

CANopen

CANopen firmware for motion control module
TMCM-142

Manual (preliminary)

Version: 0.95
September 15th, 2009



Trinamic Motion Control GmbH & Co KG
Sternstraße 67

D - 20 357 Hamburg, Germany

Phone +49-40-51 48 06 - 0

FAX: +49-40-51 48 06 - 60

<http://www.trinamic.com>

Table of Contents

1	Life support policy	4
2	Overview	5
3	General features.....	5
4	Object dictionary.....	6
4.1	Communication segment	6
4.2	Manufacturer Segment.....	10
4.2.1	Object 0x2000: Microstep resolution.....	10
4.2.2	Object 0x2001: Fullstep resolution	10
4.2.3	Object 0x2002: Brake delay times.....	10
4.2.4	Object 0x2003: Maximum current.....	10
4.2.5	Object 0x2004: Standby current.....	10
4.2.6	Object 0x2005: Limit switches	11
4.2.7	Object 0x2006: Mixed decay threshold	11
4.2.8	Object 0x200a: Enable Drive Delay Time	11
4.2.9	Object 0x200b: Encoder parameters.....	11
4.2.10	Object 0x200c: Brake current feed	11
4.2.11	Object 0x2089: Standby delay.....	12
4.2.12	Object 0x208a: Mixed decay run.....	12
4.2.13	Object 0x208b: Mixed decay standby.....	12
4.2.14	Object 0x208c: Velocity dimension index.....	12
4.2.15	Object 0x208e: Acceleration dimension index	12
4.2.16	Object 0x2100: Home offset display.....	12
4.2.17	Object 0x2101: Actual load value.....	12
4.2.18	Object 0x2102: Driver error flags	13
4.2.19	Object 0x2107: Microstep resolution display.....	13
4.2.20	Object 0x2700: TMCL direct communication	13
4.2.21	Object 0x2701: Manufacturer specific mode	13
4.2.22	Object 0x2702: Digital inputs	13
4.2.23	Object 0x2703: Digital outputs.....	13
4.2.24	Object 0x2704: CAN Bit rate.....	13
4.2.25	Object 0x2705: Node ID	14
4.2.26	Object 0x2706: Store	14
4.2.27	Object 0x2707: CAN bit rate load	14
4.2.28	Object 0x2708: Node ID load	14
4.3	Device profile (CiA-402) objects.....	14
4.3.1	Object 0x6040: Control word.....	14
4.3.2	Object 0x6041: Status word.....	14
4.3.3	Object 0x605a: Quick Stop Option Code	14
4.3.4	Object 0x605b: Shutdown Option Code.....	14
4.3.5	Object 0x605c: Disable Operation Option Code.....	14
4.3.6	Object 0x605d: Halt Option Code.....	14
4.3.7	Object 0x605e: Fault Reaction Option Code	15

4.3.8	Object 0x6060: Modes of Operation	15
4.3.9	Object 0x6061: Modes of Operation Display.....	15
4.3.10	Object 0x6062: Position Demand Value	15
4.3.11	Object 0x6063: Position Actual Internal Value.....	15
4.3.12	Object 0x6064: Position Actual Value	15
4.3.13	Object 0x6065: Following Error Window.....	15
4.3.14	Object 0x6067: Position Window	16
4.3.15	Object 0x6068: Position Window Time	16
4.3.16	Object 0x606A: Sensor selection code.....	16
4.3.17	Object 0x606c: Velocity actual value	16
4.3.18	Object 0x607a: Target Position	16
4.3.19	Object 0x607c: Home Offset.....	16
4.3.20	Object 0x607d: Software Position Limit.....	17
4.3.21	Object 0x6081: Profile Velocity (pp mode).....	17
4.3.22	Object 0x6083: Profile Acceleration (pp mode).....	17
4.3.23	Object 0x6084: Profile Deceleration (pp mode)	17
4.3.24	Object 0x6085: Quick Stop Deceleration	17
4.3.25	Object 0x6086: Motion profile type	17
4.3.26	Object 0x608f: Position Encoder Resolution.....	17
4.3.27	Object 0x6098: Homing Method	17
4.3.28	Object 0x6099: Homing Speeds.....	18
4.3.29	Object 0x609a: Homing Acceleration	18
4.3.30	Object 0x60a4: Profile Jerk	18
4.3.31	Object 0x60fd: Digital Inputs	18
4.3.32	Object 0x60ff: Target velocity (pv mode).....	18
4.3.33	Object 0x6502: Supported drive modes.....	18
5	Emergency Messages.....	19
6	Revision History	21
6.1	Document Revision	21
6.2	Software Revision.....	21
7	References.....	21

1 Life support policy

TRINAMIC Motion Control GmbH & Co. KG does not authorize or warrant any of its products for use in life support systems, without the specific written consent of TRINAMIC Motion Control GmbH & Co. KG.

