**
Byte 0	00h	One data byte is transmitted with value 0.

Example:

Direction	**Identifier**	**Format**	**Message**
From PowerCore	715h	Std	00h

The PowerCore with CAN ID 15h has entered the NMT state Pre-operational.

17. Heartbeat Message

The heartbeat mechanism for a CANopen device is established by configuring the heartbeat producer to cyclically transmitting the heartbeat message. One or more CANopen devices in the network should be aware of this message. If by any chance the heartbeat producer fails to send the heartbeat message, the local application on the heartbeat consumer will be notified.

In the case the CANopen device starts with a heartbeat producer time different from 0, then the boot-up message is regarded as the first heartbeat message.

Identifier	700h + current CAN ID	Default 715h
Byte 0	XXh	XXh: State of heartbeat producer 00h: Boot-up 04h: Stopped 05h: Operational 7Fh: Pre-operational

Example:

Direction	**Identifier**	**Format**	**Message**	**Data**
From PowerCore	715h	Std	00h	Boot up
From PowerCore	715h	Std	7Fh	Pre-operational
To PowerCore	00h	Std	01h 15h	Start PowerCore with CAN id =15h
From PowerCore	715h	Std	05h	Operational

18. Set Output HIGH

Identifier	300h + current CAN ID							Default 315h
Byte 0	J1-8	J1-7	J1-6	J1-5	J1-4	J1-3	J1-2 J1-1	'1'= set HIGH '0'= do nothing
Byte 1	0	0	0	J1-11	J1-10	J3-4	J1-16 J1-9	'1'= set HIGH '0'= do nothing
Byte 2,7	00h							Not used

Example

Direction	Identifier	Format	Message	Data
To PowerCore	315	Std	01 00 00 00 00 00 00 00	Set J1-1 HIGH
To PowerCore	315	Std	00 03 00 00 00 00 00 00	Set J1-9 and J1-16 HIGH

19. Set Output OFF

Identifier	200h + current CAN ID							Default 215h
Byte 0	J1-8	J1-7	J1-6	J1-5	J1-4	J1-3	J1-2 J1-1	'1'= turn OFF '0'= do nothing
Byte 1	0	0	0	J1-11	J1-10	J3-4	J1-16 J1-9	'1'= turn OFF '0'= do nothing
Byte 2,7	00h							Not used

Example

Direction	Identifier	Format	Message	Data
To PowerCore	215	Std	04 00 00 00 00 00 00 00	Turn OFF J1-4
To PowerCore	215	Std	02 01 00 00 00 00 00 00	Turn OFF J1-2 and J1-9

20. Fault Message

This message is sent by the PowerCore when an overcurrent is detected.

Identifier	180h + current CAN ID							Default 195h
Byte 0	J1-8	J1-7	J1-6	J1-5	J1-4	J1-3	J1-2 J1-1	'1' = fault '0'= not fault
Byte 1	0	0	0	J1-11	J1-10	J3-4	J1-16 J1-9	'1' = fault '0'= not fault
Byte 2,7	00h							Not used

Examples

Direction	Identifier	Format	Message	Data
From PowerCore	195	Std	03 00 00 00 00 00 00 00	J1-1 and J1-2 Fault state
From PowerCore	195	Std	00 08 00 00 00 00 00 00	J1-10 Fault state

21. Object 2001h: Single Pin State

This object sets and reads the state of each output pins.

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
	2Fh	Write 1-byte data
Byte 1	01h	CAN Object 2001h
Byte 2	20h	
Byte 3	00h	Highest sub-index supported (read only)
	01h	J1-1
	02h	J1-2
	03h	J1-3
	04h	J1-4
	05h	J1-5
	06h	J1-6
	07h	J1-7
	08h	J1-8
	09h	J1-9
	0Ah	J1-16
	0Bh	J3-4 CAN bus power
	0Ch	J1-10
	0Dh	J1-11
Byte 4	00h	OFF
	01h	HIGH
	02h	LOW (on supported pins)
	03h	FAULT (read only)
	04h	RESET
Byte 5,7	00h	Not used

Examples:

