CANopen Reference Manual

Stepper Motor Positioning Controls and Plug & Drive motors

NANOTEC ELECTRONIC GmbH & Co. KG
Kapellenstraße 6
D-85622 Feldkirchen b. Munich, Germany

Tel. +49 (0)89-900 686-0
Fax +49 (0)89-900 686-50
info@nanotec.com

Editorial

Nanotec® Electronic GmbH & Co. KG
Kapellenstraße 6
D-85622 Feldkirchen b. Munich, Germany

Tel.: +49 (0)89-900 686-0
Fax: +49 (0)89-900 686-50

Internet: www.nanotec.com

Translation of the original operation manual

Version/Change overview

Version	Date	Changes
2.2	12.10.2010	Revision C+P
2.3	03.11.2011	Revision C+P
2.4	12.04.2012	Update of the object descriptions. New: ENCODER_TYPE (0x2011)

About this manual

Contents

This document contains a brief overview of the most important functions of the Nanotec stepper motor positioning controls and Plug & Drive motors as well as their use via CANopen.

Important information

Nanotec® reserves the right to make technical alterations and further develop hardware and software in the interests of its customers to improve the function of this product without prior notice.

For criticisms, proposals and suggestions for improvement, please contact the above address or send an email to: info@nanotec.com

Additional manuals

Please also note the following manuals from Nanotec:

NanoCAN User Manual	Configuration of the CAN communication for CANopen-capable controls with the NanoCAN software	
Programming manual	Control programming • Command reference • NanoJ • COM interface	
Technical manuals	Connection and commissioning of stepper motor controls or Plug & Drive motors	

The manuals are available for downloading at www.nanotec.com

Contents

1 Commissioning

1.1 Configuration of CAN communication

1.1.1 Rotary switch for CAN node ID

Function

In the case of controllers with a rotary switch, how the controller determines its baud rate and node ID can be selected with the rotary switch. A hexadecimal number is set with the rotary switches that is made up of the 1st digit (of the left rotary switch) and the 16th digit (of the right rotary switch).

The node ID of a CANopen node can be set to between 1 and 127. The controller reads it from the rotary switches or from the EEPROM when the voltage is applied.

For more details on writing the node ID and the baud rate via CAN objects, see section 1.1.3.

Settings

The dependency of the node ID and the baud rate on the value set on the rotary switch are listed in the following table.

Rotary switch value (decimal, SMCI47-S)	**0**	**1-127**	**128**	**129-255**
Rotary switch value (hex, SMCI47-S)	**0x00**	**0x01-0x7F**	**0x80**	**0x81-0xFF**
Rotary switch value (decimal, PD4-N)	**0**	**1-7**	**8**	**9-15**
Rotary switch value (hex, PD4-N)	**0x0**	**0x1-0x7**	**0x8**	**0x9-0xF**
Node ID of rotary switch value		X		X-128 or X-8
Node ID from EEPROM	X		X	
Baud rate fixed to 1 Mbaud	X	X		
Baud rate from EEPROM			X	X

Example

If the left-hand rotary switch is set to 1 and the right-hand rotary switch is set to 2, this results in the number 16*2 + 1 = 33.

Here the baud rate is set to 1 Mbaud and the node ID is 33.

PD4-N

The PD4-N has only one rotary switch. The node ID can, therefore, only be set in the range from 1 to 7. Other node IDs can only be set via SDO.

SMCI12 and PD2-N

The SMCI12 and PD2-N have no rotary switches due to the confined space conditions. The node ID and baud rate can only be set via SDO.

1.1.2 Configuration with the NanoCAN software

Note:
Information on the configuration can be found in the NanoCAN User Manual.

1.1.3 Configuration via CAN objects

Procedure

Proceed as follows to put a motor controller into operation using CAN objects:

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>1</td><td>Set the value 1 (left switch to 0, right switch to 1) with the two rotary switches.
This sets the baud rate permanently to 1 Mbaud and the node ID to 1. Invalid values for the baud rate in the EEPROM are thus ignored.</td></tr>
<tr><td>2</td><td>Make default settings of the firmware. To do so, write the string “load” in the “restore all default parameters” SDO (0x1011 Subindex 1).
<table>
<tr><th>COB ID</th><th>Data bytes</th><th>Description</th></tr>
<tr><td>601</td><td>23 11 10 01 6C 6F 61 64</td><td>Load defaults</td></tr>
<tr><td>581</td><td>60 11 10 01 00 00 00 00</td><td>Response: OK (takes up to 5 seconds)</td></tr>
</table>
All settings are reset to the presettings defined in the EDS, with the exception of the object for setting the baud rate (see Step 4).</td></tr>
<tr><td>3</td><td>Set the node ID with rotary switches.
The node ID of a CANopen node can be set to between 1 and 127. This node ID is read by the controller from the rotary switches when the voltage is applied.</td></tr>
<tr><td>4</td><td>If the set value of the rotary switch results in a number between 1 and 127, the CAN baud rate is set to 1 Mbaud.
To set a different baud rate:
Set the required baud rate via a CANboard with the aid of a service data object (SDO).
The object to be written has the SDO ID 0x2005 subindex 0x0, data type unsigned8.
The following table provides information about the baud rates that can be set:
<table>
<tr><th>Baud rate</th><th>Value for SDO 0x2005 Sub 0x0, type unsigned8</th></tr>
<tr><td>20 kbaud</td><td>130</td></tr>
<tr><td>50 kbaud</td><td>131</td></tr>
<tr><td>125 kbaud</td><td>132</td></tr>
<tr><td>250 kbaud</td><td>133</td></tr>
<tr><td>500 kbaud</td><td>134</td></tr>
<tr><td>1000 kbaud</td><td>135</td></tr>
</table></td></tr>
<tr><td>5</td><td>Write the value 0x65766173 (unsigned32) into the SDO 0x1010 subindex 0x1 in order to save the value in the internal EEPROM of the controller.
It takes approx. 5 seconds until the firmware has transferred all settings into the EEPROM and has sent the SDO response (see Step 2).</td></tr>
<tr><td>6</td><td>Disconnect the controller from the power supply.</td></tr>
</table>

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>7</td><td>Set the rotary switch to the value Node ID + 128 so that the saved value is adopted as the baud rate.
If values other than the values listed in the table are set for the baud rate, the controller does respond via the CAN interface but does not accept any travel instructions.</td></tr>
<tr><td>8</td><td>Save the settings in the EEPROM.
To do so, write the value (u32)0x65766173 into the object 0x1010 Subindex 1 (the value corresponds to the string “save”).
<table>
<tr><th>COB ID</th><th>Data bytes</th><th>Description</th></tr>
<tr><td>601</td><td>23 10 10 01 73 61 76 65</td><td>Save all settings</td></tr>
<tr><td>581</td><td>60 10 10 01 00 00 00 00</td><td>Response: OK</td></tr>
</table></td></tr>
</table>

1.2 Motor configuration

General information

Before commissioning the motor controller, the motor parameters in the controller must be adapted to the respective connected motor if they differ from the default settings.

These settings include:

- Phase and quiescent current of the motor (0x2004)
- Number of pole pairs of the motor (0x2006)
- Encoder resolution (0x608F)
- Gear factor (0x6091)

Phase and quiescent current

The object 0x2004 Subindex 1 is used to set the phase current as a percentage of the maximum current. The current is specified as a percentage of the nominal current (effective value). The default setting is controller-dependent.

The quiescent current as well as the phase current is set in percent of the maximum current with Subindex 2. This current is applied to the motor winding when the motor is at a standstill. The default setting is 20%.

Motor type

The motor type is set with the object 0x2004 Subindex 7. This setting is not supported by all controllers. Please also note the corresponding data sheet. Possible values:

- 0: Stepper motor with and without quadrature encoder
- 1: BLDC with Hall sensor and without quadrature encoder
- 2: BLDC with Hall sensor and with quadrature encoder

If a BLDC motor is used, the Hall sensor must also be configured (object 0x2004 Subindex C).

The value 0x243015 must be entered for the Hall configuration (default setting) for all Nanotec motors except for the DB42 series. The value 0x510342 applies for motors of the DB42 series.

Number of pole pairs

The number of pole pairs of the stepper motor determines its step angle.
The formula for the conversion is:

Step angle = 360° / (4 * number of pole pairs)

This means that a motor with a step angle of 1.8° has 50 pole pairs (default setting in the controller) and a 0.9° motor has 100 pole pairs.

The number of pole pairs must be correctly entered in the object 0x2006 Subindex 0.

Encoder resolution

If an encoder is used, its resolution must be known to the controller. The resolution is specified in increments per rotation. Because of the quadrature principle, an encoder with 500 marks per rotation, for example, has four time the number of increments per rotation: 2000.

This value must be entered in the object 0x608F Subindex 1.
The default value is 2000.

Gear factor

When using gearing, its translation or reduction ratio can be specified in the “Gear Ratio” object (0x6091). This ratio is given as a fraction: The number of motor revolutions is in Subindex 1 and the number of revolutions of the gearing on the drive side is in Subindex 2. The default setting is 1/1.

Conversion of position, speed and acceleration

If the motor parameters are correctly set, the user does not need to adjust the ramps, travel distances and speeds when changing the step mode or when changing from open loop to closed loop mode.

In PP, HM, and IP mode, the values for speed, acceleration and position refer to the “Feed Constant” object (feed constant, SDO 0x6092).

The numerator is saved in Subindex 1 of this object and the denominator of the feed constant is saved in Subindex 2. The default value is 2000/1. For the user, this means that, independent of the step mode, a path of 2000 as target position (SDO 0x607A) always corresponds to one revolution of the driving shaft.

For speed, this applies accordingly to a second and for the acceleration, this applies correspondingly to a second squared.

In VL mode the “VL Dimension Factor” object (SDO 0x604C) is used accordingly. In contrast to the other supported modes, however, the speeds or acceleration refer here to a minute or a minute squared.

Maximum speeds

Maximum speeds and accelerations (PP and HM mode) can be found in the following table.

Motor type	1.8°	0.9°		
Pole pairs (0x2006)	50	100		
Max. rpm	4800	2400		
Step mode	**Feed (0x6092 sub1) (steps per rotation)**	**Feed (0x6092 sub1) (steps per rotation)**	**Max. Profile Velocity (0x6081)**	**Max. Accel. (0x6083, 0x6084)**
Full step	200	400	16000	1600000
Half step	400	800	32000	3200000
Quarter step	800	1600	64000	6400000
Fifth step	1000	2000	80000	8000000
Eighth step	1600	3200	128000	12800000
Tenth step	2000	4000	160000	16000000
1/16 step	3200	6400	256000	25600000
1/32 step	6400	12800	512000	51200000
1/64 step	12800	25600	1024000	102400000

2 Getting started in CANopen mode

2.1 Overview

After the supply voltage is applied, the controller is not automatically ready for operation immediately.

The following steps have to be carried out first:

- Starting CANopen node
- Requesting CANopen node status
- Switching on the power drive
- Selecting the operation mode
- Starting a travel
- Specify new end position

The individual steps are described in detail in the following sections.

2.2 Starting CANopen node

“Operational” status

In order to use the features and functions of the controller, each time the controller is switched on it must be put into operational status.

This occurs by sending a network management message with the COB ID 0x0 and the 2-byte long contents: <Command> and <node ID>.

Input in IXXAT MiniMon

The full input in IXXAT MiniMon is: “0 1 22”.

- 0: COB ID for NMT message
- 1: Start node
- 22: CANopen node ID (here 0x22 or 34)

Commands

The commands are:

- 0x01: Start node (switches to operational, status 0x05)
- 0x02: Stop node (switches to stopped, status 0x04)
- 0x80: Change to pre-operational (status 0x7F, status after application of operating voltage)
- 0x81: Restart of the firmware, reset of all CANopen settings to the last values stored in the EEPROM
- 0x82: Restart of the firmware, reset of all CANopen settings to the last values stored in the EEPROM

2.3 Requesting CANopen node status

Status requests

The status can be requested with a Remote Transmission Request (RTR) of COB ID 0x700 + node ID.

A motor with the node ID 34(dec) sends its network status to COB ID 0x700 + 34 = 0x722.

In order to receive this message, a Remote Transmission Request (RTR) must be sent for this COB ID.

It is also possible to have the motor send this message cyclically (see SDO 0x1017: Dynamic Heartbeat Time).

Possible status

The following states are available:

- Pre-operational status (status after application of the operating voltage, after restart and reset): **0x7F**
  In this state SDOs can be requested and written, but no PDOs can be read or written.
- Stopped status: **0x04**
  In this mode neither SDOs nor PDOs can be requested.
- Operational status: **0x05**
  In this mode both SDOs and PDOs can be read and written.

2.4 Switching on the power drive

Control word

The power drive is switched on via the control word. This can be reached under the service data object (SDO) 0x6040.

Request of the status word

After sending each command, it is recommended to check whether the intended status has been reached as status transitions are prevented (e.g. by a undervoltage error) or can be delayed (e.g. by the delay time of the mechanical brake or through the execution time of internal transitions) by requesting the status word.

Input in IXXAT MiniMon

For activation, several transitions of the control words are necessary. The transitions are made as an input in IXXAT MiniMon for CANopen node ID 34:

A multiple transition during switch-on (e.g. from “Switch on Disabled” to “Operation Enabled“) is not possible. Only a jump from “Ready to Switch On” to “Operational” is possible.

COB ID	Data bytes	Description
622	2B 40 60 00 00 00	Switch On Disabled (basic state)
5A2	60 40 60 00 00 00 00 00	Response: OK
622	40 41 60 00	Request of the status word
5A2	4B 41 60 00 60 02 00 00	Response: Switch on Disabled

COB ID	Data bytes	Description
622	2B 40 60 00 06 00	Ready to Switch On
5A2	60 40 60 00 00 00 00 00	Response: OK
622	40 41 60 00	Request of the status word
5A2	4B 41 60 00 21 02 00 00	Response: Ready to Switch On

COB ID	Data bytes	Description
622	2B 40 60 00 07 00	Switch On (power drive on)
5A2	60 40 60 00 00 00 00 00	Response: OK
622	40 41 60 00	Request of the status word
5A2	4B 41 60 00 33 02 00 00	Response: Switch On, Voltage Enabled

COB ID	Data bytes	Description
622	2B 40 60 00 0F 00	Operation Enabled
5A2	60 40 60 00 00 00 00 00	Response: OK
622	40 41 60 00	Request of the status word
5A2	4B 41 60 00 37 02 00 00	Response: Operation Enabled, Voltage Enabled

Explanation of the data bytes

Explanation of the data to be entered in hex:

- 622: The COB ID for sending SDOs to the CANopen node 34(dec)
- 2B: Command for writing a SDO with 2 data bytes
- 40: Low-order byte of the SDO ID 0x6040
- 60: High-order byte of the SDO ID 0x6040
- 00: Subindex 0x00
- 00, 06, 07, 0f: Least significant byte of the control word

It is important to note that the data in CANopen are always transmitted in Intel notation, i.e. the low-order bytes first.