Life support systems are equipment intended to support or sustain life, and whose failure to perform, when properly used in accordance with instructions provided, can be reasonably expected to result in personal injury or death.

© TRINAMIC Motion Control GmbH & Co. KG 2008

Information given in this data sheet is believed to be accurate and reliable. However no responsibility is assumed for the consequences of its use or for any infringement of patents or other rights of third parties, which may result from its use.

Specifications are subject to change without notice.

2 Overview

The CANopen firmware has been developed to turn the ARM-based single axis motion control modules PD-109, PD-116 and TMCM-103 (in future versions also PD-140) into a CANopen compatible motion control module. The CANopen firmware can be loaded into the modules easily just by using the firmware update function of the TMCL-IDE. The motion control module can then be controlled using the CANopen protocol standard (CiA DS-402 and DS-301).

It is also still possible to control the drive using TMCL commands via a special vendor-specific object. It is also possible to turn the drive back into a TMCL module by loading the TMCL firmware into the drive again. There are versions for the PD-116, the PD-140 and the TMCM-103 modules.

The CANopen firmware is designed to fulfil the latest versions of the DS-301 and DS-402 standards (and the CiA conformance has also been tested).

This manual assumes that the reader is already familiar with the basics of the CANopen protocol (especially DS-301 and DS-402).

3 General features

The general features of this CANopen implementation are:

- Communication according to standard CiA-301 V4.1
- CAN bit rate: 1000kBit/s
- CAN ID: 11 bit
- Node ID: 1..127, settable by switches
- NMT services: NMT slave
- SDO communication:
 - 1 server
 - expedited transfer
 - segmented transfer
 - no block transfer
- PDO communication:
 - Producer
 - Consumer
 - RPDOs: 1, 2, 3, 4, 6
 - Transmission modes: asynchronous
 - Dynamic mapping with max. 3 mapping entries
 - Default mappings: according to CiA-402 for PDO 1, 2, 3 and 6, manufacturer specific for PDO₄
 - TPDOs: 1, 2, 3, 4, 6
 - Transmission modes: asynchronous, asynchronous with event timer, synchronous
 - Dynamic mapping with max. 3 mapping entries
 - Default mappings: according to CiA-402 for PDO 1, 2, 3 and 6, manufacturer specific for PDO
- SYNC: consumer (TPDO₃ and TPDO₆ are synchronous PDOs)
- Emergency: producer
- RTR: not supported

4 Object dictionary

This chapter lists all CANopen objects that are implemented in the current version of the CANopen firmware. Some functions are described in more detail in the next chapters.

There are three groups of object: communication segment object, manufacturer specific objects and DS-402 objects.

4.1 Communication segment

Index	Type	Name	Description
0x1000	U ₃₂ / R	Device Type	Device type, reads 0x40192
0x1001	U ₃₂ / R	Error Register	Last communication error
0x1005	U ₃₂ / R	COB-ID sync	
0x1008	String / R	Manufacturer Device Name	Name of the device, reads "PD-109", "PD-116", "TMCM-103" or "PD-140"
0x1009	String / R	Hardware Version Number	Depends on the device
0x100a	String / R	Software Version Number	e.g. "2.3"
0x1014	U ₃₂ / R	COB-ID emergency	
0x1015	U ₃₂ / R	Inhibit time emergency	
0x1017	U ₃₂ / R	Producer heartbeat time	
0x1018-0	U ₈ / R	Identity object	Number of entries: 3
0x1018-1	U ₃₂ / R	Identity object / Vendor ID	Vendor ID: reads 0x0286
0x1018-2	U ₃₂ / R	Identity object / Product code	Product code: depends on the device, reads 109, 116, 103 or 140.
0x1018-3	U ₃₂ / R	Identity object / Revision number	Firmware revision number; reads 0x20003 for version 2.3
0x1023-0	U ₈ / R	OS command	Number of entries: 3 (for direct TMCL communication).
0x1023-1	Octet string / RW	OS command / Command	TMCL command
0x1023-2	U ₈ / R	OS command / Status	Error code of a TMCL command
0x1023-3	Octet string / R	OS command / Reply	Reply of a TMCL command
0x1029-0	U ₈ / R	Error behaviour	Number of entries: 2
0x1029-1	U ₃₂ / R	Error behaviour / Communication error	
0x1029-2	U ₃₂ / R	Error behaviour / Application error	
0x1400-0	U ₈ / R	RPDO 1 communication parameter	Number of entries: 2
0x1400-1	U ₃₂ / RW	RPDO 1: COB-ID	
0x1400-2	U ₈ / RW	RPDO 1: Transmission Type	
0x1401-0	U ₈ / R	RPDO 2 communication parameter	Number of entries: 2
0x1401-1	U ₃₂ / RW	RPDO 2: COB-ID	
0x1401-2	U ₈ / RW	RPDO 2: Transmission Type	