Direction	Identifier	Format	Message	Data
To PowerCore	615	Std	40 01 20 00 00 00 00 00	Read highest sub-index supported
PowerCore reply	595	Std	4F 01 20 00 0D 00 00 00	0Dh is the highest sub-index supported
To PowerCore	615	Std	40 01 20 0C 00 00 00 00	Read state J1-10
PowerCore reply	595	Std	4F 01 20 0C 01 00 00 00	Pin J1-10 HIGH
To PowerCore	615	Std	40 01 20 09 00 00 00 00	Read state J1-9
PowerCore reply	595	Std	40 01 20 09 03 00 00 00	Pin J1-9 FAULT
To PowerCore	615	Std	2F 01 20 05 04 00 00 00	RESET pin J1-5
PowerCore reply	595	Std	60 01 20 05 00 00 00 00	ACK
To PowerCore	615	Std	2F 01 20 07 00 00 00 00	Set J1-7 to OFF
PowerCore reply	595	Std	60 01 20 07 00 00 00 00	ACK

22. Object 2002h: Output Current Threshold

This object sets and reads for each output pins the current thresholds. Refer to the pin assignment table to check the maximum currents of each pin. If you set a too high value a response error occurs, and the value is set to the maximum. The default thresholds are 5 A for J3-4 and 10 A for other outputs.

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
	2Fh	Write 1-byte data
Byte 1	02h	CAN Object 2002h
Byte 2	20h	
Byte 3	00h	Highest sub-index supported (read only)
	01h	J1-1
	02h	J1-2
	03h	J1-3
	04h	J1-4
	05h	J1-5
	06h	J1-6
	07h	J1-7
	08h	J1-8
	09h	J1-9
	0Ah	J1-16
	0Bh	J3-4 CAN bus power
	0Ch	J1-10
	0Dh	J1-11
Byte 4	XXh	XXh: Threshold value in ampere (A)
Byte 5,7	00h	Not used

Examples:

Direction	Identifier	Format	Message	Data
To PowerCore	615	Std	40 02 20 00 00 00 00 00	Read the highest sub-index supported
PowerCore reply	595	Std	4F 02 20 00 0D 00 00 00	0Dh is the highest sub-index supported
To PowerCore	615	Std	40 02 20 01 00 00 00 00	Read pin J1-1 threshold
PowerCore reply	595	Std	4F 02 20 01 0A 00 00 00	10A
To PowerCore	615	Std	40 02 20 0B 00 00 00 00	Read pin J3-4 threshold
PowerCore reply	595	Std	4F 02 20 0B 05 00 00 00	5A
To PowerCore	615	Std	2F 02 20 06 0F 00 00 00	Set pin J1-6 current to 15A
PowerCore reply	595	Std	60 02 20 06 00 00 00 00	ACK
To PowerCore	615	Std	2F 02 20 08 06 00 00 00	Set pin J1-8 current to 6A
PowerCore reply	595	Std	60 02 20 08 00 00 00 00	ACK

23. Object 2003h: Read Digital Input 8-bit

This object reads digital input values. The digital input is applicable only on some pins (refer to the pin assignment table). The unsupported pins have always the value '0'.

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
Byte 1	03h	CAN Object 2003h
Byte 2	20h	
Byte 3	00h	Highest sub-index supported
	01h	Read input J1-1 to J1-8
	02h	Read input J1-9 to J1-16
Byte 4,7	00h	Not used

PowerCore reply:

Identifier	595h (580h + current CAN ID)	
Byte 0	4Fh	Response length 1-byte
Byte 1	03h	CAN Object 2003h
Byte 2	20h	

Byte 3	00h	sub-index
Byte 4	02h	Highest sub-index supported
Byte 5,7	00h	Not used

Byte 3	01h	sub-index
Byte 4	J1-8 J1-7 J1-6 J1-5 0 J1-3 J1-2 0	Digital input
Byte 5,7	00h	Not used

Byte 3	02h	sub-index
Byte 4	0 0 0 0 0 0 J1-16 J1-9	Digital input
Byte 5,7	00h	Not used

24. Object 2004h: Read Analog Input

This object reads analog input values with 8-bit resolution. 14V=FFh.

Expected value: $(Vin \cdot 255/14)_h$

Note: this feature requires the installation of an additional component on the PowerCore.