SDO addressing via CAN

Service data objects (SDO) are used to configure the respective CAN nodes. Because there are many different possible CANopen nodes (motors, sensors, etc.), a maximum of 2^16 = 65536 SDOs are provided in the CANopen standard.

The number of SDOs exceeds the number of COB-IDs in CAN 2^11 = 2048. Nevertheless, to be able to address the SDOs via CAN, the SDO ID is transferred as part of the data section of a COB.

In addition, 256 sub-indices are still possible for each SDO. If more than only Subindex 0 is occupied in a SDO, the number of available subindices of the respective SDOs is noted in Subindex 0.

2.5 Selecting the operation mode

Requirements

Changes of the mode can take place in the “Operation Enabled” status.

It is important to ensure that the motor does not move when commanding a mode change.

Example

The selection of a mode is shown using the example of PP mode (Profile Position or Positioning mode):

COB ID	Data bytes	Description
622	2F 60 60 00 01	Mode: Profile Position (PP)
5A2	60 60 60 00 00 00 00 00	Response: OK

2.6 Starting a travel

Step 1

Set the “new setpoint” bit in the control word to start a travel:

COB ID	Data bytes	Description
622	2b 40 60 00 1F 00	Start of a movement
5A2	60 40 60 00 00 00 00 00	Response: OK

Step 2

Reset the bit to enable subsequent travel movements:

COB ID	Data bytes	Description
622	2b 40 60 00 0F 00	Reset of the start bit
5A2	60 40 60 00 00 00 00 00	Response: OK

The motor should now have moved a little. The motor no longer reacts to any further toggling of the bit as it has reached its end position.

2.7 Specifying a new end position (0x12345)

Step 1

COB ID	Data bytes	Description
622	23 7A 60 00 45 23 01 00	End position to 0x12345
5A2	60 7A 60 00 00 00 00 00	Response: OK

Step 2

COB ID	Data bytes	Description
622	2b 40 60 00 1F 00	Start of a movement
5A2	60 40 60 00 00 00 00 00	Response: OK

Step 3

COB ID	Data bytes	Description
622	2b 40 60 00 0F 00	Reset of the start bit
5A2	60 40 60 00 00 00 00 00	Response: OK

3 Control and status word

3.1 Introduction

The control and status word are Service Data Objects (SDO) and are used to control and request the motor status. They are made up of individual bits.

The control and status word are mapped to Process Data Objects (PDO) as standard.

In the respective available modes (positioning, speed and reference mode) some bits always have the same meaning whereas others are specific to the respective mode.

Only those bits that have the same meaning in all modes are described.

3.2 Control word (SDO 0x6040)

Object description

Designation	Control word
SDO ID	0x6040
Type	u16
Value range	Bits

This object is available in the default setting as a PDO with the COB ID 0x200 + node ID. An RTR is not configured.

Bits 0, 1 and 3

Bit 0: Switch On

Bit 1: Enable Voltage

Bit 3: Enable Operation

The motor commands the state from Switch On Disabled to Operation Enabled with bits 0, 1 and 3 (see also section 2).

The states are:

- Switch On Disabled
- Ready to Switch On
- Switch on
- Operation Enabled

From the Switch On state, the holding brake is released and the electric field of the motor is active. A movement of the motor is only possible in the Operation Enabled state.

Additional states are:

- Quick Stop Active
- Fault Reaction Active
- Fault

The states from Switch On Disabled to Operation Enabled must be run through in the specified order. This occurs through the consecutive following setting of bits 0, 1 and 3. All three bits are set at the end of the switch-on process.

Bit 2

Quick Stop (inverted: 0 means Quick Stop Active)

Bit 2 must always be set to “1” unless a Quick Stop is required. If this bit is set to “0”, the motor carries out a Quick Stop. During the Quick Stop, the motor is in the state “Quick Stop Active”. After the Quick Stop, the motor automatically goes into the state “Switch On Disabled”.

Bits 4 to 15

Bits 4 to 6: Mode-specific.

Bit 7: Fault Reset.

If an error occurs, the firmware is in the Fault state after the error reaction. To set the firmware back to “Switch On Disabled”, this bit must carry out a transition from “0” to “1” (a duration of “1” is not sufficient here).

Bit 8: Stop (mode-specific).

Bit 9: Mode-specific.

Bit 10: Reserved.

Bits 11 to 15: Manufacturer-specific.

3.3 Status word (SDO 0x6041)

Object description

Designation	Status word
SDO ID	0x6041
Type	u16
Value range	Bits

This object is available as a PDO with the COB ID 0x180 + node ID in the default setting. An RTR is not configured. Each time the status word changes, this object is automatically sent as a PDO.

Bits

Bit 0: Ready to switch on

Bit 1: Switched on

Bit 2: Operation enabled: The selected operating Mode is active and accepts commands (e.g. Profile Position mode)

Bit 3: Fault: Set in case of a fault

Bit 4: Voltage enabled: Bit is set when the motor is provided with current

Bit 5: Quick stop

Bit 6: Switch on disabled

Bit 7: Warning

Bit 8: PLL sync complete: This bit is set as soon as the synchronization with the SYNC object is complete.

Bit 9: Remote

Bit 10: Target reached: This bit is set when the motor has reached its target (Profile Position Mode)

Bit 11: Internal limit active: This bit is set when the desired values exceed the maximum limits.

Bits 12, 13: Mode-specific

Bits 14, 15: Manufacturer-specific (not used)

4 Profile Position Mode (PP)

4.1 Introduction

Function

The Profile Position Mode is used to reach positions relative to the last target position or absolute to the last reference position with a predefined speed and ramp.

Carrying out travel orders

This mode also supports the sending of new travel orders to the controller although the current travel order is not yet complete.

Depending on the setting in the control word, the travel order is only carried out after the current order (no option) or the new travel order interrupts the current travel order (“Change Set Immediately”). It is also possible that the current travel order is still carried out to the target position, but the transition to the new record is carried out as “flying” i.e. without reducing the speed (Change on Setpoint).

Position controller

If the position controller (closed loop) is active, the objects 0x6067 and 0x6068 are used in order to decide when the travel is regarded as complete. In addition, the SDO 0x6065 and 0x6066 are evaluated in order to display the following error in the status word.

If the position controller is not active (open loop), a position error correction after a travel can be carried out if an encoder is present. To do so, the settings in SDO 0x2004 Subindices 5 and 6 are used. If the error is greater than the tolerance set there, a correction travel is started.

4.2 Control word (SDO 0x6040)

Object description

Designation	Control word
SDO ID	0x6040
Type	u16
Value range	Bits

This object is available in the default setting as a PDO with the COB ID 0x200 + node ID. An RTR is not configured.

The motor is switched on and travel commands can be carried out with this object.

Bits

Bits 0, 1 and 3 are used for starting up the power drive. These are described in the General section.

Bit 2 is used to trigger an emergency stop. If it is set to “0”, the motor carries out a quick stop with the ramp set in SDO 0x6085. The motor then goes into “Switch On Disabled” (see general description of the control word)

Bit 4 starts a travel order. This is carried out on a transition of “0” to “1”.

Bit 5: If this bit is set to “1”, a travel order triggered by bit 4 is immediately carried out. If it is “0”, the travel order just being carried out is completed and only then is the next

travel order started. If no travel order is being carried out at the moment of starting, the next travel order is started immediately.

Bit 6: If “0”, the target position (SDO 0x607A) is absolute and if “1”, the target position is relative to the current position.

Bit 7: Error Reset (see general description of the control word)

Bit 8: Halt: If this bit is set to “0”, the motor accelerates to the target speed with the set ramp. If it is set to “1”, the motor brakes and comes to a stop.

Bit 9: If this bit is set and the new travel command should only be carried out after the end of the current travel command, the speed is only changed when the first target position is reached. This means that braking is not performed before the first destination is reached as the motor should not stop at this position.

Bits 10 to 15: Reserved, to be set to 0.

4.3 Status word (SDO 0x6041)

Object description

Designation	Status word
SDO ID	0x6041
Type	u16
Value range	Bits

This object is available as a PDO with the COB ID 0x180 + node ID in the default setting. An RTR is not configured. Each time the status word changes, this object is automatically sent as a PDO.

Bits

Bits 0 to 9, 11, 14, 15: See section 3 “Control and status word”.

Bit 10: Target reached: This bit is set to 1 when the last target has been reached (motor at a standstill)

Bit 11: Internal limit active: This bit is set when one of the nominal values exceeds internal range limits. (e.g. default speed is greater than the maximum possible speed)

Bit 12: Set-point acknowledge: This bit confirms receipt of a new target point. It is set and reset synchronously to the “New set-point” bit in the control word.

An exception is if a new travel is started when another travel has not yet been completed and the next travel should only be carried out after the end of the first travel. In this case, the bit is only reset when the command has been accepted and the controller is ready to carry out new travel commands. If a new travel order is sent, the latest travel order is overwritten although this bit is still set.

Bit 13: Following error: This bit is set in Closed-Loop mode if the following error is greater than the set limits (see chapter 9.32 and 9.33)

4.4 Quick Stop Option Code (SDO 0x605A)

Object description

Designation	Quick Stop Option Code
SDO ID	0x605A
Type	s16, ro
Value range	3

Specifies the behavior upon a Quick Stop. Braking is currently only supported with maximum current and subsequent change to “Switch On Disabled”.

4.5 Position Demand Value (SDO 0x6062)

Object description

Designation	Position Demand Value
SDO ID	0x6062
Type	s32 ro
Value range	s32

Specifies the current demanded position.

4.6 Target Position (SDO 0x607A)

Object description

Designation	Target Position
SDO ID	0x607A
Type	s32 rw
Value range	-100000000 to 100000000

Specifies the target position.

Depending on the command of the control word, the end position is interpreted as relative to the current position or absolute to the reference position.

The direction can be reversed with the object 0x607E (polarity).

4.7 Home Offset (SDO 0x607C)

Object description

Designation	Home Offset
SDO ID	0x607C
Type	s32 rw
Value range	s32

Specifies the difference between the zero position the application and the reference point the machine.

4.8 Profile Velocity (SDO 0x6081)

Object description

Designation	Profile Velocity
SDO ID	0x6081
Type	u32 rw
Value range	Depending on the Feed Constant

Specifies the maximum traveling speed in steps per second.

4.9 End Velocity (SDO 0x6082)

Object description

Designation	End Velocity
SDO ID	0x6082
Type	u32 rw
Value range	Depending on the Feed Constant

Specifies the minimum traveling speed for a trapezoidal ramp in steps per second.

4.10 Profile Acceleration (SDO 0x6083)

Object description

Designation	Profile Acceleration
SDO ID	0x6083
Type	u32
Value range	1 to 100000

Specifies the acceleration ramp in steps/s².

4.11 Profile Deceleration (SDO 0x6084)

Object description

Designation	Profile Deceleration
SDO ID	0x6084
Type	u32
Value range	1 to 100000

Specifies the braking ramp in steps/s².

4.12 Quick Stop Deceleration (SDO 0x6085)

Object description

Designation	Quick Stop Deceleration
SDO ID	0x6085
Type	u32
Value range	1 to 100000

Specifies the emergency stop braking ramp in steps/s².

4.13 Motion profile Type (0x6086)

Object description

Designation	Motion profile type
SDO ID	0x6086
Type	s16
Value range	0-1

Specifies the ramp type.

Currently only a Sin2 (value=1) and a linear/trapezoidal ramp is supported (value = 0).

5 Homing Mode

5.1 Control word (0x6040)

Object description

Designation	Control word
SDO ID	0x6040
Type	u16
Value range	Bits

This object is available in the default setting as a PDO with the COB ID 0x200 + node ID. An RTR is not configured.

The motor is switched on and travel commands can be started with this object.

Bits

Bits 0, 1 and 3 are used for starting up the power drive. See section 3 “Control and status word”.

Bit 2 is used to trigger an emergency stop. If it is set to “0”, the motor carries out a quick stop with the ramp set in SDO 0x6085. The motor then goes into “Switch On Disabled” (see section 3 “Control and status word”.)

Bit 4 starts the referencing. This is carried out until either the reference position has been reached or bit 4 is set to “0” again.

Bits 5 to 6: Reserved.

Bit 7: See section 3 “Control and status word”.

Bit 8: If this bit is set, the motor stops.

Bits 9 to 15: See section 3 “Control and status word”.

5.2 Status word (0x6041)

Object description

Designation	Status word
SDO ID	0x6041
Type	u16
Value range	Bits

This object is available as a PDO with the COB ID 0x180 + node ID in the default setting. An RTR is not configured. Each time the status word changes, this object is automatically sent as a PDO.

Bits

Bits 0 to 9, 11, 14, 15: See section 3 “Control and status word”.

Bit 10: Target reached: Set to “1” when the motor is at a standstill.

Bit 11: Unused.

Bit 12: Homing attained: Set to “1” when the reference position is reached.

Bit 13: Set to “1” when an error has occurred.

5.3 Homing Method (0x6098)

Object description

Designation	Homing Method
SDO ID	0x6098
Type	u8
Value range	19, 20, 21, 22 ,33, 34, 35, -2 to -7

This object selects the Homing mode.

Modes 21 and 22 are only available from firmware 15-12-2008 (SMCI47-S).