0x1402-0	U8 / R	RPDO 3 communication parameter	Number of entries: 2
0x1402-1	U32 / RW	RPDO 3: COB-ID	
0x1402-2	U8 / RW	RPDO 3: Transmission Type	
0x1403-0	U8 / R	RPDO 4 communication parameter	Number of entries: 2
0x1403-1	U32 / RW	RPDO 4: COB-ID	
0x1403-2	U8 / RW	RPDO 4: Transmission Type	
0x1405-0	U8 / R	RPDO 5 communication parameter	Number of entries: 2
0x1405-1	U32 / RW	RPDO 5: COB-ID	
0x1405-2	U8 / RW	RPDO 5: Transmission Type	
0x1600-0	U8 / RW	RPDO ₁ mapping parameter	Number of entries: 1..3
0x1600-1	U32 / RW	RPDO 1: Mapping entry 1	
0x1600-2	U32 / RW	RPDO 1: Mapping entry 2	
0x1600-3	U32 / RW	RPDO 1: Mapping entry 3	
0x1601-0	U8 / RW	RPDO 2 mapping parameter	Number of entries: 1..3
0x1601-1	U32 / RW	RPDO 2: Mapping entry 1	
0x1601-2	U32 / RW	RPDO 2: Mapping entry 2	
0x1601-3	U32 / RW	RPDO 2: Mapping entry 3	
0x1602-0	U8 / RW	RPDO 3 mapping parameter	Number of entries: 1..3
0x1602-1	U32 / RW	RPDO 3: Mapping entry 1	
0x1602-2	U32 / RW	RPDO 3: Mapping entry 2	
0x1602-3	U32 / RW	RPDO 3: Mapping entry 3	
0x1603-0	U8 / RW	RPDO 4 mapping parameter	Number of entries: 1..3
0x1603-1	U32 / RW	RPDO 4: Mapping entry 1	
0x1603-2	U32 / RW	RPDO 4: Mapping entry 2	
0x1603-3	U32 / RW	RPDO 4: Mapping entry 3	
0x1605-0	U8 / RW	RPDO 6 mapping parameter	Number of entries: 1..3
0x1605-1	U32 / RW	RPDO 6: Mapping entry 1	
0x1605-2	U32 / RW	RPDO 6: Mapping entry 2	
0x1605-3	U32 / RW	RPDO 6: Mapping entry 3	
0x1800-0	U8 / R	TPDO 1 communication parameter	Number of entries: 5
0x1800-1	U32 / RW	TPDO 1: COB-ID	
0x1800-2	U8 / RW	TPDO 1: Transmission Type	
0x1800-3	U16 / RW	TPDO 1: Inhibit Time	
0x1800-4	U8 / R	TPDO 1: Compatibility Entry	
0x1800-5	U16 / RW	TPDO 1: Event Timer	
0x1801-0	U8 / R	TPDO 2	Number of entries: 5