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
Byte 1	04h	CAN Object 2004h
Byte 2	20h	
Byte 3	00h	Highest sub-index supported
	01h	Input J1-1
	02h	Input J1-2
	03h	Input J1-3
	04h	Input J1-4
	05h	Input J1-5
	06h	Input J1-6
	07h	Input J1-7
	08h	Input J1-8
Byte 4	(Vin ·255/14)h	Expected value
Byte 5,7	00h	Not used

Examples:

Direction	Identifier	Format	Message	Data
To PoweCore	615	Std	40 04 20 00 00 00 00 00	Read the highest sub-index supported
PowerCore reply	595	Std	4F 04 20 08 00 00 00 00	08h is the highest sub-index supported
To PowerCore	615	Std	40 04 20 01 00 00 00 00	Read J1-1
PowerCore reply	595	Std	4F 04 20 01 FF 00 00 00	14V
To PowerCore	615	Std	40 04 20 07 00 00 00 00	Read J1-7
PowerCore reply	595	Std	4F 04 20 07 B5 00 00 00	10V
To PowerCore	615	Std	40 04 20 08 00 00 00 00	Read J1-8
PowerCore reply	595	Std	4F 04 20 08 5A 00 00 00	5V

25. Object 2007h: Read PowerCore Electronic Values

This object reads the analog values of the output currents, the total current and the battery voltage.

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
Byte 1	07h	CAN Object 2007h
Byte 2	20h	
Byte 3	00h	Highest sub-index supported (read only)
	01h	J1-1 current Amp
	02h	J1-2 current Amp
	03h	J1-3 current Amp
	04h	J1-4 current Amp
	05h	J1-5 current Amp
	06h	J1-6 current Amp
	07h	J1-7 current Amp
	08h	J1-8 current Amp
	09h	J1-9 current Amp
	0Ah	J1-16 current Amp
	0Bh	J3-4 CAN bus power current Amp
	0Ch	J1-10 current Amp
	0Dh	J1-11 current Amp
	0Eh	Battery voltage: 35,5V = FFh
	0Fh	Total current
Byte 4,7	00h	Not used

Answer:

Identifier	**595h (580h + current CAN ID)**	
Byte 0	4Fh	1-byte response
Byte 1	07h	CAN Object 2007h
Byte 2	20h	
Byte 3	XXh	XXh: Sub index echo
Byte 4	YYh	YYh: Current [A] Voltage= YYd · 35,5/255
Byte 5,7	00h	Not used

26. Object 2008h: Output State

This object sets and reads the state of each output pin.

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
	23h	Write 4-bytes
Byte 1	08h	CAN Object 2008h
Byte 2	20h	
Byte 3	00h	Highest sub-index supported (read only)
	01h	Output OFF
	02h	Output HIGH
	03h	Output LOW
	04h	Output PWM (on supported pins)[1]
	06h	Output FAULT (read only)
	07h	Output RESET (write only)
Byte 4	J1-8 J1-7 J1-6 J1-5 J1-4 J1-3 J1-2 J1-1	Each bit sets a pin
Byte 5	0 0 0 J1-11 J1-10 J3-4 J1-16 J1-9	Each bit sets a pin
Byte 6	XXh (PWM state only)	Duty cycle level (D)[2] 00h-0Ah→0%-100%
	00h (other states)	Not used
Byte 7	00h	Not used

Examples:

Direction	**Identifier**	**Format**	**Message**	**Data**
To PowerCore	615	Std	23 08 20 02 08 00 00 00	Set J1-4 HIGH
PowerCore reply	595	Std	60 08 20 02 00 00 00 00	ACK
To PowerCore	615	Std	40 08 20 03 00 00 00 00	Who is LOW?
PowerCore reply	595	Std	43 08 20 03 00 00 00 00	No output LOW
To PowerCore	615	Std	40 08 20 01 00 00 00 00	Who is OFF?
PowerCore reply	595	Std	43 08 20 01 01 00 00 00	Only J1-1 OFF

[1]Note the PWM state is available only on the PINS controlled by BYTE 4 using a maximum of four simultaneously and is applicable to a signal whose frequency does not exceed 195Hz.

[2]With this parameter it is possible to adjust the power supplied to the outputs from 0% to 100% with variations of the 10% per level as shown in the table below:

Level	Duty cycle [%]	Value [h]
0	0 (OFF)	00
1	10	01
2	20	02
3	30	03
4	40	04
5	50	05
6	60	06
7	70	07
8	80	08
9	90	09
10	100 (ON)	0A

To PowerCore	615	Std	23 08 20 04 18 00 04 00	Set PWM on J1-4 and J1-5 with D=40%
PowerCore reply	595	Std	60 08 20 04 00 00 00 00	ACK
To PowerCore	615	Std	23 08 20 04 80 00 01 00	Set PWM on J1-8 with D=10%
PowerCore reply	595	Std	60 08 20 04 00 00 00 00	ACK

27. Object 2010h: Baud rate settings

This object sets and reads the baud rate.