Mode 19: External reference travel – switch as normally closed

- Search of the switch
- Motor rotates in a clockwise direction
- Speed from object 0x6099_1 (Search for switch)
- As long as input 6 is high
- As soon as input 6 becomes low (switch reached) the direction is reversed
- Motor rotates in a clockwise direction
- Speed from object 0x6099_2 (Search for zero)
- Until input 6 is high again (switch free again)
- Motor stops

Mode 20: External reference travel – switch as normally open

- Search of the switch
- Motor rotates in a clockwise direction
- Speed from object 0x6099_1 (Search for switch)
- As long as input 6 is low
- When the switch is reached (input 6 high), the direction is reversed
- Motor rotates in a clockwise direction
- Speed from object 0x6099_2 (Search for zero)
- Until input 6 becomes low again
- Motor stops

Mode 21: External reference travel – switch as normally closed

- Search of the switch
- Motor rotates in a clockwise direction
- Speed from object 0x6099_1 (Search for switch)
- As long as input 6 is high
- When the switch is reached (input 6 low), the direction is reversed
- Motor rotates in a clockwise direction
- Speed from object 0x6099_2 (Search for zero)
- Until input 6 becomes high again
- Motor stops

Mode 22: External reference travel – switch as normally open

- Search of the switch

- Motor rotates in a clockwise direction
- Speed from object 0x6099_1 (Search for switch)
- As long as input 6 is low
- When the switch is reached (input 6 high), the direction is reversed
- Motor rotates in a clockwise direction
- Speed from object 0x6099_2 (Search for zero)
- Until input 6 becomes low again
- Motor stops

Mode 33: Internal reference run

- Search for the index mark of the internal encoder
- Motor rotates in a clockwise direction
- Speed from object 0x6099_2 (Search for zero)
- Until index mark is reached
- When the index mark is reached, the direction is reversed
- Motor rotates in a clockwise direction
- Motor shuts down as of the index mark
- Motor stops

Mode 34: Internal reference run

- Search for the index mark of the internal encoder
- Motor rotates in a clockwise direction
- Speed from object 0x6099_2 (Search for zero)
- Until the index mark is reached
- When the index mark is reached, the direction is reversed
- Motor rotates in a clockwise direction
- Motor shuts down as of the index mark
- Motor stops

Mode 35: Position reset

- Sets the current position to Home Offset without the shaft moving

Mode -2: Reference run set to blocking

- Mode only functions with an encoder (OL and CL)
- First run: Motor rotates in a clockwise direction with speed from object 0x6099_1 (Search for switch) until the shaft is blocked.
  The objects “Following Error Window” and “Following Error Timeout” are evaluated here.
- Motor moves backwards by one electrical revolution
- Second run: Motor rotates in a clockwise direction with speed from object 0x6099_2 (Search for zero) until the shaft is blocked.
  The objects “Following Error Window” and “Following Error Timeout” are evaluated here.
- Motor moves backwards by one electrical revolution
- Motor moves to precisely the blocked position of the second run and sets the position to ”Home Offset”

Mode -3: Reference run set to blocking

- Like mode -2, only counterclockwise.

Mode -4: Reference run to external IO node

- Like mode 19, only instead of input 6 an external IO node is used as limit switch. (see also SDO 0x2010)

Mode -5: Reference run to external IO node

- Like mode 20, only instead of input 6 an external IO node is used as limit switch. (see also SDO 0x2010)

Mode -6: Reference run to external IO node

- Like mode 21, only instead of input 6 an external IO node is used as the limit switch. (see also SDO 0x2010)

Mode -7: Reference run to external IO node

- Like mode 22, only instead of input 6 an external IO node is used as the limit switch. (see also SDO 0x2010)

5.4 Homing Speeds (0x6099)

Object description

Designation	Homing Speeds
SDO ID	0x6099
Type	2x u32
Value range	1 to 25000

Specifies the speeds for the Homing mode in steps/s.

The speed for the search of the switch is specified in Subindex 1.

The (lower) speed for the search for the reference position is specified in Subindex 2.

Notes

The speed in Subindex 2 is also the starting speed for starting the acceleration ramp. If this is set too high, the motor loses steps or does not rotate at all. An excessive setting also leads to the index marking being overlooked. This speed should, therefore, be less than 1000 steps per second.

In addition, the speed in Subindex 1 must be greater than the speed in Subindex 2.

See also Homing Method (SDO 0x6098, section 5.3)

5.5 Homing Acceleration (0x609A)

Object description

Designation	Homing Acceleration
SDO ID	0x609A
Type	u32
Value range	1 to 100000

Specifies the acceleration ramp for the Homing mode in steps/s². The ramp is only used when starting off. When the switch is reached, the unit is automatically switched to the lower speed and is stopped as soon as it reaches the limit position.

6 Velocity mode

6.1 Control word (0x6040)

Object description

Designation	Control word
SDO ID	0x6040
Type	u16
Value range	Bits

This object is available as a static PDO with COB ID 0x200 + node ID. An RTR is not possible.

The motor is switched on and travel commands can be started with this object.

Bits

Bits 0, 1 and 3 are used for starting up the power drive. These are described in the General section.

Bit 2 is used to trigger an emergency stop. If it is set to “0”, the motor carries out a quick stop with the ramp set in SDO 0x604A. The motor then goes into “Switch On Disabled” (see general description of the control word)

Bit 8: Halt: On a transition of “0” to “1” the motor accelerates up to the target speed with the set ramp.
On a transition of “0” to “1” the motor brakes and comes to a stop.

6.2 Status word (0x6041)

Object description

Designation	Status word
SDO ID	0x6041
Type	u16
Value range	Bits

This object is available as a static PDO with COB ID 0x180 + node ID. An RTR is possible. Each time the status word changes, this object is automatically sent as a PDO.

bit 11

Internal Limit Active: This bit is set when a nominal value exceeds or undercuts internal limits (e.g. target speed (SDO 0x6082) greater than 25000 steps/s).

6.3 VL target velocity (0x6042)

Object description

Designation	VL target velocity
SDO ID	0x6042
Type	s16
Value range	-25000 to 25000

Specifies the target speed in steps/s.

Values greater than 0 stand for right-hand rotation, values less than 0 stand for left-hand rotation.

The direction can be reversed with the “Polarity” object.

6.4 VL velocity demand (0x6043)

Object description

Designation	VL velocity demand
SDO ID	0x6043
Type	s16
Value range	-25000 to 25000

Specifies the current target speed in steps/s.

This object is read only.

6.5 VL velocity actual value (0x6044)

Object description

Designation	VL velocity actual value
SDO ID	0x6044
Type	s16
Value range	-25000 to 25000

Specifies the current actual speed in steps/s.

A value is only output when the closed loop is activated.

This object is read only.

6.6 VL velocity min max amount (0x6046)

Object description

Designation	VL velocity min max amount
SDO ID	0x6046
Type	2x u32
Value range	1 to 25000

The minimum speed and maximum speed in steps/s can be set with this object.

Subindex 1 contains the minimum speed.

Subindex 2 contains the maximum speed.

Notes

If the magnitude of a target speed (SDO 0x6042) is less than the minimum speed, the minimum speed applies. If the target speed is 0, the motor stops.

A target speed greater than the maximum speed sets the speed to the maximum speed and sets bit 11 (internal limit active) in the status word (SDO 0x6041).

6.7 VL velocity acceleration (0x6048)

Object description

Designation	VL velocity acceleration
SDO ID	0x6048
Type	u32, u16
Value range	u32, u16, both not equal 0

Sets the acceleration ramp in VL mode.

The acceleration is specified as a fraction:
Speed change per time change.

Subindex 1 contains the speed change in steps/s (u32).

Subindex 2 contains the time change in s (u16).

Note

Neither the numerator nor the denominator must be set to 0.

6.8 VL velocity deceleration (0x6049)

Object description

Designation	VL velocity deceleration
SDO ID	0x6049
Type	u32, u16
Value range	u32, u16, both not equal 0

Sets the braking ramp in VL mode.

6.9 VL velocity quick stop (0x604A)

Object description

Designation	VL velocity quick stop
SDO ID	0x604A
Type	u32, u16
Value range	u32, u16, both not equal 0

Sets the braking ramp for the quick stop in VL mode.

The acceleration is specified as a fraction:
Speed change per time change.

Subindex 1 contains the speed change in steps/s (u32).

Subindex 2 contains the time change in s (u16).

Note

Neither the numerator nor the denominator must be set to 0.

6.10 VL dimension factor (0x604C)

Object description

Designation	VL dimension factor
SDO ID	0x604C
Type	2x s32
Value range	s32, s32, both greater than 0

This object is a fraction which is used to convert the setpoint value presettings of the user in Velocity mode into rpm.
(In VL mode the motor controller indicates the internal rotational speed in rpm.)

The numerator is in Subindex 1 here and the denominator of this fraction is in Subindex 2.

Objects

The conversion is applied to the following objects:

- 0x6042 (VL Target Velocity)
- 0x6043 (VL Velocity Demand)
- 0x6044 (VL Velocity Actual Value)
- 0x6046 (VL Velocity min/max Amount)
- 0x6048 (VL Velocity Acceleration)
- 0x6049 (VL Velocity Deceleration)
- 0x604A (VL Velocity Quick Stop)

Conversion

The conversion is carried out according to the following principle:

User unit * conversion factor = unit in rpm

Example

At a value of 2000, the user-specific unit should lead to one revolution per second (default in the firmware).

The conversion factor is then:
60/2000 (SDO 0x604C Sub1/Sub2)

The firmware then executes the conversion as follows:
2000 * 60/2000 = 60 RPM (= 1 revolution per second)

For values that are updated by the firmware (SDO 0x6043 and 0x6044), the conversion runs accordingly so that the user can read the value from the SDO in the user-specific unit.

7 Interpolated Position mode

7.1 Introduction

Firmware

This mode is available as of firmware version 06-04-2009.

Function

The Interpolated Position mode is used to synchronize several axes. For this, a higher-order controller accepts the ramp or path calculation and transfers the respective demanded position at which the axis should find itself at a specific time to the controller. The controller interpolates between these position sampling points.

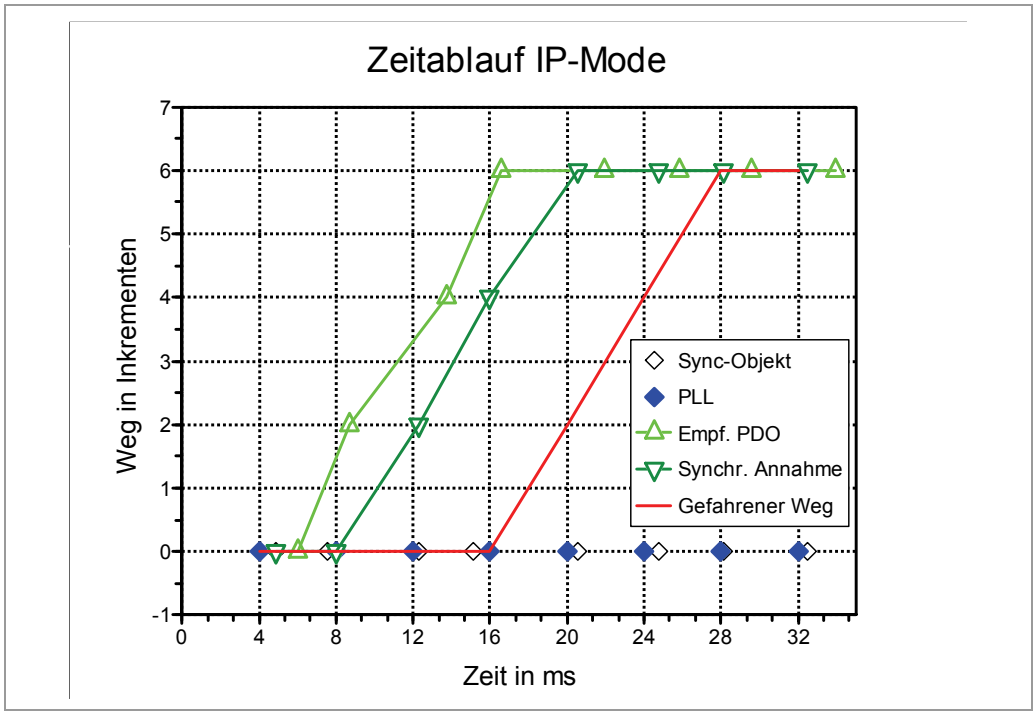
Synchronization to the SYNC object

For Interpolated Position mode, it is necessary that the controller is synchronized to the SYNC object. This SYNC object must be sent by the higher-order controller at regular time intervals. The synchronization is carried out as soon as the controller is switched to NMT “Operational” mode (see section 2.2).

7.2 Chronological progress

Representation

The following figure shows the time progression in Interpolated Position mode. The time is plotted on the x-axis and the position on the y-axis.



Explanations of the legend

Sync object

The transparent diamond shapes, plotted along the x-axis, mark the points in time at which the controller receives a Sync object from the higher-order controller.

The time intervals of the sync objects are not constant in this case. The deviation of the points in time that results when the Sync object is received at equal time intervals is called jitter. Jitter is caused by a high CAN bus load, for instance.

PLL

The controller compensates the jitter of the Sync object by synchronizing it to the Sync objects with the aid of a so-called “Phase Locked Loop” (PLL). The resulting points in time (solid diamond shapes on the x-axis) are almost free of jitter.

Rcvd. PDO

The triangles that point upwards indicate the points in time at which a new demanded position is received. The height of the triangles on the y-axis indicates the value of the demanded position. It is clear that the resulting curve between t=4ms and t=20ms is not a straight line although the position is always increased by two increments over the previous value. This occurs because of the time-related jitter that is also associated with the objects for the demanded position.

Synchr. acceptance

At the points in time which are marked with the downward pointing triangles, the controller takes the PDO with the demanded position into its internal buffer. This point in time is precisely the time at which the sync object is received. From this point in time, it would also be possible to read this value via the SDO protocol. This curve also is not a straight line due to the jitter of the received Sync objects.

Traveled path

The continuous line indicates the route traveled in Interpolated Position mode. Because the interpolation time points are derived from the PLL, the jitter of the sync object has no influence on the shape of the curve.

The time delay between the demanded position and the traveled path is necessary because the starting and demanded position of the current interval must already be known at the beginning of the interpolation period (here 2*4ms) of the controller in order to be able to calculate the speed this interval.

7.3 Control word (SDO 0x6040)

Object description

Designation	Control word
SDO ID	0x6040
Type	u16
Value range	Bits

The power drive can be started up and the IP mode activated with this object.

Bits

Bits 0, 1 and 3 are used for starting up the power drive. See section 3 “Control and status word”.

Bit 2 is used to trigger an emergency stop. If it is set to “0”, the motor stops abruptly. The motor then goes into “Switch On Disabled” (see section 3 “Control and status word”).

Bit 4 activates the IP mode.

7.4 Status word (SDO 0x6041)

Object description

Designation	Status word
SDO ID	0x6041
Type	u16
Value range	Bits

Bit12: IP mode active: This bit is set if the IP mode is active.

Remaining bits: See section 3 “Control and status word”.

7.5 Interpolation Sub-Mode Select (SDO 0x60C0)

Object description

Designation	Interpolation Sub-Mode Select
SDO ID	0x60C0
Type	s16
Value range	0

The interpolation mode is selected with this object. Only a linear interpolation (value 0) is supported. Other values are not allowed and will be ignored.