		communication parameter	
0x1801-1	U32 / RW	TPDO 2: COB-ID	
0x1801-2	U8 / RW	TPDO 2: Transmission Type	
0x1801-3	U16 / RW	TPDO 2: Inhibit Time	
0x1801-4	U8 / R	TPDO 2: Compatibility Entry	
0x1801-5	U16 / RW	TPDO 2: Event Timer	
0x1802-0	U8 / R	TPDO 3 communication parameter	Number of entries: 5
0x1802-1	U32 / RW	TPDO 3: COB-ID	
0x1802-2	U8 / RW	TPDO 3: Transmission Type	
0x1802-3	U16 / RW	TPDO 3: Inhibit Time	
0x1802-4	U8 / R	TPDO 3: Compatibility Entry	
0x1802-5	U16 / RW	TPDO 3: Event Timer	
0x1803-0	U8 / R	TPDO 4 communication parameter	Number of entries: 5
0x1803-1	U32 / RW	TPDO 4: COB-ID	
0x1803-2	U8 / RW	TPDO 4: Transmission Type	
0x1803-3	U16 / RW	TPDO 4: Inhibit Time	
0x1803-4	U8 / R	TPDO 4: Compatibility Entry	
0x1803-5	U16 / RW	TPDO 4: Event Timer	
0x1805-0	U8 / R	TPDO 6 communication parameter	Number of entries: 5
0x1805-1	U32 / RW	TPDO 6: COB-ID	
0x1805-2	U8 / RW	TPDO 6: Transmission Type	
0x1805-3	U16 / RW	TPDO 6: Inhibit Time	
0x1805-4	U8 / R	TPDO 6: Compatibility Entry	
0x1805-5	U16 / RW	TPDO 6: Event Timer	
0x1a00-0	U8 / RW	TPDO 1 mapping parameter	Number of entries: 1..3
0x1a00-1	U32 / RW	TPDO 1: Mapping entry 1	
0x1a00-2	U32 / RW	TPDO 1: Mapping entry 2	
0x1a00-3	U32 / RW	TPDO 1: Mapping entry 3	
0x1a01-0	U8 / RW	TPDO 2: mapping parameter	Number of entries: 1..3
0x1a01-1	U32 / RW	TPDO 2: Mapping entry 1	
0x1a01-2	U32 / RW	TPDO 2: Mapping entry 2	
0x1a01-3	U32 / RW	TPDO 2: Mapping entry 3	
0x1a02-0	U8 / RW	TPDO 3 mapping parameter	Number of entries: 1..3
0x1a02-1	U32 / RW	TPDO 3: Mapping entry 1	
0x1a02-2	U32 / RW	TPDO 3: Mapping entry 2	
0x1a02-3	U32 / RW	TPDO 3: Mapping entry 3	
0x1a03-0	U8 / RW	TPDO 4: mapping parameter	Number of entries: 1..3

0x1a03-1	U32 / RW	TPDO 4: Mapping entry 1	
0x1a03-2	U32 / RW	TPDO 4: Mapping entry 2	
0x1a03-3	U32 / RW	TPDO 4: Mapping entry 3	
0x1a05-0	U8 / RW	TPDO 6: mapping parameter	Number of entries: 1..3
0x1a05-1	U32 / RW	TPDO 6: Mapping entry 1	
0x1a05-2	U32 / RW	TPDO 6: Mapping entry 2	
0x1a05-3	U32 / RW	TPDO 6: Mapping entry 3	

4.2 Manufacturer Segment

The manufacturer segment contains manufacturer specific objects. These objects control the special features of the Trinamic motion control modules.

4.2.1 Object 0x2000: Microstep resolution

Type: UNSIGNED8

Range: 0..11

Default value: 6

This object sets the microstep resolution of the drive. A value of 0 means 2048 microsteps ($2048/2^x$). It is only writeable in the SWITCHED_ON_DISABLED state, but always readable.

4.2.2 Object 0x2001: Fullstep resolution

Type: UNSIGNED16

Read only

Value: 200

This object shows the fullstep resolution of the motor shipped with a PANdrive. It is normally 200 (1.8° motor).

4.2.3 Object 0x2002: Brake delay times

Type: ARRAY OF UNSIGNED16

Sub index 1: Time between applying the brake and disabling the power stage

Sub index 2: Time between releasing the brake and switching the state machine to operational

Both times are given in ms (range 0..65535).

Default values: 0

With this object the delay times for applying and releasing an (optional) brake can be defined. Please see also object 0x200A for an additional delay between enabling the power stage and releasing the brake.

4.2.4 Object 0x2003: Maximum current

Type: UNSIGNED8

Range: 0..15

Default value: 7

This object defines the current used when the motor is moving. A value of 15 means 100% of the maximum current of the drive.

4.2.5 Object 0x2004: Standby current

Type: UNSIGNED8

Range: 0..15

Default value: 0

This object defines the current used when the motor is standing. A value of 15 means 100% of the maximum current of the drive.