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
	2Fh	Write 1-byte data
Byte 1	10h	CAN Object 2010h
Byte 2	20h	
Byte 3	00h	Sub index
Byte 4	XXh	XXh: 00h: 125k 01h: 250k 02h: 500k
Byte 5,7	00h	Not used

Examples:

Direction	**Identifier**	**Format**	**Message**	**Data**
To PowerCore	615	Std	40 10 20 00 00 00 00 00	Read baud rate
PowerCore reply	595	Std	4F 10 20 00 01 00 00 00	01h = 250k
To PowerCore	615	Std	2F 10 20 00 02 00 00 00	Set 02h = 500k
PowerCore reply	595	Std	60 10 20 00 00 00 00 00	ACK
To PowerCore	615	Std	2F 10 20 00 00 00 00 00	Set 00h = 125k
PowerCore reply	595	Std	60 10 20 00 00 00 00 00	ACK

28. Object 2011h: Set node ID

This object sets and reads the CANopen node-id.

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
	2Fh	Write 1-byte data
Byte 1	11h	CAN Object 2011h
Byte 2	20h	
Byte 3	00h	Sub index
Byte 4	XXh	New node id: (01h – 7Fh)
Byte 5,7	00h	Not used

Examples:

Direction	Identifier	Format	Message	Data
To PowerCore	615	Std	40 11 20 00 00 00 00 00	Read Node ID
PowerCore reply	595	Std	4F 11 20 00 15 00 00 00	ID = 15h
To PowerCore	615	Std	2F 11 20 00 16 00 00 00	Set ID = 16h
PowerCore reply	595	Std	60 11 20 00 00 00 00 00	ACK

29. Object 2012h: Set Periodic Messages

This object sets the periodic transmission of TPDO 180h (Fault message).

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
	2Fh	Write 1-byte data
Byte 1	12h	CAN Object 2012h
Byte 2	20h	
Byte 3	00h	Sub index
Byte 4	XXh	XXh: Period in milliseconds * 10 (00h = OFF, maximum 63h)
Byte 5,7	00h	Not used

Periodic transmission timer: XXh (from 00h to 63h: 0ms to 990ms)

Examples:

Direction	Identifier	Format	Message	Data
To PowerCore	615	Std	40 12 20 00 00 00 00 00	Read register
PowerCore reply	595	Std	4F 12 20 00 0A 00 00 00	Period = 100ms
To PowerCore	615	Std	2F 12 20 00 32 00 00 00	Set period = 500ms
PowerCore reply	595	Std	60 12 20 00 00 00 00 00	ACK

30. Object 2013h: Set Device Active on Startup

If device is active on startup don' t need start command from master.

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
	2Fh	Write 1-byte data
Byte 1	13h	CAN Object 2013h
Byte 2	20h	
Byte 3	00h	Sub index
Byte 4	XXh	00h: NOT ACTIVE 01h: ACTIVE
Byte 5,7	00h	Not used

Examples:

Direction	Identifier	Format	Message	Data
To PowerCore	615	Std	40 13 20 00 00 00 00 00	Read register
PowerCore reply	595	Std	4F 13 20 00 00 00 00 00	Not active at start
To PowerCore	615	Std	2F 13 20 00 01 00 00 00	Set active on start
PowerCore reply	595	Std	60 13 20 00 00 00 00 00	ACK

31. Object 2014h: Set CAN Bus Power Supply Active on Startup

This object sets and reads the pin J3-4 state at startup.

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
	2Fh	Write 1-byte data
Byte 1	14h	CAN Object 2014h
Byte 2	20h	
Byte 3	00h	Sub index
Byte 4	XXh	00h: NOT ACTIVE 01h: ACTIVE
Byte 5,7	00h	Not used

Examples:

Direction	Identifier	Format	Message	Data
To PowerCore	615	Std	40 14 20 00 00 00 00 00	Read register
PowerCore reply	595	Std	4F 14 20 00 00 00 00 00	Not active at start
To PowerCore	615	Std	2F 14 20 00 01 00 00 00	Set active on start
PowerCore reply	595	Std	60 14 20 00 00 00 00 00	ACK

32. Object 2015h: Set Boot-up Service

This object enables and disables the boot-up message.