7.6 Interpolation Data Record (SDO 0x60C1)

Object description

Designation	Interpolation Data Record
SDO ID	0x60C1
Type	Array
Number of entries	1

This object contains the next target position to be moved to.

Subindex 1

Designation	1st Setpoint
SDO ID	0x60C1
Subindex	1
Type	s32
Value range	s32

The next target position is to be written in this subindex. When using the Interpolated Position Mode, it is recommended to map this object in a synchronous RPDO.

7.7 Interpolation Time Period (0x60C2)

Object description

In the controller, these objects are present for reasons of compatibility. Set values are ignored. In the controller the interpolation time is derived from the time intervals of the Synchronous object.

Designation	Interpolation Time Period
SDO ID	0x60C2
Type	Array
Number of entries	2

Subindex 1

Designation	Interpolation time period value
SDO ID	0x60C2
Subindex	1
Type	u8
Value range	u8

Subindex 2

Designation	Interpolation time index
SDO ID	0x60C2
Subindex	2
Type	s8
Value range	s8

7.8 Interpolation Data Configuration (0x60C4)

Object description

This object is used as a placeholder for future modes. Its values currently have no influence on the behavior of the Interpolated Position mode.

Designation	Interpolation Data Configuration
SDO ID	0x60C4
Type	Array
Number of entries	6

Subindex 1

Designation	Maximum Buffer Size
SDO ID	0x60C4
Subindex	1
Type	u32
Value range	1

Maximum length of the buffer

Subindex 2

Designation	Actual Buffer Size
SDO ID	0x60C4
Subindex	2
Type	u32
Value range	1

Actual length of the buffer

Subindex 3

Designation	Buffer Organization
SDO ID	0x60C4
Subindex	3
Type	u8
Value range	0

Selection between ring or FIFO buffer.

Subindex 4

Designation	Buffer Position
SDO ID	0x60C4
Subindex	4
Type	u16
Value range	1

Current buffer position. (With the controller always 1; not requested or updated)

Subindex 5

Designation	Size of data record
SDO ID	0x60C4
Subindex	5
Type	u8
Value range	4

Has a meaning if, for example, a spline interpolation is used. Several values are necessary per interpolation section for this case. This object specifies the size of a record.

Subindex 6

Designation	Buffer Clear
SDO ID	0x60C4
Subindex	6
Type	u8
Value range	0

The buffer could be deleted with this object. Since the buffer of the controller has a length of 1 only, deleting does not make any sense.

7.9 Recommendations for PDO Mapping

Control word (RPDO1)

The PDO mapping of the control word (RPDO1) could be changed in such a way that, in addition, the “Modes of Operation” object (SDO 0x6060) is also received with this PDO of the controller. The Transmission Type can be left as 255 (asynchronous). However, it is also possible to readily configure this object as a synchronous PDO (Transmission Type 0 to 240). Once the IP mode is in operation, it is not necessary to continually update this object as that would cause unnecessary CAN bus load.

Status words (TPDO1)

The PDO mapping of the status word (TPDO1) could be changed in such a way that, in addition, the “Modes of Operation Display” object (SDO 0x6061) is also sent with this PDO of the controller. The Transmission Type can be left as 255 (asynchronous). However, it is also possible to readily configure this object as a synchronous PDO (Transmission Type 0). It is then sent synchronously to the Sync object if the status word or “Modes of Operation Display” changes which does not occur during normal operation of the Interpolated Position Mode. If logging of the position is desired, the “Position actual Value” object (SDO 0x6064) can also be attached to this TPDO as well. Then it is imperative, however, for this PDO to be configured as synchronous and cyclical (Transmission Type 1 to 240). It would then also be conceivable to set the Transmission Type to values greater than 1 so the current position is not sent in every Sync object. The user must decide what is sensible for him here.

Interpolation Data Record (SDO 0x60C1)

The main object for the IP mode is the “Interpolation Data Record” (SDO 0x60C1). This object has no standard mapping. It would be sensible to use mapping to RPDO2, Transmission Type 0 (cyclically synchronous). The higher-order controller must send the next demanded position in this PDO after each Sync object. The demanded position is then adopted in the next Sync object (see Figure 3). When using several controllers, it makes sense to pack the demanded position for two controllers in one PDO and to mask the target position of the other controller by an s32 dummy object

(SDO 0x0004) in the respective controller. Thus the necessary CAN bandwidth can be reduced and the saved bandwidth can be used for a higher interpolation frequency.

7.10 Information about commissioning

If other modes have been used before the IP mode is used, it is very probable that the current position of the rotor is not zero. If the IP mode is then selected, the user must ensure that the demanded position in the “Interpolation Data Record” object (SDO 0x60C1) corresponds to the actual position before interpolation is started with bit 4 of the control word.

If this is not done, a movement to the zero position can occur within an interpolation period (the time interval between two Sync objects) when switching on the interpolation. In Open Loop mode this is expressed by jerking of the shaft or short whistling (steps are lost here). In Closed Loop Mode the controller uses maximum current until the target position is reached.

To prevent this, the “Position Demand value” object (SDO 0x6062) must be adopted for Open Loop mode and the “Position actual value” object (SDO 0x6064) must be adopted for Closed Loop mode as the starting position for the Interpolation Data Record (SDO 0x60C1).

8 Torque Mode

8.1 Introduction

The Torque Mode serves to set the torque according to a specified target value.

As the resultant rotational speed depends on the properties of the motor and the load torque, it is possible to limit the maximum rotational speed by means of the SDO 0x607F.

The Torque mode is selected with the value '4' in SDO 0x6060 (Modes of Operation).

For the Torque Mode, the Closed Loop Mode must be active.

8.2 Control word (SDO 0x6040)

Designation	Control word
SDO ID	0x6040
Type	u16
Value range	Bits

This object is available in the default setting as a PDO with the COB ID 0x200 + node ID. An RTR is not configured.

The motor is switched on and travel commands can be carried out with this object.

Bits 0, 1 and 3 are used for starting up the power drive. These are described in the General section.

Bit 2 is used to trigger an emergency stop. If it is set to “0”, the motor carries out a quick stop with the ramp set in SDO 0x6085. The motor then goes into “Switch On Disabled” (see general description of the control word)

Bit 8 ("stop bit") serves to start and stop the motor. If the value is set to '1', the motor will be stopped. During a transition from '1' to '0', the motor begins to turn and the torque is set according to SDO 0x6071.

Bits 4, 6, 7, 9-15 are not used in the Torque Mode and are set to 0.

8.3 Status word (SDO 0x6041)

Designation	Status word
SDO ID	0x6041
Type	u16
Value range	Bits

Bit 10 serves as the torque status indicator.

Bit 10 Status word	Bit 8 Control word	Meaning
0	0	Specified torque not attained
0	1	Motor brakes
1	0	Specified torque attained
1	1	Motor idle

Bits 0-9, 11-15, refer to the general description.

8.4 Target torque (SDO 0x6071)

Designation	Target torque
SDO ID	0x6071
Type	s16
Value range	-1000 to 1000

This object includes the target value for the torque to be set. The torque is directly proportionate to the current, which is why the value specified in thousands of the maximum settable current.

Ex.:

- Controller SMCI47-S with nominal current 7.5 A
- SDO 0x2004 Sub 1 to 20 (20% of 7.5 A)
- SDO 0x6071 to 500

→ 500 / 1000 * 0.2 * 7.5A = 0.75 A

8.5 Max Profile Velocity (SDO 0x607F)

Designation	Max Profile Velocity
SDO ID	0x607F
Type	u32
Value range	u32

This object includes the maximum permissible rotational speed as an amount for both rotational directions, which can be set in the Torque mode.

The value is specified in RPM.

9 SDOs that are not mode-specific

9.1 Digital inputs (SDO 0x60FD)

Object description

Designation	Digital Inputs
SDO ID	0x60FD
Type	u32
Value range	32-bit bitmask

The digital inputs of the motor can be read with this SDO.

This SDO is read only.

Bits

Bits 0 to 1: Unassigned (“0”)

Bit 2: Input 6 (reference switch)

Bits 3 to 15: Unassigned (“0”)

Bits 16 to 21: Input 1 to Input 6

Bit 22 to 31: Unassigned (“0”).

9.2 Digital outputs (SDO 0x60FE Subindex 1)

Object description

Designation	Digital outputs
SDO ID	0x60FE
Subindex	1
Type	u32
Value range	32-bit bitmask

The digital outputs of the motor can be written with this SDO.

The writing of the outputs via this SDO is only possible if they are also released with the output mask (SDO 0x60FE Subindex 2) for the user. Otherwise, the outputs specify the motor status (see also manual).

Bits

Bits 0 to 15 Unassigned (“0”)

Bits 16 to 18: Output 1 to Output 3

Bits 19 to 31: Unassigned (“0”).

9.3 Digital outputs bitmask (SDO 0x60FE Subindex 2)

Object description

Designation	Bitmask
SDO ID	0x60FE
Subindex	2
Type	u32
Value range	32-bit bitmask

The digital outputs of the motor can be reserved for the user with this SDO.

A “0” means that the respective output is reserved for the user and can thus be changed with SDO 0x60FE Subindex 1. A “1” in the mask means that the status of the firmware has an effect on the outputs. These cannot then be affected by SDO 0x60FE Subindex 1.

Bits

Bits 0 to 15 Unassigned (“0”)

Bits 16 to 18: Mask for Output 1 to Output 3

Bits 19 to 31: Unassigned (“0”)

If unassigned bits have a “1” written to them, the entire mask is discarded.

9.4 Polarity (0x607E)

Object description

Designation	Polarity
SDO ID	0x607E
Type	u8
Value range	8-bit bitmask, bits 6 and 7 can be used, bits 0 to 5 are reserved (= 0)

Bits

Bit 6: Reverses the direction of rotation in Velocity Mode.

Bit 7: Reverses the direction of rotation in Profile Position Mode.

(“1” means reversal is activated, “0” means direction of rotation as described in the respective mode).

9.5 Producer Heartbeat Time (0x1017)

Object description

Designation	Producer Heartbeat Time
SDO ID	0x1017
Type	u16
Value range	u16

Sets the Heartbeat Time in ms (milliseconds).

Default setting is 0 (deactivated).

If the value is set to 0, the Heartbeat is switched off.

With values that are not equal to 0, the CANopen network status is sent to the COB ID 0x700 + node ID in the set interval.

Attention:
By setting shorter intervals, the CAN bus can be so heavily loaded that other communication is hardly still possible.

Requesting the network status

If the Heartbeat is switched off, the network status can be requested by sending a Remote Transmission Request to the COB ID 0x700 + node ID.

The network status is a u8 and can have the following values:

- 0x7F: Pre-Operational (status after application of the operating voltage)
- 0x04: Node Disabled
- 0x05: Node Operational.

9.6 DEVICE TYPE (0x1000)

Object description

Designation	Device Type
SDO ID	0x1000
Type	u16, ro
Value range	u16

Describes the control type. Here: 0x00008 (stepper motor)

9.7 ERROR REGISTER (0x1001)

Object description

Designation	Error Register
SDO ID	0x1001
Type	u8
Value range	u8, ro

Error register: In the event of error the corresponding error bit is set.

Bit 0: generic error

Bit 1: current (0)

Bit 2: voltage (0)

Bit 3: temperature (0)

Bit 4: communication error (overrun, error state)

Bit 5: device profile specific (0)

Bit 6: reserved (0)

Bit 7: manufacturer specific (0)

9.8 COB_ID_EMERGENCY_MESSAGE (0x1014)

Object description

Designation	Emergency Message
SDO ID	0x1014
Type	u32, ro
Value range	u32

Default Value 80h + node ID

Bits

Bit 31 (MSB): 0 = EMCY exists, 1 = EMCY does not exist

Bit 30: 0 = reserved (always 0)

Bit 29: 0 = 11-bit ID (CAN 2.0A), 1 = 29-bit ID (CAN 2.0B)

Bit 28-11: 0 = If bit 29=0; bits 28-11 of the 29-bit COB-ID if bit 29=1:

Bit 10-0 (LSB): Bits 10-0 of the COB ID

9.9 IDENTITY_OBJECT_VENDOR_ID (0x1018)

Object description

Designation	Identity Object
SDO ID	0x1018
Type	Record

Subindex 0h: Number of entries

Type	u8
Value range	1 ... 4, ro

Subindex 1h: Vendor ID

Type	u32, ro

Subindex 2h: Product code

Type	u32, ro

Subindex 3h: Revision number

Type	u32, ro

Subindex 4h: Serial number

Type	u32, ro

9.10 RPDO1_COMMUNICATION_PARAMETER (0x1400)

Object description

Designation	Receive PDO1 Communication Parameter
SDO ID	0x1400
Type	Record
Number of entries	2

Contains the communication parameters for RPDO 1.

Subindex 01h: COB ID

Type	u32, rw
Default value	0x200+Node ID The default value does not take effect by changing the node ID, but only by resetting the EEPROM (see section 1.1.3).

Bits

Bit 31: 1 = PDO deactivated

Bit 30: 1 = RTR not allowed

Bit 29: 1 = 29-bit ID, 0 = 11-bit ID

Bits 28-11: Bits 28-11 of the 29-bit ID, otherwise 0

Bits 10-0: Bits 10-0 of the COB ID

Subindex 02h: Transmission Type

Type	u8, rw
Default value	255

Values

0 to 240: Update of the object after receiving the next Sync object.

255: Immediate update of the object.

9.11 RPDO2_COMMUNICATION_PARAMETER (0x1401)

Object description

Designation	Receive PDO2 Communication Parameter
SDO ID	0x1401
Type	Record
Number of entries	2

Contains the communication parameters for RPDO 2.

Subindex 01h: COB ID

Type	u32, rw
Default value	0x300 + Node ID The default value does not take effect by changing the node ID, but only by resetting the EEPROM (see section 1.1.3).

Bits

Bit 31: 1 = PDO deactivated

Bit 30: 1 = RTR not allowed

Bit 29: 1 = 29-bit ID, 0 = 11-bit ID

Bits 28-11: Bits 28-11 of the 29-bit ID, otherwise 0

Bits 10-0: Bits 10-0 of the COB ID

Subindex 02h: Transmission Type

Type	u8, rw
Default value	255

Values

0 to 240: Update of the object after receiving the next Sync object.

255: Immediate update of the object.