4.2.6 Object 0x2005: Limit switches

Type: UNSIGNED₃₂

Range: 0..7

Default value: 0

This object defines which limit switches are to be used. Bit 0 stands for the left and bit 1 stands for the right limit switch. If a bit is set the corresponding limit switch will not be used. So this object has to be set to the value 3 if not limit switches are connected. The object can only be written when the drive is in the SWITCHEDED_ON_DISABLED state (but is always readable).

With bit 2 and bit 3 the limit switches can be inverted. Bit 2 inverts the left limit switch, bit 3 inverts the right limit switch.

4.2.7 Object 0x2006: Mixed decay threshold

Type: UNSIGNED₃₂

Range: -1..2048

Default value: -1

This object defines the behaviour of the mixed decay feature of the drive. When set to -1, fast decay is always used. When set to a value between 0..2047, fast decay will only be used when the internal velocity is higher or equal to that value. When set to 2048, fast decay will never be used.

It is recommended to use -1. Only when StallGuard is to be used this value has to be set to 2048 first.

4.2.8 Object 0x200a: Enable Drive Delay Time

Type: UNSIGNED₁₆

Range: 0..65535

Default Value: 0

This is an additional delay time (in milliseconds) between enabling the power stage and releasing the brake. It can be used to prevent the brake from being released too early (before the hold current in the motor has been reached).

Please see also object 0x2002.

4.2.9 Object 0x200b: Encoder parameters

Type: ARRAY

Sub index 1: (UNSIGNED8) null channel polarity

4.2.10 Object 0x200c: Brake current feed

Type: ARRAY

Sub index 1: (UNSIGNED 8) apply current; default: 0

Sub index 2: (UNSIGNED 8) release current; default: 255

This object configures how much current has to be fed into the brake to apply and to release it. 0 means 0%, 255 means 100% of the maximum current (this depends on the module). In most cases it is needed to feed current into the brake to release it. The default configuration is made for this case.

Please note that on the TCM-103 and TCM-140 modules only 0 (brake output switched off) and 255 (brake output switched on) can be used. Set both values to zero if no brake is used. In such a case the brake output can be used as a normal digital output.

4.2.11 Object 0x2089: Standby delay

Type: UNSIGNED16

Range: 0..4096

Default value: 0

Time after the motor is stopped until the motor current is changed to standby current (in units of 4.096ms). This can only be changed in SWITCHED_ON_DISABLED mode.

4.2.12 Object 0x208a: Mixed decay run

Type: UNSIGNED8

Range: 0..1

Default value: 1

Use mixed decay when the motor is running if set to 1. This can only be changed in SWITCHED_ON_DISABLED mode.

4.2.13 Object 0x208b: Mixed decay standby

Type: UNSIGNED8

Range: 0..1

Default value: 0

Use mixed decay when the motor is standing if set to 1. This can only be changed in SWITCHED_ON_DISABLED mode.

4.2.14 Object 0x208c: Velocity dimension index

Type: UNSIGNED8

Range: 0 or 181

Default value: 0

Writing 0 selects internal units, writing 181 sets PPS units for velocity and PPS/s units for acceleration. This can only be changed in SWITCHED_ON_DISABLED mode.

4.2.15 Object 0x208e: Acceleration dimension index

Type: UNSIGNED8

Read only (0 or 179)

This object reads 0 when internal units are selected or 179 when user units (PPS/s) are selected.

4.2.16 Object 0x2100: Home offset display

4.2.17 Object 0x2101: Actual load value

Type: UNSIGNED8

Read only (0..7)

This object shows the actual StallGuard load value. It is needed for configuring the StallGuard feature, to find the optimal velocity (please see the TMCL manual for more about using StallGuard).

4.2.18 Object 0x2102: Driver error flags

Type: UNSIGNED8

Read only (0..255)

This object shows the hardware error flags of the motor driver IC. There are the following error bits:

4.2.19 Object 0x2107: Microstep resolution display

Type: UNSIGNED8

Read only (0..6)

This object shows the microstep resolution (set by object 0x2000 when internal units are selected or calculated when user units are selected).

4.2.20 Object 0x2700: TMCL direct communication

Type: UNSIGNED32

Make signature: 0x656b616d

After writing the "make signature" 0x656b616d to this object the drive switches to TMCL mode. The drive can then only be controlled via TMCL commands written to the "OS command" object (0x1023/1).