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
	2Fh	Write 1-byte data
Byte 1	15h	CAN Object 2015h
Byte 2	20h	
Byte 3	00h	Sub index
Byte 4	XXh	00h: NOT ACTIVE 01h: ACTIVE
Byte 5,7	00h	Not used

Examples:

Direction	Identifier	Format	Message	Data
To PowerCore	615	Std	40 15 20 00 00 00 00 00	Read register
PowerCore reply	595	Std	4F 15 20 00 00 00 00 00	Not active at start
To PowerCore	615	Std	2F 15 20 00 01 00 00 00	Set active on start
PowerCore reply	595	Std	60 15 20 00 00 00 00 00	ACK

33. Object 1017h: Producer Heartbeat Time

The producer heartbeat time shall indicate the configured cycle time of the heartbeat.

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
	2Bh	Set device register
Byte 1	17h	CAN Object 1017h
Byte 2	10h	
Byte 3	00h	Sub index
Byte 4	YYh	YYh: Heartbeat time in milliseconds
Byte 5	XXh	XXh: Heartbeat time in milliseconds
Byte 6, 7	00h	Not used

Heartbeat time: XXYYh (from 000Ah to 03E7h: 10ms to 999 milliseconds)

Examples:

Direction	Identifier	Format	Message	Data
To PowerCore	615	Std	40 17 10 00 00 00 00 00	Read heartbeat time
PowerCore reply	595	Std	4B 17 10 00 64 00 00 00	Heartbeat time = 100ms
To PowerCore	615	Std	2B 17 10 00 00 00 00 00	Switch off the heartbeat
PowerCore reply	595	Std	60 17 10 00 00 00 00 00	
To PowerCore	615	Std	2B 17 10 00 32 00 00 00	Heartbeat time = 50ms
PowerCore reply	595	Std	60 17 10 00 00 00 00 00	
To PowerCore	615	Std	2B 17 10 00 F4 01 00 00	Heartbeat time = 500ms
PowerCore reply	595	Std	60 17 10 00 00 00 00 00	

34. Object 1000h: Device Type

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
Byte 1	00h	CAN Object 1000h
Byte 2	10h	
Byte 3	00h	Sub Index
Byte 4, 7	00h	Not used

Example:

Direction	Identifier	Format	Message
To PowerCore	615	Std	40 00 10 00 00 00 00 00
PowerCore reply	595	Std	43 00 10 00 91 01 87 00

Device profile number: 87191h generic I/O module.

I/O Functionality: digital I/O, analog Input.

Mapping PDOs: Device specific PDO supported.

35. Object 1400h: Receive PDO 0 Communication Parameter

Describes the RPDO parameters for turning OFF the outputs.

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
Byte 1	00h	CAN Object 1400h
Byte 2	14h	
Byte 3	00h	Highest sub-index supported
	01h	COB-ID used by RPDO
	02h	Transmission type
Byte 4,7	00h	Not used

Examples:

Direction	Identifier	Format	Message	Data
To PowerCore	615	Std	40 00 14 00 00 00 00 00	
PowerCore reply	595	Std	4F 00 14 00 02 00 00 00	02h
To PowerCore	615	Std	40 00 14 01 00 00 00 00	
PowerCore reply	595	Std	43 00 14 01 15 02 00 00	0000 0215h
To PowerCore	615	Std	40 00 14 02 00 00 00 00	
PowerCore reply	595	Std	4F 00 14 02 FE 00 00 00	FEh

Highest sub-index supported: 02h;
COB-ID used by RPDO: 0000 0215h; 0000 0200h + Node-ID;
Transmission type: event-driven (manufacturer-specific).

36. Object 1401h: Receive PDO 1 Communication Parameter

Describes the RPDO parameters for turning HIGH the outputs.

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
Byte 1	01h	CAN Object 1401h
Byte 2	14h	
Byte 3	00h	Highest sub-index supported
	01h	COB-ID used by RPDO
	02h	Transmission type
Byte 4,7	00h	Not used

Examples:

Direction	Identifier	Format	Message	Data
To PowerCore	615	Std	40 01 14 00 00 00 00 00	
PowerCore reply	595	Std	4F 01 14 00 02 00 00 00	02h
To PowerCore	615	Std	40 01 14 01 00 00 00 00	
PowerCore reply	595	Std	43 01 14 01 15 03 00 00	0000 0315h
To PowerCore	615	Std	40 01 14 02 00 00 00 00	
PowerCore reply	595	Std	4F 01 14 02 FE 00 00 00	FEh

Highest sub-index supported: 02h;
COB-ID used by RPDO: 0000 0315h; 0000 0300h + Node-ID.
Transmission type: event-driven (manufacturer-specific).