9.12 RPDO3_COMMUNICATION_PARAMETER (0x1402)

Object description

Designation	Receive PDO3 Communication Parameter
SDO ID	0x1403
Type	Record
Number of entries	2

Contains the communication parameters for RPDO 3.

Subindex 01h: COB ID

Type	u32, rw
Default value	0x400 + Node ID The default value does not take effect by changing the node ID, but only by resetting the EEPROM (see section 1.1.3).

Bits

Bit 31: 1 = PDO deactivated

Bit 30: 1 = RTR not allowed

Bit 29: 1 = 29-bit ID, 0 = 11-bit ID

Bits 28-11: Bits 28-11 of the 29-bit ID, otherwise 0

Bits 10-0: Bits 10-0 of the COB ID

Subindex 02h: Transmission Type

Type	u8, rw
Default value	255

Values

0 to 240: Update of the object after receiving the next Sync object.

255: Immediate update of the object.

9.13 RPDO4_COMMUNICATION_PARAMETER (0x1403)

Object description

Designation	Receive PDO4 Communication Parameter
SDO ID	0x1403
Type	Record
Number of entries	2

Contains the communication parameters for RPDO 4.

Subindex 01h: COB ID

Type	u32, rw
Default value	0x500 + Node ID The default value does not take effect by changing the node ID, but only by resetting the EEPROM (see section 1.1.3).

Bits

Bit 31: 1 = PDO deactivated

Bit 30: 1 = RTR not allowed

Bit 29: 1 = 29-bit ID, 0 = 11-bit ID

Bits 28-11: Bits 28-11 of the 29-bit ID, otherwise 0

Bits 10-0: Bits 10-0 of the COB ID

Subindex 02h: Transmission Type

Type	u8, rw
Default value	255

Values

0 to 240: Update of the object after receiving the next Sync object.

255: Immediate update of the object.

9.14 RPDO1_MAPPING (0x1600)

Object description

Designation	Receive PDO Mapping Parameter
SDO ID	0x1600
Type	Record
Number of mapped objects	0-8

Describes which object is mapped to the COB ID defined in the object 0x1400.

Subindex 1-8: Mapped objects

Designation	Receive PDO1 Mapping Parameter
Type	u32, rw

Describes one mapped object at a time.

Bits

Bits 31-16: index

Bits 15-8: subindex

Bits 7-0: object length in bits

9.15 RPDO2_MAPPING (0x1601)

Object description

Designation	Receive PDO2 Mapping Parameter
SDO ID	0x1601
Type	Record
Number of mapped objects	0-8

Describes which object is mapped to the COB ID defined in the object 0x1401.

Subindex 1-8: Mapped objects

Designation	Receive PDO Mapping Parameter
Type	u32, rw

Describes one mapped object at a time.

Bits

Bits 31-16: index

Bits 15-8: subindex

Bits 7-0: object length in bits

9.16 RPDO3_MAPPING (0x1602)

Object description

Designation	Receive PDO3 Mapping Parameter
SDO ID	0x1602
Type	Record
Number of mapped objects	0-8

Describes which object is mapped to the COB ID defined in the object 0x1402.

Subindex 1-8: Mapped objects

Designation	Receive PDO Mapping Parameter
Type	u32, rw

Describes one mapped object at a time.

Bits

Bits 31-16: index

Bits 15-8: subindex

Bits 7-0: object length in bits

9.17 RPDO4_MAPPING (0x1603)

Object description

Designation	Receive PDO4 Mapping Parameter
SDO ID	0x1603
Type	Record
Number of mapped objects	0-8

Describes which object is mapped to the COB ID defined in the object 0x1403.

Subindex 1-8: Mapped objects

Designation	Receive PDO Mapping Parameter
Type	u32, rw

Describes one mapped object at a time.

Bits

Bits 31-16: index

Bits 15-8: subindex

Bits 7-0: object length in bits

9.18 TPDO1_COMMUNICATION_PARAMETER (0x1800)

Object description

Designation	Transmit PDO1 Communication Parameter
SDO ID	0x1800
Type	Record
Number of entries	5

Contains the communication parameters for the TPDO 1.

Subindex 01h: COB ID

Type	u32, rw
Default value	0x180 + Node ID The default value does not take effect by changing the node ID, but only by resetting the EEPROM (see section 1.1.3).

Bits

Bit 31: 1 = PDO deactivated

Bit 30: 1 = RTR not allowed

Bit 29: 1 = 29-bit ID, 0 = 11-bit ID

Bits 28-11: Bits 28-11 of the 29-bit ID, otherwise 0

Bits 10-0: Bits 10-0 of the COB ID

Subindex 02h: Transmission Type

Type	u8, rw
Default value	255

Values

0: Sending of the object after the next Sync object if it has changed its value.

1 to 240: Sending of the object after the set number of Sync objects (1 to 240), independently of whether the object has changed.

255: Sending the object after a change, independent of the Sync object.

Subindex 03h: inhibit time

Type	u16, rw
Default value	1000

When Transmission Type 255 (asynchronous transmission on a change) is used, this value indicates the minimum time between the transmission of two consecutive objects in 100µs steps. For example, this can prevent the current position which changes continuously during travel from blocking the CAN bus.

Subindex 04h: reserved

Type	u8, rw
Default value	0

No meaning.

Subindex 05h: event timer

Type	u16, rw
Default value	0

For Transmission Type 255 (asynchronous transmission upon a change), this value indicates the maximum time between two transmitted objects of the same type. This setting can be used to cyclically send objects that rarely change. A value of “0” in this setting deactivates this behavior (default).

9.19 TPDO2_COMMUNICATION_PARAMETER (0x1801)

Object description

Designation	Transmit PDO2 Communication Parameter
SDO ID	0x1801
Type	Record
Number of entries	5

Contains the communication parameters for the TPDO 2.

Subindex 01h: COB ID

Type	u32, rw
Default value	0x280 + Node ID The default value does not take effect by changing the node ID, but only by resetting the EEPROM (see section 1.1.3).

Bits

Bit 31: 1 = PDO deactivated

Bit 30: 1 = RTR not allowed

Bit 29: 1 = 29-bit ID, 0 = 11-bit ID

Bits 28-11: Bits 28-11 of the 29-bit ID, otherwise 0

Bits 10-0: Bits 10-0 of the COB ID

Subindex 02h: Transmission Type

Type	u8, rw
Default value	255

Values

0: Sending of the object after the next Sync object if it has changed its value.

1 to 240: Sending of the object after the set number of Sync objects (1 to 240), independently of whether the object has changed.

255: Sending the object after a change, independent of the Sync object.

Subindex 03h: inhibit time

Type	u16, rw
Default value	1000

When Transmission Type 255 (asynchronous transmission on a change) is used, this value indicates the minimum time between the transmission of two consecutive objects in 100µs steps. For example, this can prevent the current position which changes continuously during travel from blocking the CAN bus.

Subindex 04h: reserved

Type	u8, rw
Default value	0

No meaning

Subindex 05h: event timer

Type	u16, rw
Default value	0

For Transmission Type 255 (asynchronous transmission upon a change), this value indicates the maximum time between two transmitted objects of the same type. This setting can be used to cyclically send objects that rarely change. A value of “0” in this setting deactivates this behavior (default).

9.20 TPDO3_COMMUNICATION_PARAMETER (0x1802)

Object description

Designation	Transmit PDO3 Communication Parameter
SDO ID	0x1802
Type	Record
Number of entries	5

Contains the communication parameters for the TPDO 3.

Subindex 01h: COB ID

Type	u32, rw
Default value	0x380 + Node ID The default value does not take effect by changing the node ID, but only by resetting the EEPROM (see section 1.1.3).

Bits

Bit 31: 1 = PDO deactivated

Bit 30: 1 = RTR not allowed

Bit 29: 1 = 29-bit ID, 0 = 11-bit ID

Bits 28-11: Bits 28-11 of the 29-bit ID, otherwise 0

Bits 10-0: Bits 10-0 of the COB ID

Subindex 02h: Transmission Type

Type	u8, rw
Default value	255

Values

0: Sending of the object after the next Sync object if it has changed its value.

1 to 240: Sending of the object after the set number of Sync objects (1 to 240), independently of whether the object has changed.

255: Sending the object after a change, independent of the Sync object.

Subindex 03h: inhibit time

Type	u16, rw
Default value	1000

When Transmission Type 255 (asynchronous transmission on a change) is used, this value indicates the minimum time between the transmission of two consecutive objects in 100µs steps. For example, this can prevent the current position which changes continuously during travel from blocking the CAN bus.

Subindex 04h: reserved

Type	u8, rw
Default value	0

No meaning.

Subindex 05h: event timer

Type	u16, rw
Default value	0

For Transmission Type 255 (asynchronous transmission upon a change), this value indicates the maximum time between two transmitted objects of the same type. This setting can be used to cyclically send objects that rarely change. A value of “0” in this setting deactivates this behavior (default).

9.21 TPDO4_COMMUNICATION_PARAMETER (0x1803)

Object description

Designation	Transmit PDO4 Communication Parameter
SDO ID	0x1803
Type	Record
Number of entries	5

Contains the communication parameters for the TPDO 4.

Subindex 01h: COB ID

Type	u32, rw
Default value	0x480 + Node ID The default value does not take effect on changing the node ID, but only by resetting the EEPROM (see section 1.1.3).

Bits

Bit 31: 1 = PDO deactivated

Bit 30: 1 = RTR not allowed

Bit 29: 1 = 29-bit ID, 0 = 11-bit ID

Bits 28-11: Bits 28-11 of the 29-bit ID, otherwise 0

Bits 10-0: Bits 10-0 of the COB ID

Subindex 02h: Transmission Type

Type	u8, rw
Default value	255

Values

0: Sending of the object after the next Sync object if it has changed its value.

1 to 240: Sending of the object after the set number of Sync objects (1 to 240), independently of whether the object has changed.

255: Sending the object after a change, independent of the Sync object.

Subindex 03h: inhibit time

Type	u16, rw
Default value	1000

When Transmission Type 255 (asynchronous transmission on a change) is used, this value indicates the minimum time between the transmission of two consecutive objects in 100µs steps. For example, this can prevent the current position which changes continuously during travel from blocking the CAN bus.

Subindex 04h: reserved

Type	u8, rw
Default value	0

No meaning

Subindex 05h: event timer

Type	u16, rw
Default value	0

For Transmission Type 255 (asynchronous transmission upon a change), this value indicates the maximum time between two transmitted objects of the same type. This setting can be used to cyclically send objects that rarely change. A value of “0” in this setting deactivates this behavior (default).

9.22 TPDO1_MAPPING (0x1A00)

Object description

Designation	Transmit PDO1 Mapping Parameter
SDO ID	0x1A00
Type	Record
Number of mapped objects	0-8

Describes which object is mapped to the COB ID defined in the object 0x1800.

Subindex 1-8: Mapped objects

Designation	Transmit PDO Mapping Parameter
Type	u32, rw

Describes one mapped object at a time.

Bits

31-16: index

15-8: subindex

7-0: object length in bits

9.23 TPDO2_MAPPING (0x1A01)

Object description

Designation	Transmit PDO2 Mapping Parameter
SDO ID	0x1A01
Type	Record
Number of mapped objects	0-8

Describes which object is mapped to the COB ID defined in the object 0x1801.

Subindex 1-8: Mapped objects

Designation	Transmit PDO Mapping Parameter
Type	u32, rw

Describes one mapped object at a time.

Bits

31-16: index

15-8: subindex

7-0: object length in bits

9.24 TPDO3_MAPPING (0x1A02)

Object description

Designation	Transmit PDO2 Mapping Parameter
SDO ID	0x1A02
Type	Record
Number of mapped objects	0-8

Describes which object is mapped to the COB ID defined in the object 0x1802.

Subindex 1-8: Mapped objects

Designation	Transmit PDO Mapping Parameter
Type	u32, rw

Describes one mapped object at a time.

Bits

31-16: index

15-8: subindex

7-0: object length in bits

9.25 TPDO4_MAPPING (0x1A03)

Object description

Designation	Transmit PDO4 Mapping Parameter
SDO ID	0x1A03
Type	Record
Number of mapped objects	0-8

Describes which object is mapped to the COB ID defined in the object 0x1803.

Subindex 1-8: Mapped objects

Designation	Transmit PDO Mapping Parameter
Type	u32, rw

Describes one mapped object at a time.

Bits

31-16: index

15-8: subindex

7-0: object length in bits

9.26 ERROR_CODE (0x603F)

Object description

Designation	Error Code
SDO ID	0x603F
Type	u16, ro
Value range	u16

Includes the error code of the last error that occurred.

9.27 VL_DIMENSION_FACTOR (0x604C)

Object description

Designation	vl dimension factor
SDO ID	0x604C
Type	Record
Number of entries	2

Conversion factor for the user-specific speed (e.g. 1/s, increments/s, m/s,...) in revolutions per minute.

Speed [rpm] = speed [user-specific unit]/conversion factor [rpm/user-specific unit].

Subindex 01h: vl dimension factor numerator

Type	u32, rw
Default value	1

Subindex 02h: vl dimension factor denominator

Type	u32, rw
Default value	1

9.28 MODES_OF_OPERATION (0x6060)

Object description

Designation	Modes of operation
SDO ID	0x6060
Type	s8, rw
Value range / Default	-128 - 10 / 0

Contains the current operating mode.

Values

State	Description
-128 to -1	Manufacturer-specific operation modes
-2	Short closed-loop test run (alignment)
-1	Closed-loop test run
0	No mode change/no mode assigned
+1	Profile Position Mode
+2	Velocity Mode
+3	Profile Velocity Mode
+4	Torque Profile Mode
+5	Reserved
+6	Homing Mode
+7	Interpolated Position mode
+11 to +127	Reserved

9.29 MODES_OF_OPERATION_DISPLAY (0x6061)

Object description

Designation	Modes of operation display
SDO ID	6061
Type	s8, ro
Value range	s8

Contains the current operating mode. -> see MODES_OF_OPERATION (SDO 0x6060)

9.30 POSITION_ACTUAL_INTERNAL_VALUE (0x6063)

Designation	Position actual internal value
SDO ID	0x6063
Type	s32, ro
Value range	s32

Contains the current encoder position.

9.31 POSITION_ACTUAL_VALUE (0x6064)

Object description

Designation	Position actual value
SDO ID	0x6064
Type	s32, ro
Value range	s32

Contains the current actual position (encoder position converted acc. to Feed Constant and Gear Ratio).