The drive has to be reset (using the DS-301 Reset Application command) to return to normal CANopen functionality.

4.2.21 Object 0x2701: Manufacturer specific mode

Type: UNSIGNED32

Read: 0=manufacturer specific mode is inactive,
1=manufacturer specific mode is active

Write: Make signature: 0x656b616d

Kill signature: 0x6c6c696b

Writing the "make signature" to this object turns on the manufacturer specific mode. The manufacturer specific mode can be turned off again by writing the "kill signature" to this object.

The manufacturer specific mode has the following features:

- PDOs do not need to be disabled and re-enabled when the PDO mapping is to be changed
- The RTR bit in the COB-ID of PDO definitions is ignored

4.2.22 Object 0x2702: Digital inputs

4.2.23 Object 0x2703: Digital outputs

4.2.24 Object 0x2704: CAN Bit rate

Type: UNSIGNED16

Read only (20/50/125/250/500/800/1000)

Shows the selected CAN bit rate.

4.2.25 Object 0x2705: Node ID

Type: UNSIGNED8

Shows the selected node ID.

4.2.26 Object 0x2706: Store

4.2.27 Object 0x2707: CAN bit rate load

4.2.28 Object 0x2708: Node ID load

4.3 Device profile (CiA-402) objects

The following objects are the drive profile objects according to the CiA-402 standard.

4.3.1 Object 0x6040: Control word

Type: UNSIGNED16

The CiA-402 state machine can be controlled using this object.

4.3.2 Object 0x6041: Status word

Type: UNSIGNED16

Read only

This object reflects the status of the CiA-402 state machine.

4.3.3 Object 0x605a: Quick Stop Option Code

Type: SIGNED16

Range: 1, 2, 3, 5, 6 or 7.

Default: 2.

The following quick stop option codes are supported in the current version if the CANopen firmware: 1, 2, 3, 5, 6, or 7.

4.3.4 Object 0x605b: Shutdown Option Code

Type: UNSIGNED16

Read only.

Always reads as 0 as only shutdown option code 0 is supported.

4.3.5 Object 0x605c: Disable Operation Option Code

Type: UNSIGNED16

Read only.

Always reads as 1 as only disable option code 1 is supported.

4.3.6 Object 0x605d: Halt Option Code

Type: UNSIGNED16

Read only.

Always reads as 1 as only halt option code 1 is supported.

4.3.7 Object 0x605e: Fault Reaction Option Code

Type: UNSIGNED16

Read only.

Always reads as 2 as only fault reaction option code 2 is supported.

4.3.8 Object 0x6060: Modes of Operation

Type: SIGNED8

Range: 0, 1, 3 or 6.

This object defines the operating mode. Supported operating modes are:

- 0: No mode. The motor will not run when the operating mode is set to 0 or it will be stopped when the motor is running in one of the supported operating modes and the operating mode is then switched to 0.
- 1: Profile positing mode (pp)
- 3: Profile velocity mode (pv)
- 6: Homing mode (hm)

Other modes are not supported and thus cannot be set.

4.3.9 Object 0x6061: Modes of Operation Display

Type: SIGNED8

Read only

This object shows the operating mode that is set.

4.3.10 Object 0x6062: Position Demand Value

Type: SIGNED32

Read only

This is the actual position that the motor should have. It is not to be confused with objects 0x6063 and object 0x6064.

4.3.11 Object 0x6063: Position Actual Internal Value

Type: SIGNED32

Read only

The actual position of the encoder, re-scaled to the microstep resolution of the motor.

4.3.12 Object 0x6064: Position Actual Value

Type: SIGNED32

Read only

This object always contains the same value as object 0x6063.

4.3.13 Object 0x6065: Following Error Window

Type: UNSIGNED32

Range: 0..8388607

When the difference between motor position (object 0x6062) and encoder position (object 0x6063 or 0x6064) is greater than the value set here, the motor will be stopped and an emergency message will be sent. Setting this object to zero will turn off this feature completely. Setting this object to a too low value will lead to false alarms.

4.3.14 Object 0x6067: Position Window

Type: UNSIGNED₃₂

Range: 0..8388607

Default: 0

If this object is set to zero, the target reached event will be signalled when the demand position (0x6062) has reached the target position (0x607a).

When the position window is set to a value greater than zero, the target reached event will be signalled when the actual encoder position value (0x6064) is within the target position – position window and target position + position window.