37. Object 1600h: Receive PDO 0 Mapping Parameter

Describes the RPDO 0 mapping parameters for turning OFF the outputs.

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
Byte 1	00h	CAN Object 1600h
Byte 2	16h	
Byte 3	00h	Highest sub-index supported
	01h	Application object: Set Output OFF
Byte 4,7	00h	Not used

Examples:

Direction	Identifier	Format	Message	Data
To PowerCore	615	Std	40 00 16 00 00 00 00 00	
PowerCore reply	595	Std	4F 00 16 00 01 00 00 00	01
To PowerCore	615	Std	40 00 16 01 00 00 00 00	
PowerCore reply	595	Std	43 00 16 01 10 01 08 20	2008 01 10

Highest sub-index supported: 01h;

Application object: set output OFF, index 2008h, sub-index 01h, length 10h.

38. Object 1601h: Receive PDO 1 Mapping Parameter

Describes the RPDO 1 mapping parameters for turning HIGH the outputs.

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
Byte 1	01h	CAN Object 1601h
Byte 2	16h	
Byte 3	00h	Highest sub-index supported
	01h	Application object: Set Output HIGH
Byte 4,7	00h	Not used

Examples:

Direction	Identifier	Format	Message	Data
To PowerCore	615	Std	40 01 16 00 00 00 00 00	
PowerCore reply	595	Std	4F 01 16 00 01 00 00 00	01
To PowerCore	615	Std	40 01 16 01 00 00 00 00	
PowerCore reply	595	Std	43 01 16 01 10 02 08 20	2008 02 10

Highest sub-index supported: 01h;

Application object: set output HIGH, index 2008h, sub-index 02h, length 10h.

39. Object 1800h: Transmit PDO 0 Communication Parameter

Describes the TPDO communication parameters for outputs FAULT state.

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
Byte 1	00h	CAN Object 1800h
Byte 2	18h	
Byte 3	00h	Highest sub-index supported
	01h	COB-ID used by TPDO
	02h	Transmission type
Byte 4,7	00h	Not used

Examples:

Direction	Identifier	Format	Message	Data
To PowerCore	615	Std	40 00 18 00 00 00 00 00	
PowerCore reply	595	Std	4F 00 18 00 02 00 00 00	02
To PowerCore	615	Std	40 00 18 01 00 00 00 00	
PowerCore reply	595	Std	43 00 18 01 95 01 00 00	0000 0195h
To PowerCore	615	Std	40 00 18 02 00 00 00 00	
PowerCore reply	595	Std	4F 00 18 02 FE 00 00 00	FEh

Highest sub-index supported: 02h;

COB-ID used by TPDO: 0000 0195h; 0000 0180h + Node-ID;

Transmission type: event-driven (manufacturer-specific).

40. Object 1A00h: Transmit PDO 0 Mapping Parameter

Describes the TPDO 0 communication parameters for read digital input 8 bit.

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
Byte 1	00h	CAN Object 1A00h
Byte 2	1Ah	
Byte 3	00h	Number of mapped objects
	01h	1st application object
	02h	2nd application object
Byte 4,7	00h	Not used

Examples:

Direction	Identifier	Format	Message	Data
To PowerCore	615	Std	40 00 1A 00 00 00 00 00	
PowerCore reply	595	Std	4F 00 1A 00 02 00 00 00	02
To PowerCore	615	Std	40 00 1A 01 00 00 00 00	
PowerCore reply	595	Std	43 00 1A 01 08 01 03 20	2003 01 08
To PowerCore	615	Std	40 00 1A 02 00 00 00 00	
PowerCore reply	595	Std	43 00 1A 02 08 02 03 20	2003 02 08

Number of mapped objects: 2;

Application object: Read digital input 8 bit;
Index 2003h; sub-index 01h; length 08h;
Index 2003h; sub-index 02h; length 08h.

41. Object 1A04h: Transmit PDO 4 Mapping Parameter

Describes the TPDO 4 communication parameters for reading output FAULT states.