9.32 FOLLOWING_ERROR_WINDOW (0x6065)

Object description

Designation	Following error window
SDO ID	0x6065
Type	U32, rw
Value range	u32

Specifies the maximum following error symmetrically to the demanded position. If the actual position deviates too greatly from the demanded position, a following error is issued.

9.33 FOLLOWING_ERROR_TIME_OUT (0x6066)

Object description

Designation	Following error timeout
SDO ID	0x6066
Type	u16, rw
Value range	u16

Time in milliseconds until too large a following error leads to an error message.

9.34 POSITION_WINDOW (0x6067)

Object description

Designation	Position window
SDO ID	0x6067
Type	u32, rw
Value range	u32

Specifies a symmetrical range relative to the target position within which the target is considered to be reached.

9.35 POSITION_WINDOW_TIME (0x6068)

Object description

Designation	Position window time
SDO ID	0x6068
Type	u16, rw
Value range	u16

For this time period, the actual position must be within the position window so that the target position is considered to be reached.

9.36 POS_RANGE_LIMIT (0x607B)

Object description

Designation	Position range limit
SDO ID	0x607B
Type	Record
Number of entries	2

Contains the minimum and maximum position. If this range is exceeded or undercut, an overflow occurs. To prevent this overflow, see also SW_POS_LIMIT (SDO 0x607D).

Subindex 01h: Min position range limit

Type	s32, rw
Default value	-100000000

Subindex 02h: Max position range limit

Type	s32, rw
Default value	100000000

9.37 SW_POS_LIMIT (0x607D)

Designation	Software position limit
SDO ID	0x607D
Type	Record
Number of entries	2

The target position must lie within the limits set here. Before the check, the home offset (SDO 0x607C) is deducted each time:

corrected min position limit = min position limit - home offset

corrected max position limit = max position limit - home offset.

Subindex 01h: Min position limit

Type	s32, rw
Default value	-100000000

Subindex 02h: Max position limit

Type	s32, rw
Default value	100000000

9.38 POSITION_ENCODER_RESOLUTION (0x608F)

Object description

Designation	Position encoder resolution
SDO ID	0x608F
Type	Record
Number of entries	2

Encoder increments per revolution:

position encoder resolution = encoder increments / motor revolutions.

Subindex 01h: encoder increments

Type	u32, rw
Default value	2000

Subindex 02h: motor revolutions

Type	u32, rw
Default value	1

9.39 GEAR_RATIO (0x6091)

Object description

Designation	Gear ratio
SDO ID	0x6091
Type	Record
Number of entries	2

Number of motor revolutions per revolution of the driving axis:

gear ratio = motor shaft revolutions / driving shaft revolutions.

Subindex 01 h: Motor revolutions

Type	u32, rw
Default value	1

Subindex 02h: Shaft revolutions

Type	u32, rw
Default value	1

9.40 FEED_CONSTANT (0x6092)

Object description

Designation	Feed constant
SDO ID	0x6092
Type	Record
Number of entries	2

Specifies the feed per revolution for a linear drive:

Feed constant = feed / revolutions.

Subindex 01h: Feed

Type	u32, rw
Default value	2000

Subindex 02h: Shaft revolutions

Type	u32, rw
Default value	1

9.41 MAX_ACCEL (0x60C5)

Object description

Designation	Max acceleration
SDO ID	0x60C5
Type	u32, rw
Value range	u32

Contains the maximum admissible acceleration ramp. → see also MAX_DECEL (SDO 0x60C6).

9.42 MAX_DECEL (0x60C6)

Object description

Designation	Max deceleration
SDO ID	0x60C6
Type	u32, rw
Value range	u32

Contains the maximum braking ramp. → see also MAX_ACCEL (0x60C5).

9.43 SUPPORTED_DRIVE_MODES (0x6502)

Object description

Designation	Supported drive modes
SDO ID	0x6502
Type	record, ro
Value range	u32

The object describes the supported drive modes. Only PP, HM, VL, IP and TQ are currently supported in CANopen operation.

Bits

Bit	Meaning	Supported
31-16	Not used (manufacturer-specific)	0
15-20	Reserved	0
9	CST	0
8	CSV	0
7	CSP	0
6	IP (Interpolated Position mode)	1 (as of firmware version 06042009)
5	HM (Homing mode)	1
4	R	0
3	TQ (Torque Mode)	1
2	PV	0
1	VL (Velocity mode)	1
0	PP (Profile Position mode)	1

9.44 Boolean Dummy Object (0x0001)

Object description

Designation	Boolean
SDO ID	0x0001
Type	Bit, rw
Value range	Bit

Intended as a dummy object for PDO mapping.

Cannot be mapped as PDO.

9.45 Integer8 Dummy Object (0x0002)

Object description

Designation	Signed Integer 8
SDO ID	0x0002
Type	Integer 8, rw
Value range	-128 to +128

Can be used as a dummy object for RxPDO.

9.46 Integer16 Dummy Object (0x0003)

Object description

Designation	Signed Integer 16
SDO ID	0x0003
Type	Integer 16, rw
Value range	-32768 to +32767

Can be used as a dummy object for RxPDO.

9.47 Integer32 Dummy Object (0x0004)

Object description

Designation	Signed Integer 32
SDO ID	0x0004
Type	Integer 32, rw
Value range	-2147483648 to +2147483647

Can be used as a dummy object for RxPDO.

9.48 Unsigned8 Dummy Object (0x0005)

Object description

Designation	Signed Integer 8
SDO ID	0x0005
Type	Integer 8, rw
Value range	0 to 255

Can be used as a dummy object for RxPDO.

9.49 Unsigned16 Dummy Object (0x0006)

Object description

Designation	Signed Integer 16
SDO ID	0x0006
Type	Integer 16, rw
Value range	0 to 65535

Can be used as a dummy object for RxPDO.

9.50 Unsigned32 Dummy Object (0x0007)

Object description

Designation	Signed Integer 32
SDO ID	0x0007
Type	Integer 32, rw
Value range	0 to 4294967295

Can be used as a dummy object for RxPDO.

9.51 ERROR CODE LOG

Object description

Designation	Error Code Log
SDO ID	1003
Subindex	1 – 5
Type	u32, ro
Value range	u32

This SDO contains a history the last errors that occurred. Subindex 0 thereby contains the number of the logged errors and subindex 1 the last errors that occurred.

Each entry thereby contains in the lower-order word the error code that occurred and, in the higher-order word, a firmware-internal error number.

10 Analog input

10.1 Overview

Generic

The SMCI47-S controller has an analog input that supports the trigger conditions as they are described in CiA CANopen 401.

Resolution

The input has a resolution of 10 bits at a conversion rate of 1kHz. The full resolution refers to an input voltage range of -10V to +10V.

Values

The output values are of the s16 type where only positive values are used. The corresponding SDO (0x6401,1) is continuously updated. The trigger conditions for correspondingly mapped TPDOs can be set in the objects 0x6423 to 0x6428. Scaling or offset calculation is not supported.

Designation	Read analog input 16-bit
SDO ID	0x6401 Subindex 1
Type	s16, ro
Value range	0x0000 to 0x03FF
Contains the last converted value of the analog input. This object can be mapped as a PDO. Special trigger conditions are supported for the PDO	

10.2 Analog Input (0x6401)

Object description

This object contains the converted value of the analog input and can be mapped as TxPDO.

Designation	Analog Input
SDO ID	0x6401
Type	Array
Number of entries	1

Designation	Number of analog inputs
Subindex	0
Type	u8, ro
Value range	1

Designation	Analog Input 1 conversion value
Subindex	1
Type	u16, ro
Value range	0x0 to 0x3FF (10-bit)

10.3 Analog Global Interrupt Enable (0x6423)

Object description

Only when this object has been set to 1 can a change of the analog value trigger the sending of the corresponding PDO.

Designation	Analog Global Interrupt Enable
SDO ID	0x6423
Type	U8, rw
Value range	0 or 1
Default	0

10.4 Analog Input Interrupt Upper Limit (0x6424)

Object description

If the analog value changes and is greater than the limit set in this object, a PDO is sent.

If the upper limit is less than the lower limit, the range between the upper and lower limit is active.

If the upper limit is greater than the lower limit, the ranges between 0 and lower limit and between upper limit and infinity are active.

Designation	Analog Input Interrupt Upper Limit
SDO ID	0x6424
Type	Array
Number of entries	1

Designation	Number of analog inputs
Subindex	0
Type	u8, ro
Value range	1

Designation	Analog 1 upper limit
Subindex	1
Type	s32, rw
Value range	0x0 to 0x3FF (10-bit)

10.5 Analog Input Interrupt Lower Limit (0x6425)

Object description

If the analog value changes and is less than the limit set in this object, a PDO is sent.

If the upper limit is less than the lower limit, the range between the upper and lower limit is active.

If the upper limit is greater than the lower limit, the ranges between 0 and lower limit and between upper limit and infinity are active.

Designation	Analog Input Interrupt Lower Limit
SDO ID	0x6425
Type	Array
Number of entries	1

Designation	Number of analog inputs
Subindex	0
Type	u8, ro
Value range	1

Designation	Analog 1 lower limit
Subindex	1
Type	s32, rw
Value range	0x0 to 0x3FF (10-bit)

10.6 Analog Input Interrupt Delta (0x6426)

Object description

If the analog value changes and the difference to the last sent value is greater than the value set in this object, a PDO is sent. In addition, the limits from the upper and lower limits are evaluated.

Designation	Analog Input Delta
SDO ID	0x6426
Type	Array
Number of entries	1

Designation	Number of analog inputs
Subindex	0
Type	u8, ro
Value range	1

Designation	Change limit Analog 1
Subindex	1
Type	u32, rw
Value range	0x0 to 0x3FF (10-bit)

10.7 Analog Input Interrupt negative Delta (0x6427)

Object description

If the analog value changes to a negative value and the difference to the last sent value is greater than the value set in this object, a PDO is sent. In addition, the limits from the upper and lower limits are evaluated. If this value is less than the analog input delta, it is not evaluated.

Designation	Analog Input negative Delta
SDO ID	0x6427
Type	Array
Number of entries	1

Designation	Number of analog inputs
Subindex	0
Type	u8, ro
Value range	1

Designation	Negative change limit Analog 1
Subindex	1
Type	u32, rw
Value range	0x0 to 0x3FF (10-bit)

10.8 Analog Input Interrupt positive Delta (0x6428)

Object description

If the analog value changes to a positive value and the difference to the last sent value is greater than the value set in this object, a PDO is sent. In addition, the limits from the upper and lower limits are evaluated. If this value is less than the analog input delta, it is not evaluated.

Designation	Analog Input negative Delta
SDO ID	0x6428
Type	Array
Number of entries	1

Designation	Number of analog inputs
Subindex	0
Type	u8, ro
Value range	1

Designation	Positive change limit Analog 1
Subindex	1
Type	u32, rw
Value range	0x0 to 0x3FF (10-bit)

11 Manufacturer-specific CAN Objects

11.1 STEP_MODE (0x2000)

Object description

Designation	Step mode
SDO ID	0x2000
Type	u8, rw
Value range/Default value	1,2,4,5,8,10,16,32,64,255

This object is obsolete. The step mode that can be set here results from the objects “Feed Constant” and “Gear Ratio”. A smooth movement is given through microstep interpolation which is always active.

11.2 ENABLE_CL (0x2001)

Object description

Designation	Enable closed loop
SDO ID	0x2001
Type	u8, rw
Value range	0, 1, 2, 3

If the value is set to '1', '2' or '3', the firmware is instructed to activate the control loop. This is only activated, however, if specific prerequisites are met:

Value	Description
0	The control loop is immediately deactivated.
1	Closed loop is activated as soon as the index has been recognized and the controller is back in "Ready" status ("Auto-Enable after the travel").
2	Closed Loop is activated as soon as the index has been recognized ("Auto-Enable during the travel").
3	Closed loop is activated as soon as a short CL test run has been carried out (Modes of Operation (SDO 0x6060) = -2). This mode is available as of firmware version 24-10-2011.

Prerequisites

Before using closed loop mode for the first time, the controller must be adapted to the motor/encoder combination. This is carried out using the rotor position test run (Modes of Operation (SDO 0x6060) = -1). Otherwise, there is a high probability that the controller will rotate with maximum current in the wrong direction.

11.3 CL_CONFIGURATION (0x2002)

Object description

Designation	Closed loop Configuration
SDO ID	0x2002
Type	Record
Number of entries	24

Contains the control parameters for the closed loop mode.

Subindex 01h: KP_V_Z

Type	u16, rw
Default value	2

Numerator of the proportional component of the speed controller.

Subindex 02h: KP_V_N

Type	u16, rw
Default value	0

Denominator of the proportional component of the speed controller as a power of 2.

Subindex 03h KI_V_Z

Type	u16, rw
Default value	1

Numerator of the integral component of the speed controller.

Subindex 04h: KI_V_N

Type	u16, rw
Default value	6

Denominator of the integral component of the speed controller as a power of 2.

Subindex 05h KD_V_Z

Type	u16, rw
Default value	0

Numerator of the differential component of the speed controller.

Subindex 06h: KD_V_N

Type	u16, rw
Default value	0

Denominator of the differential component of the speed controller as a power of 2.

Subindex 07h KP_S_Z

Type	u16, rw
Default value	100

Numerator of the proportional component of the position controller.

Subindex 08h: KP_S_N

Type	u16, rw
Default value	0

Denominator of the proportional component of the position controller as a power of 2.

Subindex 09h KI_S_Z

Type	u16, rw
Default value	2

Numerator of the integral component of the position controller.

Subindex 0Ah: KI_S_N

Type	u16, rw
Default value	0

Denominator of the integral component of the position controller as a power of 2.

Subindex 0Bh KD_S_Z

Type	u16, rw
Default value	300

Numerator of the differential component of the position controller.

Subindex 0Ch: KD_S_N

Type	u16, rw
Default value	0

Denominator of the differential component of the position controller as a power of 2.

Subindex 0Dh KP_CSV_Z

Type	u16, rw
Default value	50

Numerator of the proportional component of the speed cascade controller.

Subindex 0Eh: KP_CSV_N

Type	u16, rw
Default value	0

Denominator of the proportional component of the speed cascade controller.

Subindex 0Fh KI_CSV_Z

Type	u16, rw
Default value	2

Numerator of the integral component of the speed cascade controller.

Subindex 10h: KI_CSV_N

Type	u16, rw
Default value	0

Denominator of the integral component of the speed cascade controller.

Subindex 11h KD_CSV_Z

Type	u16, rw
Default value	0

Numerator of the differential component of the speed cascade controller.