4.3.15 Object 0x6068: Position Window Time

Type: UNSIGNED₃₂

Default: 0

If this object is set to a value greater than zero and also the position window (0x6067) is set to a value greater than zero the target reached event will not be signalled until the actual position (0x6064) is at least as many milliseconds within the position window as defined by this object.

4.3.16 Object 0x606A: Sensor selection code

Type: INTEGER₁₆

Default: -1

This object selects if an encoder is to be used. Set to 0 to use an encoder or set it to -1 for no encoder. It can only be modified in SWITCHED_ON_DISABLED mode.

4.3.17 Object 0x606c: Velocity actual value

Type: INTEGER₃₂

Read only

This object shows the actual velocity of the motor.

4.3.18 Object 0x607a: Target Position

Type: SIGNED₃₂

Range: -2147483647..+2147483647

This is the target position used for driving the motor in positioning mode.

4.3.19 Object 0x607c: Home Offset

Type: SIGNED₃₂

Range: -2147483647..+2147483647

Offset between the home switch and the zero position. The effect of setting the home position to a non zero value depends on the selected homing method.

4.3.20 Object 0x607d: Software Position Limit

Type: ARRAY

Sub index 1: Minimum software position limit (SIGNED₃₂, Range -2147483647..+2147483647)

Sub index 2: Maximum software position limit (SIGNED₃₂, Range -2147483647..+2147483647)

Two limits for moving the motor in positioning mode can be set here. It will then not be possible to leave this window.

4.3.21 Object 0x6081: Profile Velocity (pp mode)

Type: UNSIGNED₃₂

Range: depends on the units.

This is the maximum velocity used when driving to a new position (in profile positioning mode).

4.3.22 Object 0x6083: Profile Acceleration (pp mode)

Type: UNSIGNED₃₂

Range: depends on the units

This object sets the maximum acceleration to be used in profile positioning mode.

4.3.23 Object 0x6084: Profile Deceleration (pp mode)

Type: UNSIGNED₃₂

Range: depends on the units

This object sets the maximum deceleration to be used in profile positioning mode.

4.3.24 Object 0x6085: Quick Stop Deceleration

Type: UNSIGNED₃₂

Range: depends on the unit setting.

This object defines the deceleration that is to be used during a quick stop operation.

4.3.25 Object 0x6086: Motion profile type

Type: UNSIGNED₁₆

Range: 0 or 1

Set this object to 0 for trapezoid ramps or 1 for S-shaped ramps.

4.3.26 Object 0x608f: Position Encoder Resolution

Type: ARRAY

Sub-index 1: Encoder increments per motor revolution (0..2147483647)

Sub-index 2: read only (always 1)

4.3.27 Object 0x6098: Homing Method

Type: SIGNED₈

Range: 0, 3, 5, 19 or 21.

The homing method to be used in homing mode can be selected here. Homing methods supported by the current firmware are:

- 0: The current position is defined as the home position.
- 3: Search home switch in positive direction, then search the next encoder index pulse.

- 5: Search home switch in negative direction, then search the next encoder index pulse.
- 19: Search the home switch in positive direction.
- 21: Search the home switch in negative direction.
- 33: Search next index pulse in positive direction
- 34: Search next index pulse in negative direction

Other homing methods are not supported by the current firmware, but will be added in later firmware releases.

4.3.28 Object 0x6099: Homing Speeds

Type: ARRAY

Sub index 1: fast homing speed (UNSIGNED₃₂, range depends on the unit setting).

Sub index 2: slow homing speed (UNSIGNED₃₂, range depends on the unit setting).

Using this object a fast and a slow homing speed can be set. In most homing modes, the home switch is searched with the fast speed first. When the home switch has been found, the motor will be decelerated to the slow speed (using the homing acceleration, object 0x609a) and then stopped at the exact switch point.

4.3.29 Object 0x609a: Homing Acceleration

Type: UNSIGNED₃₂

Range: depends on the unit setting.

With this object the acceleration used for homing (for accelerating to the fast homing speed and for decelerating to the slow homing speed) can be set.

4.3.30 Object 0x60a4: Profile Jerk

Type: ARRAY

Sub-index 1: bow of S-shaped ramp (1..18)

This object sets the bow of S-shaped ramps. It can only be used when S-shaped ramps are selected (by setting object 0x6086 to 1).

4.3.31 Object 0x60fd: Digital Inputs

Type: UNSIGNED₃₂

Read only.