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
Byte 1	04h	CAN Object 1A04h
Byte 2	1Ah	
Byte 3	00h	Number of mapped objects
	01h	1st application object
Byte 4,7	00h	Not used

Examples:

Direction	Identifier	Format	Message	Data
To PowerCore	615	Std	40 04 1A 00 00 00 00 00	
PowerCore reply	595	Std	4F 04 1A 00 01 00 00 00	01
To PowerCore	615	Std	40 04 1A 01 00 00 00 00	
PowerCore reply	595	Std	43 04 1A 01 10 06 08 20	2008 06 10

Number of mapped objects: 1;
Application object: Read output FAULT state;
Index 2008h; sub-index 06h; length 10h.

42. Object 1A05h: Transmit PDO 5 Mapping Parameter

Describes the TPDO 5 communication parameters for reading analog inputs.

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
Byte 1	05h	CAN Object 1A05h
Byte 2	1Ah	
Byte 3	00h	Number of mapped objects
	01h	1st application object: read analog input J1-1
	02h	2nd application object: read analog input J1-2
	03h	3rd application object: read analog input J1-3
	04h	4th application object: read analog input J1-4
	05h	5th application object: read analog input J1-5
	06h	6th application object: read analog input J1-6
	07h	7th application object: read analog input J1-7
	08h	8th application object: read analog input J1-8
Byte 4,7	00h	Not used

Examples:

Direction	Identifier	Format	Message	Data
To PowerCore	615	Std	40 05 1A 00 00 00 00 00	
PowerCore reply	595	Std	4F 05 1A 00 08 00 00 00	08
To PowerCore	615	Std	40 05 1A 01 00 00 00 00	
PowerCore reply	595	Std	43 05 1A 01 08 01 04 20	2004 01 08
To PowerCore	615	Std	40 05 1A 02 00 00 00 00	
PowerCore reply	595	Std	43 05 1A 02 08 02 04 20	2004 02 08
To PowerCore	615	Std	40 05 1A 03 00 00 00 00	
PowerCore reply	595	Std	43 05 1A 03 08 03 04 20	2004 03 08
To PowerCore	615	Std	40 05 1A 04 00 00 00 00	
PowerCore reply	595	Std	43 05 1A 04 08 04 04 20	2004 04 08
To PowerCore	615	Std	40 05 1A 05 00 00 00 00	
PowerCore reply	595	Std	43 05 1A 05 08 05 04 20	2004 05 08
To PowerCore	615	Std	40 05 1A 06 00 00 00 00	
PowerCore reply	595	Std	43 05 1A 06 08 06 04 20	2004 06 08
To PowerCore	615	Std	40 05 1A 07 00 00 00 00	
PowerCore reply	595	Std	43 05 1A 07 08 07 04 20	2004 07 08
To PowerCore	615	Std	40 05 1A 08 00 00 00 00	
PowerCore reply	595	Std	43 05 1A 08 08 08 04 20	2004 08 08

Number of mapped objects: 8;

Application object: Read analog input;
J1-1: Index 2004h; sub-index 01h; length 08h;
J1-2: Index 2004h; sub-index 02h; length 08h;
J1-3: Index 2004h; sub-index 03h; length 08h;
J1-4: Index 2004h; sub-index 04h; length 08h;
J1-5: Index 2004h; sub-index 05h; length 08h;
J1-6: Index 2004h; sub-index 06h; length 08h;
J1-7: Index 2004h; sub-index 07h; length 08h;
J1-8: Index 2004h; sub-index 08h; length 08h.

43. Object 1A06h: Transmit PDO 6 Mapping Parameter

Describes the TPDO 6 communication parameters for reading output currents J1-1..J1-8

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
Byte 1	06h	CAN Object 1A06h
Byte 2	1Ah	
Byte 3	00h	Number of mapped objects
	01h	1st application object: read output current J1-1
	02h	2nd application object: read output current J1-2
	03h	3rd application object: read output current J1-3
	04h	4th application object: read output current J1-4
	05h	5th application object: read output current J1-5
	06h	6th application object: read output current J1-6
	07h	7th application object: read output current J1-7
	08h	8th application object: read output current J1-8
Byte 4,7	00h	Not used

Examples:

Direction	Identifier	Format	Message	Data
To PowerCore	615	Std	40 06 1A 00 00 00 00 00	
PowerCore reply	595	Std	4F 06 1A 00 08 00 00 00	08
To PowerCore	615	Std	40 06 1A 01 00 00 00 00	
PowerCore reply	595	Std	43 06 1A 01 08 01 07 20	2007 01 08
To PowerCore	615	Std	40 06 1A 02 00 00 00 00	
PowerCore reply	595	Std	43 06 1A 02 08 02 07 20	2007 02 08
To PowerCore	615	Std	40 06 1A 03 00 00 00 00	
PowerCore reply	595	Std	43 06 1A 03 08 03 07 20	2007 03 08
To PowerCore	615	Std	40 06 1A 04 00 00 00 00	
PowerCore reply	595	Std	43 06 1A 04 08 04 07 20	2007 04 08
To PowerCore	615	Std	40 06 1A 05 00 00 00 00	
PowerCore reply	595	Std	43 06 1A 05 08 05 07 20	2007 05 08
To PowerCore	615	Std	40 06 1A 06 00 00 00 00	
PowerCore reply	595	Std	43 06 1A 06 08 06 07 20	2007 06 08
To PowerCore	615	Std	40 06 1A 07 00 00 00 00	
PowerCore reply	595	Std	43 06 1A 07 08 07 07 20	2007 07 08
To PowerCore	615	Std	40 06 1A 08 00 00 00 00	
PowerCore reply	595	Std	43 06 1A 08 08 08 07 20	2007 08 08

Number of mapped objects: 8;

Application object: Read output current;

J1-1: Index 2007h; sub-index 01h; length 08h;
J1-2: Index 2007h; sub-index 02h; length 08h;
J1-3: Index 2007h; sub-index 03h; length 08h;
J1-4: Index 2007h; sub-index 04h; length 08h;
J1-5: Index 2007h; sub-index 05h; length 08h;
J1-6: Index 2007h; sub-index 06h; length 08h;
J1-7: Index 2007h; sub-index 07h; length 08h;
J1-8: Index 2007h; sub-index 08h; length 08h.

44. Object 1A07h: Transmit PDO 7 Mapping Parameter

Describes the TPDO 7 communication parameters for reading output currents J1-9..J1-1, total current and battery voltage.

Identifier	600h + current CAN ID	Default 615h
Byte 0	40h	Read Device Register
Byte 1	07h	CAN Object 1A07h
Byte 2	1Ah	
Byte 3	00h	Number of mapped objects
	01h	1st application object: read output current J1-9
	02h	2nd application object: read output current J1-16
	03h	3rd application object: read output current J3-4
	04h	4th application object: read output current J1-10
	05h	5th application object: read output current J1-11
	06h	6th application object: read battery voltage
	07h	7th application object: read output total current
Byte 4,7	00h	Not used

Examples:

Direction	Identifier	Format	Message	Data
To PowerCore	615	Std	40 07 1A 00 00 00 00 00	
PowerCore reply	595	Std	4F 07 1A 00 07 00 00 00	07
To PowerCore	615	Std	40 07 1A 01 00 00 00 00	
PowerCore reply	595	Std	43 07 1A 01 08 09 07 20	2007 09 08
To PowerCore	615	Std	40 07 1A 02 00 00 00 00	
PowerCore reply	595	Std	43 07 1A 02 08 0A 07 20	2007 0A 08
To PowerCore	615	Std	40 07 1A 03 00 00 00 00	
PowerCore reply	595	Std	43 07 1A 03 08 0B 07 20	2007 0B 08
To PowerCore	615	Std	40 07 1A 04 00 00 00 00	
PowerCore reply	595	Std	43 07 1A 04 08 0C 07 20	2007 0C 08
To PowerCore	615	Std	40 07 1A 05 00 00 00 00	
PowerCore reply	595	Std	43 07 1A 05 08 0D 07 20	2007 0D 08
To PowerCore	615	Std	40 07 1A 06 00 00 00 00	
PowerCore reply	595	Std	43 07 1A 06 08 0E 07 20	2007 0E 08
To PowerCore	615	Std	40 07 1A 07 00 00 00 00	
PowerCore reply	595	Std	43 07 1A 07 08 0F 07 20	2007 0F 08

Number of mapped objects: 7;

Application object: Read output current;

J1-9: Index 2007h; sub-index 09h; length 08h;
J1-16: Index 2007h; sub-index 0Ah; length 08h;
J3-4: Index 2007h; sub-index 0Bh; length 08h;
J1-10: Index 2007h; sub-index 0Ch; length 08h;
J1-11: Index 2007h; sub-index 0Dh; length 08h;
Battery voltage: Index 2007h; sub-index 0Eh; length 08h;
Total current: Index 2007h; sub-index 0Fh; length 08h.

45. Revision history

Date	Manual Revision	Comment	Related SW version
27/05/2016	1.0	First release	
07/09/2018	1.1	New release	x.x