Subindex 12h: KD_CSV_N

Type	u16, rw
Default value	0

Denominator of the differential component of the speed cascade controller.

Subindex 13h KP_CSS_Z

Type	u16, rw
Default value	2

Numerator of the proportional component of the position cascade controller.

Subindex 14h: KP_CSS_N

Type	u16, rw
Default value	0

Denominator of the proportional component of the position cascade controller.

Subindex 15h KI_CSS_Z

Type	u16, rw
Default value	1

Numerator of the integral component of the position cascade controller.

Subindex 16h: KI_CSS_N

Type	u16, rw
Default value	6

Denominator of the integral component of the position cascade controller.

Subindex 17h KD_CSS_Z

Type	u16, rw
Default value	0

Numerator of the differential component of the position cascade controller.

Subindex 18h: KD_CSS_N

Type	u16, rw
Default value	0

Numerator of the differential component of the position cascade controller.

11.4 A/D Converter (0x2003)

Object description

Designation	Actual value of the A/D converter
SDO ID	0x2003
Type	Record
Number of entries	4

Contains the actual values of the A/D converter (Raw Values).

Sub-Index 01h: A/D 1 – Temperature Channel

Type	u16, ro
Default value	-

Sub-Index 02h: A/D 2 – Power Channel

Type	u16, ro
Default value	-

Sub-Index 03h: A/D 3 – Analog Input Channel

Type	u16, ro
Default value	-

Sub-Index 04h: A/D 4 – Auxiliary Power Channel

Type	u16, ro
Default value	-

11.5 Current control (0x2004)

Object description

Designation	Current limiter
SDO ID	0x2004
Type	Record
Number of entries	14

Contains the set current values.

Subindex 01h: drive current

Type	u8, rw
Default value	20

Current that is used for normal travel. Data in %.

Subindex 02h: current reduction

Type	u8, rw
Default value	20

Current that is used for the reduction. Data in %.

Subindex 03h: current reduction time

Type	u16, rw
Default value	80

Time in milliseconds from a standstill of the motor in Open Loop mode until the current is reduced.

Subindex 04h: current for block reference run

Type	u8, rw
Default value	0

Current for the block reference run in percent.
If the value is 0, the phase current is used.

Subindex 05h: Swing out time

Type	u16, rw
Default value	80

Time in milliseconds between a standstill of the axis and position reset after a block reference run and between standstill of the axis and a ready message in PP mode.

Subindex 06h: Position tolerance

Type	u32, rw
Default value	0xFFFFFFFF

Tolerance after a travel in PP mode (Open Loop).

Subindex 07h: Motor Type

Type	u8, rw
Default value	0

Motor type:

- 0: Stepper motor with and without quadrature encoder
- 1: BLDC with Hall sensor and without quadrature encoder
- 2: BLDC with Hall sensor and with quadrature encoder

Subindex 08h: Load angle sampling spacing

Type	u16, rw
Default value	4096

Sampling point spacing of the load angle curve in SDO 0x200C.
A value of 8192 corresponds to 1000 rpm.

Subindex 09h: Cascade controller switch-on speed

Type	u32, rw
Default value	327680

The cascade controller is activated as of this speed.
A value of 8192 corresponds to 1000 rpm.

Subindex 0Ah: Cascade controller switch-off speed

Type	u32, rw
Default value	512

The cascade controller is deactivated as of this speed.
A value of 8192 corresponds to 1000 rpm.

Subindex 0Bh: Cascade controller status

Type	u8, ro
Default value	0

Specifies whether the cascade controller is active.

Subindex 0Ch: Hall sensor mode

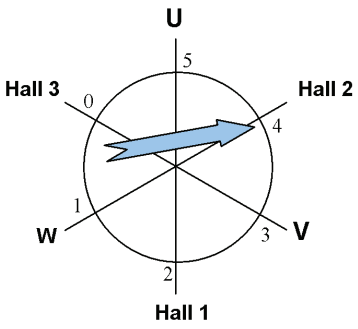
Type	u32, rw
Default value	2371605 (0x243015)

Used with BLDC motors to adjust the Hall sensor sections to the respective motor. Here the least significant 6 digits in hexadecimal notation represent one quadrant of the Hall sensors.

The value results from the motor type and the sequence of the Hall signals. The index is calculated as follows: (Hall1 * 1) + (Hall2 * 2) + (Hall3 * 4).

Motor assignment

Index	1	2	3	4	5	6
Type 1 (not DB47)	1	3	0	2	5	4
Type 2 (DB47)	4	0	3	5	2	1



Example: Motor DB57

Hall sequence according to the data sheet:

Hall 1	0	0	0	1	1	1
Hall 2	0	1	1	1	0	0
Hall 3	1	1	0	0	0	1
Index	4	6	2	3	1	5
Value (Type 1)	2	4	3	0	1	5

The index can be calculated from the Hall states. On the basis of the table for the motor assignment, the values 2, 4, 3, 0, 1, 5 then result. These are entered in the controller as Hex 0x243015.

Subindex 0Dh: Peak current

Type	u32, rw
Default value	0 (0x0)

Used with BLDC motors to adjust the maximum admissible peak current. If this value is less than the motor current (Subindex 1), it is ignored. If the value is greater, the motor current can be momentarily exceeded depending on the time constant.

Subindex 0Eh: Peak current time constant

Type	u32, rw
Default value	0 (0x0)

Used with BLDC motors to adjust the time constant for the peak current. Enables a momentary exceeding of the set motor current (Subindex 1).

11.6 CAN enable and baud rate (0x2005)

Object description

The CAN baud rate is adjusted with this object. For a more detailed description, see section 1.1.3.

Designation	CAN enable and baud rate
SDO ID	0x2005
Type	u8
Value range	130,131,132,133,134,135
Default value	135 (is not reset by an EEPROM Reset)

11.7 Motor Pole Pairs (0x2006)

Object description

Designation	Number of pole pairs of the motor
SDO ID	0x2006
Type	u16
Value range	50 and 100
Default value	50

Specifies the number of pole pairs of the motor. After changing this parameter, the controller must be provided with power again. For use in Closed Loop, the motor must be readjusted with the aid of the rotor position test run.

Pool pair count	**Step angle**
50	1.8°
100	0.9°

11.8 Brake Wait Time (0x2007)

Object description

Designation	Wait time for mech. brake
SDO ID	0x2007
Type	Record
Number of entries	3

The SMCI47-S has an output for connection of a spring-loaded, electromagnetically releasable brake. Delay times for this brake can be set with this SDO.

While a braking wait time is carried out, no commands for changing the mode are executed.

Subindex 01h: Milliseconds Power on to Brake off

Type	u16, rw
Default value	0
Specifies the wait time between switch-on of the power drive and release of the brake in milliseconds.	

Subindex 02h: Milliseconds Brake off to Operational

Type	u16, rw
Default value	0
Specifies the wait time between releasing the brake and reaching the “Operational” state in milliseconds. This setting prevents the motor from rotating due to a command while the brake has not yet fully released.	

Subindex 03h: Milliseconds Brake on to Power off

Type	u16, rw
Default value	0
Specifies the wait time between arresting the brake and switching off the power drive in milliseconds. This setting prevents the motor from moving due to a mechanical torque when the brake has not yet fully arrested.	

11.9 Milliseconds Input Debounce Time (0x2008)

Object description

Used to debounce the digital inputs of the controller. After a flank on an input, no further flanks are processed within the time set in this object in milliseconds. A flank is only detected again after expiration of the debounce time. A debounce time of an input that is running has no effect on the detection of flanks on the other inputs.

Designation	Debounce time
SDO ID	0x2008
Type	u8, rw
Value range	0 to 255
Default value	20

11.10 Node ID (0x2009)

Object description

This object is used to adjust the node ID if the node ID is set to 0 with the hex switches. If a node ID is set with the hex switches, this object can be described and saved in the EEPROM, but contains the value of the hex switches again after a restart.

Designation	Node ID
SDO ID	0x2009
Type	u8, rw
Value range	1 to 127
Default value	1 (is not reset by an EEPROM Reset)

11.11 CL is enabled (0x200A)

Object description

It is possible to determine whether the Closed Loop Mode is active with the aid of this object. If the object returns the value 1, Closed Loop Mode is active.

Designation	CL is enabled
SDO ID	0x200A
Type	u8, ro
Value range	0 and 1
Default value	0

11.12 CL POSCNT Offset (0x200B)

Object description

Specifies the distance between encoder index and rotor alignment. A value of 65536 corresponds to an electrical angle of 360° or 0°. This value is set by the CL test run and is different for each individual motor.

Designation	CL POSCNT Offset
SDO ID	0x200B
Type	u16, rw
Value range	0 to 65535
Default value	0

11.13 CL load angle curve (0x200C)

Object description

Specifies the speed-dependent load angle. A value of 65536 corresponds to an electrical angle of 360° or 0°. This value is set by the CL test run and is identical for motors of the same series. Each of the subindices of 1 to 7 contains a load angle for a speed range. The speed interval between the individual values can be set in SDO 2004 Subindex 8.

Designation	CL load angle curve
SDO ID	0x200C
Type	Array
Number of entries	10

Subindex 00h Number of entries

Type	u8, ro
Default value	10

Subindex 01h: Load angle at a standstill

Type	u16, rw
Default value	16384

Subindex 02h Load angle at speed = load angle sampling spacing * 1

Type	u16, rw
Default value	17000

Subindex 03h Load angle at speed = load angle sampling spacing * 2

Type	u16, rw
Default value	17500

Subindex 04h Load angle at speed = load angle sampling spacing * 3

Type	u16, rw
Default value	17750

Subindex 05h Load angle at speed = load angle sampling spacing * 4

Type	u16, rw
Default value	18000

Subindex 06h Load angle at speed = load angle sampling spacing * 5

Type	u16, rw
Default value	18000

Subindex 07h Load angle at speed = load angle sampling spacing * 6

Type	u16, rw
Default value	18000

Subindex 08h is not used

Type	u16, rw
Default value	18000

Subindex 09h is not used

Type	u16, rw
Default value	18000

Subindex 0Ah is not used

Type	u16, rw
Default value	18000

11.14 Encoder rotation direction change (0x200D)

Object description

If this object is set 1, the direction of the quadrature encoder is reversed.

Designation	Encoder Reverse Direction
SDO ID	0x200D
Type	u8, rw
Value range	0 or 1
Default value	0

11.15 DSPdrive current controller parameter (0x200E)

Object description

For controllers with DSPdrive (PD2-N, PD4-N, SMCI12, SMCI36) the current controller parameters can be changed.

Designation	DSPdrive current controller parameters
SDO ID	0x200E
Type	Array
Number of entries	6

Subindex 00h Number of entries

Type	u8, ro
Default value	6

Subindex 01h: KP low (not used)

Type	u16, rw
Default value	1

Subindex 02h KP high

Type	u16, rw
Default value	10

KP value at a standstill.

Subindex 03h KP scale

Type	u16, rw
Default value	58

KP value is increased in proportion to the speed.

Subindex 04h KI low (not used)

Type	u16, rw
Default value	1

Subindex 05h KI high

Type	u16, rw
Default value	10

KI value at a standstill

Subindex 06h KI scale

Type	u16, rw
Default value	200

KI value is increased in proportion to the speed.

11.16 Speed mode controller type (0x200F)

Object description

If this object is set to 1, the position controller is used in VL mode instead of the speed controller. This can be an advantage at very low speeds.

Designation	Controller type speed mode
SDO ID	0x200F
Type	u8, rw
Value range	0 or 1
Default value	0

11.17 External reference run IO (0x2010)

Object description

An external IO node can also be used as a reference switch instead of input 6 of the controller.

Designation	External reference run IO
SDO ID	0x2010
Type	Array
Number of entries	2

Subindex 00h Number of entries

Type	u8, ro
Default value	2

Subindex 01h: Bit number

Type	u8, rw
Default value	0

Selects the bit in Subindex 2 that is to be interpreted as a reference switch.

Subindex 02h Reference switch

Type	u8, rw
Default value	-

This object is used to be mapped as RxPDO. A bit from this object indicates the state of the limit switch to which the controller reacts in the corresponding reference run.

11.18 ENCODER_TYPE (0x2011)

Object description

Designation	Encoder type
SDO ID	0x2011
Type	u8, rw
Value range	0, 1, 2, 3 (see below)

Sets the type of encoder which is connected. Each type is represented by a unique value.

Values

Value	Encoder type
0	No encoder
1	Incremental encoder with index
2	Incremental encoder without index
3	Absolute encoder, single-turn

This command is available as of firmware version 24-10-2011.

11.19 COB ID SYNC message (0x1005)

Object description

Designation	COB ID SYNC message
SDO ID	0x1005
Type	u32, rw
Value range	0 to 0x7FF (see below)
Default value	0x80

The COB ID of the Sync object is set with this object.

Bits

The individual bits have the following meaning:

Bit 31: No meaning

Bit 30: Device creates a Sync message (always 0)

Bit 29 uses 29-bit COB ID (always 0)

Bit 28-11: Extended COB ID (always 0)

Bit 10-0: COB ID

11.20 Hardware Version (0x1009)

Object description

Designation	Hardware Version
SDO ID	0x1009
Type	String (ro)
Value range	-
Default value	-

This object contains the hardware version as a character string.

The length of the character string appears in Subindex 0 of this SDO. The individual characters are contained as of Subindex 1. The character string is not terminated by a zero string.

11.21 Software Version (0x100A)

Object description

Designation	Software Version
SDO ID	0x1009
Type	String (ro)
Value range	-
Default value	-

This object contains the software version as character string.

The length of the character string appears in Subindex 0 of this SDO. The individual characters are contained as of Subindex 1. The character string is not terminated by a zero string.

11.22 Guard Time (0x100C)

Object description

Designation	Guard Time
SDO ID	0x100C
Type	u16, rw
Value range	0 to 65535
Default value	0

The Guard Time is set in ms. It specifies the time intervals at which a request of the NMT status (RTR of COB ID 0x700 + node ID) is expected.

The Node Guard function can be deactivated with the value 0.

11.23 Life Time Factor (0x100D)

Object description

Designation	Life Time Factor
SDO ID	0x100D
Type	u8, rw
Value range	0 to 255
Default value	0

The Life Time Factor specifies after how many missing NMT status requests (RTR of 0x700+ node ID) an error is generated.

The time resulting from the objects “Guard Time” and “Life Time Factor” must not exceed one minute.