This object contains the states of the digital inputs of the module. Starting from bit 0, every bit reflects the state of one digital input. The number of valid bits depends on the number of digital inputs on the module used.

4.3.32 Object 0x60ff: Target velocity (pv mode)

Type: INTEGER₃₂

Range: -2147483647..+2147483647 (depends on selected units)

This object sets the target velocity when using profile velocity mode. The drive then accelerates or decelerates to that velocity using the acceleration and deceleration set by objects 0x6083 and 0x6084.

4.3.33 Object 0x6502: Supported drive modes

Type: UNSIGNED₃₂

Read only.

This object always contains the value 0x0000025 which means that the modes 0 (no mode), 1 (profile position mode), 3 (profile velocity mode) and 6 (homing mode) are supported by the drive.

5 Emergency Messages

Error code	Add. byte					Description
	1	2	3	4	5	
0x0000	0	0	0	0	0	Fault reset The fault reset command has been executed.
0x1000	1	0	0	0	0	Generic error: open load bridge A The motor driver indicates open load on bridge A. It is possible that the motor cable is broken or that there is an error in the power amplifier itself.
0x1000	2	0	0	0	0	Generic error: open load bridge B The motor driver indicates open load on bridge B. It is possible that the motor cable is broken or that there is an error in the power amplifier itself.
0x2310	0	0	0	0	0	Overcurrent high side The motor driver indicates an overcurrent on the high side. This can be caused by a short circuit in the driver stage.
0x2311	0	0	0	0	0	Overcurrent bridge B The motor driver indicates that there is overcurrent on bridge B. This can be caused by a short circuit in the motor itself or in the motor driver stage.
0x2312	0	0	0	0	0	Overcurrent bridge A The motor driver indicates that there is overcurrent on bridge A. This can be caused by a short circuit in the motor itself or in the motor driver stage.
0x3230	0	0	0	0	0	StallGuard error The actual load value exceeds the StallGuard limit.
0x4310	1	0	0	0	0	Overtemperature pre-warning The temperature in the motor driver exceeds the pre-warning limit.
0x4310	2	0	0	0	0	Overtemperature error The motor driver has been switched off because the temperature limit has been exceeded.
0x5441	0	0	0	0	0	Shutdown switch active The enable signal is missing (due to the shutdown switch) and the motor driver has been switched off.
0x6320	0	0	0	0	0	Parameter error The data in the received PDO is either wrong or cannot be accepted due to the internal state of the drive.
0x8110	1	0	0	0	0	CAN controller overflow The receive message buffer of the CAN controller hardware is full and some CAN messages are lost.
0x81110	2	0	0	0	0	CAN Tx buffer overflow The software CAN transmit buffer is full and thus some CAN messages are lost.
0x81110	3	0	0	0	0	CAN Rx buffer overflow The software CAN receive buffer is full and so some CAN messages are lost.
0x8120	0	0	0	0	0	CAN error passive The CAN controller has detected communication errors and has entered the CAN Error passive state.

0x8140	0	0	0	0	0	<p>CAN controller recovered from bus-off state</p> <p>The CAN controller had detected too many errors and had changed into the bus-off state. The drive has been stopped and disabled. This message is sent after the CAN controller has recovered from bus-off state and is bus-on again.</p>
0x8611	0	0	0	0	0	<p>Following error</p> <p>The deviation between motor position counter and encoder position counter has exceeded the following error window.</p>
0xff00	0	0	0	0	0	<p>Undervoltage</p> <p>The supply voltage is too low to drive a motor.</p>
0xff01	1	0	0	0	0	<p>Positive software limit</p> <p>The actual position is outside the range defined by object 0x607d.</p>
0xff01	2	0	0	0	0	<p>Negative software limit</p> <p>The actual position is outside the range defined by object 0x607d.</p>
0xff01	3	0	0	0	0	<p>Positive limit switch</p> <p>The positive limit switch has been touched outside of the homing function.</p>
0xff01	4	0	0	0	0	<p>Negative limit switch</p> <p>The negative limit switch has been touched outside of the homing function.</p>

6 Revision History

6.1 Document Revision

Version	Comment	Author	Description
0.95	15-Sep-09	OK	Initial version for TMC457 based modules

Table 6.1 : Document Revision

6.2 Software Revision

Version	Comment	Description
3.00	15-Sep-09	First version for TMC457 based modules

Table 6.2: Software Revision

7 References

[TMC-142] TMC-142 manual