11.24 Store Parameters (0x1010)

Object description

Designation	Store Parameters
SDO ID	0x1010
Type	u32, rw
Value range	0x65766173
Default value	-

If 0x65766173 is written in this object, the entire object directory is written into the EEPROM of the controller. Thus, settings are retained after a restart of the controller. However, a switch-on (Power State machine) is still necessary.

11.25 Restore Parameters (0x1011)

Object description

Designation	Restore Parameters
SDO ID	0x1011
Type	u32, rw
Value range	0x64616F6C
Default value	-

If 0x64616F6C is written in this object, the entire object directory is reset to the default values. The default values are also immediately transferred into the EEPROM.

11.26 Acceleration notation index (0x608D)

Object description

Designation	Acceleration notation index
SDO ID	0x608D
Type	u8, rw
Value range	u8
Default value	0

Reserved.

11.27 Acceleration dimension index (0x608E)

Object description

Designation	Acceleration dimension index
SDO ID	0x608E
Type	s8, rw
Value range	s8
Default value	0

Reserved.

12 Process Data Objects (PDO)

12.1 Purpose of the PDOs

Process Data Objects (PDOs) are used for the transfer of objects that need to be updated frequently during operation of the controller. For example, this is useful for the “Position Actual Value” object.

The advantages of PDOs (compared with SDOs) are the higher and adjustable priority, the low overhead and additional functions such as automatic sending upon a change or cyclical sending.

The higher priority and the low overhead of the PDOs result because the corresponding objects from the object directory are allocated to a CAN object with a certain COB ID without use of the SDO protocol. These allocations are set during the PDO mapping process.

PDOs can be differentiated into receive PDOs and transmit PDOs (RPDO, TPDO). RPDOs are received by the SMCI47-S and the received data are used in the set objects. TPDOs are transmitted by the SMCI47-S in certain (adjustable) situations.

12.2 PDO mapping

General procedure

The mapping of RPDOs/TPDOs is carried out in several steps:

Step	Action
1	Change to the “Pre-Operational” mode.
2	Deactivate the PDO to be changed and reset.
3	Write changed mapping and activate PDO again.
4	Bring the controller into “Operational” mode again in order to use the PDO features and functions.

Example: RPDO Mapping

The changing of an RPDO mapping is described below by way of example. The changing of a TPDO mapping is carried out accordingly. The objects that are necessary for adjusting the other PDOs are described from section 9.10.

Pre-Operational mode

In order for the objects to be mapped, it is necessary to switch into Pre-Operational mode (0x80) first. For more details, see Chapter 2.2.

COB ID	Data bytes	Description
0	80 01	Node 1 after Pre-Operational
701	7F	Response Node 1: Pre-Operational

Deactivating PDO

To deactivate a PDO, the most significant bit (MSB) in the object “Receive/Transmit PDO Communication Parameter : COB ID” must be set.

→ e.g. object 0x1400 Subindex 1 set to (u32)0x80000000 (deactivates RPDO 1).

COB ID	Data bytes	Description
601	23 00 14 01 00 00 00 80	Write COB ID
581	60 00 14 01 00 00 00 00	Response: OK

Set CAN object Identifier (COB ID)

A COB-ID must be assigned for the actual mapping. It is important to note that the COB ID is only assigned to a PDO. The smaller the COB-ID, the higher the priority on the CAN bus.

→ e.g. object 0x1400 Subindex 1 set to (u32)0x80000201.

COB ID	Data bytes	Description
601	23 00 14 01 01 02 00 80	Write COB ID
581	60 00 14 01 00 00 00 00	Response: OK

Set Transmission Type

The Transmission Type is specified in SDO 0x1400 Subindex 2. If the Transmission Type is set to 255, the objects mapped to the PDO are immediately updated on reception of the PDO. If the Transmission Type is set to a value of between 0 and 240, the objects mapped in the PDO are updated when receiving the Sync object.

-> e.g. object 0x1400 Subindex 2 set to (u8)255.

COB ID	Data bytes	Description
601	2F 00 14 02 FF 00 00 00	Write Transmission Type
581	60 00 14 02 00 00 00 00	Response: OK

At this point the Subindices 3 and 5 can then also be written with TPDO. These contain the Inhibit Time and the Event Time.

Deactivating mapping

The number of mapped objects reset to 0.

→ e.g. object 0x1600 subindex 0 set to (u8)0.

COB ID	Data bytes	Description
601	2F 00 16 00 00 00 00 00	Write number of mappings to 0
581	60 00 16 00 00 00 00 00	Response: OK

Changing the mapping

The SDO 0x1600 Subindices 1 to 8 defines which objects are mapped from the object directory to this PDO. Each subindex can be written with a u32 that contains the index and subindex of the object to be mapped. Bits 31 to 16 contain the index and bits 15 to 8 contain the subindex. Bits 7 to 0 specify the length of the object to be mapped in bits. Bits 7 to 0 must not be set, these are adopted from the firmware.

In the following example the control word (0x6040 sub 0) is mapped to RPDO 1:

→ e.g. object 0x1600 Subindex 1 set to (u32)60400000.

COB ID	Data bytes	Description
601	23 00 16 01 00 00 40 60	Write Mapping 1
581	60 00 16 01 00 00 00 00	Response: OK

If several objects are mapped to a PDO, it is important to note that a PDO has a maximum length of 8 bytes. If the maximum length of a PDO is exceeded, the firmware registers an error during mapping.

Activating mapping

For activation of the mapping, it is necessary to write the number of objects to be mapped to SDO 0x1600 Subindex 0.

→ e.g. object 0x1600 Subindex 0 set to (u8)1.

COB ID	Data bytes	Description
601	2F 00 16 00 01 00 00 00	Write number of mappings to 1
581	60 00 16 00 01 00 00 00	Response: OK

Activating PDO

The mapping must be activated as the last step. To do so, the MSB in the SDO 0x1400 Subindex 0 must be deleted.

→ e.g. object 0x1400 Subindex 1 set to (u32)0x201.

COB ID	Data bytes	Description
601	2F 00 14 01 01 02 00 00	Write COB ID
581	60 00 14 01 00 00 00 00	Response: OK

Activation of the node

As the PDOs only function in “Operational” mode, the mode must be changed to this mode after the end of PDO mapping per network management. See chapter .2.2.

COB ID	Data bytes	Description
0	01 01	Node after Operational
701	5	Response Node 1: Operational

Save the settings in the EEPROM

So that the changed PDO mapping is not lost when the power supply is switched off, all settings can be saved in the EEPROM from which these are loaded on each startup.

To do so, the value (u32)0x65766173 must be written to the object 0x1010 Subindex 1 (the value corresponds to the string “save”).

COB ID	Data bytes	Description
601	23 10 10 01 73 61 76 65	Save all settings
581	60 10 10 01 00 00 00 00	Response: OK

12.3 Dummy Objects

Dummy objects (SDO 0x0002 to 0x0007) can be used to hide the parts of an RPDO that are determined for another controller from their own controller.

To do this, only during PDO mapping at the required point must a corresponding dummy SDO be mapped instead of a user data SDO. A dummy object can be used repeatedly. Data that are written to a dummy object have no other effect on the controller, except that the dummy object is written.

13 Commissioning via RS485

In order to operate the controller via the RS485 interface (if present), the corresponding firmware must be installed. Operation of the controller in RS485 mode is not possible with the CANopen firmware.

14 Appendix: Abbreviations used

Abbreviation	Meaning
COB ID	CAN object ID
EMCY	Emergency Object
HM	Homing Mode (Reference Mode)
IP	Interpolated Position Mode
PDO	Process Data Object
PP	Profile Position Mode
ro	read only
RPDO	Receive Process Data Object
rw	read write
SDO	Service Data Object
SYNC	Synchronization Object
TPDO	Transmit Process Data Object
TQ	Torque Mode
VL	Velocity Mode

15 Appendix: Possible error messages

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

16 Appendix: Possible error codes

Error code	Description
0x0000	CAN_EMERGENCY_ERROR_CODE_NO_ERROR_OR_RESET
0x1000	CAN_EMERGENCY_ERROR_CODE_GENERIC_ERROR
0x2000	CAN_EMERGENCY_ERROR_CODE_CURRENT
0x2100	CAN_EMERGENCY_ERROR_CODE_CURRENT_INPUT
0x2200	CAN_EMERGENCY_ERROR_CODE_CURRENT_INSIDE
0x2300	CAN_EMERGENCY_ERROR_CODE_CURRENT_OUTPUT
0x3000	CAN_EMERGENCY_ERROR_CODE_VOLTAGE
0x3100	CAN_EMERGENCY_ERROR_CODE_VOLTAGE_MAINS
0x3200	CAN_EMERGENCY_ERROR_CODE_VOLTAGE_INSIDE
0x3300	CAN_EMERGENCY_ERROR_CODE_VOLTAGE_OUTPUT
0x4000	CAN_EMERGENCY_ERROR_CODE_TEMPERATURE
0x4100	CAN_EMERGENCY_ERROR_CODE_TEMPERATURE_AMBIENT
0x4200	CAN_EMERGENCY_ERROR_CODE_TEMPERATURE_DEVICE
0x5000	CAN_EMERGENCY_ERROR_CODE_DEVICE_HARDWARE
0x6000	CAN_EMERGENCY_ERROR_CODE_DEVICE_SOFTWARE
0x6100	CAN_EMERGENCY_ERROR_CODE_DEVICE_SOFTWARE_INTERNAL
0x6200	CAN_EMERGENCY_ERROR_CODE_DEVICE_SOFTWARE_USER
0x6300	CAN_EMERGENCY_ERROR_CODE_DEVICE_SOFTWARE_DATA
0x7000	CAN_EMERGENCY_ERROR_CODE_ADDITIONAL_MODULES
0x8000	CAN_EMERGENCY_ERROR_CODE_MONITORING
0x8100	CAN_EMERGENCY_ERROR_CODE_MONITORING_COMM
0x8110	CAN_EMERGENCY_ERROR_CODE_MONITORING_COMM_OVERRUN
0x8120	CAN_EMERGENCY_ERROR_CODE_MONITORING_COMM_PASSIVE
0x8130	CAN_EMERGENCY_ERROR_CODE_MONITORING_COMM_LIFEGUARD
0x8140	CAN_EMERGENCY_ERROR_CODE_MONITORING_COMM_BUSRECOVERY
0x8150	CAN_EMERGENCY_ERROR_CODE_MONITORING_COMM_TXCOBCOLLISION
0x8200	CAN_EMERGENCY_ERROR_CODE_MONITORING_PROT
0x8210	CAN_EMERGENCY_ERROR_CODE_MONITORING_PROT_PDO_NOPROCLENGTH
0x8220	CAN_EMERGENCY_ERROR_CODE_MONITORING_PROT_PDO_LENGTH
0x9000	CAN_EMERGENCY_ERROR_CODE_EXTERNAL
0x9100	CAN_EMERGENCY_ERROR_CODE_EXTERNAL_DOORACCESS
0xF000	CAN_EMERGENCY_ERROR_CODE_ADDITIONAL
0xFF00	CAN_EMERGENCY_ERROR_CODE_DEVICESPEC
0xFFFF	CAN_EMERGENCY_ERROR_CODE_DEVICESPEC_UNKNOWN

17 Appendix: Motor data

17.1 Default values for stepper motors

Load angle	Value
1	16384
2	18384
3	20384
4	22384
5	24384
6	26384
7	28384

17.2 Default values for BLDC motors

Load angle	Value
1	16384
2	16500
3	17000
4	17500
5	18000
6	18500
7	19000

17.3 Stepper motors of the series STxxxx

The following table applies to stepper motors of the series ST2018, ST3518, ST4118, ST4209, ST4218, ST5709, ST5909, ST5918, ST6018, ST6318, ST8918, ST11018.

Load angle	Value
1	16384
2	16500
3	17000
4	17500
5	18000
6	18500
7	19000

17.4 BLDC motors of the series DB22

DB22L01

Load angle	Value
1	16000
2	16500
3	17000
4	17500
5	18000
6	18500
7	19000

DB22M01

Load angle	Value
1	16000
2	16500
3	17000
4	17500
5	18000
6	18500
7	18500

17.5 BLDC motors of the series DB28

DB28M01

Load angle	Value
1	16000
2	17000
3	17000
4	17000
5	18000
6	18000
7	18000

DB28S01

Load angle	Value
1	16000
2	16500
3	17000
4	17500
5	18000
6	18500
7	18500

17.6 BLDC motors of the series DB33

DB33S01

Load angle	Value
1	16000
2	16000
3	16500
4	16500
5	17000
6	17000
7	17000

17.7 BLDC motors of the series DB42

DB42C01

Load angle	Value
1	16000
2	18000
3	20000
4	20000
5	20000
6	21000
7	20000

DB42C02

Load angle	Value
1	16000
2	18000
3	20000
4	20000
5	20000
6	21000
7	22000

DB42C03

Load angle	Value
1	16000
2	16500
3	16800
4	17100
5	17400
6	17700
7	17800

DB42L01

Load angle	Value
1	16000
2	17000
3	17500
4	17500
5	17700
6	18300
7	18400

DB42M01

Load angle	Value
1	16000
2	16500
3	17000
4	17500
5	18500
6	18750
7	19000

DB42M02

Load angle	Value
1	16000
2	18000
3	20000
4	20000
5	20000
6	21000
7	22000

DB42M03

Load angle	Value
1	16000
2	17000
3	17000
4	17000
5	18000
6	19000
7	19000

DB42S01

Load angle	Value
1	16000
2	16500
3	17000
4	17500
5	18000
6	18000
7	18500

DB42S02

Load angle	Value
1	16000
2	18000
3	18000
4	18000
5	18500
6	19000
7	19000

DB42S03

Load angle	Value
1	16000
2	18000
3	20000
4	20000
5	20000
6	21000
7	22000

17.8 BLDC motors of the series DB57

DB57C01

Load angle	Value
1	16000
2	16500
3	16500
4	16500
5	17000
6	17000
7	17000

DB57L01

Load angle	Value
1	16000
2	17000
3	17000
4	17000
5	17000
6	17000
7	17000

DB57S01

Load angle	Value
1	16500
2	17000
3	17000
4	17000
5	17000
6	17500
7	17500

17.9 BLDC motors of the series DB87

DB87L01-S

Load angle	Value
1	16384
2	17000
3	17000
4	17000
5	17000
6	17000
7	17000

DB87M01-S

Load angle	Value
1	16384
2	18384
3	20384
4	22384
5	24384
6	26384
7	28384

DB87S01-S

Load angle	Value
1	16000
2	16500
3	17000
4	17250
5	17500
6	17500
7	18000