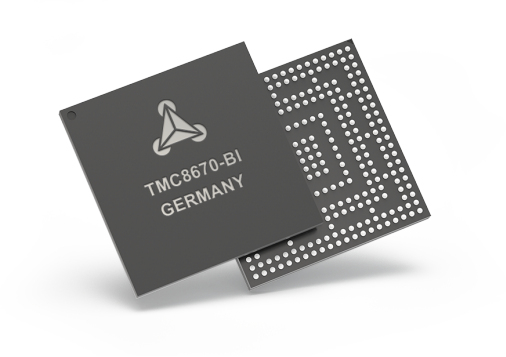
TMC8670 Datasheet

Firmware Version V1.0 | Document Revision V1.00 • 2016-Nov-22

The TMC8670 is a CANopen-over EtherCAT (CoE) field oriented control (FOC) servo controller for torque, velocity, and position control. It comes with a fully integrated EtherCAT Slave Controller (ESC), a flexible sensor engine for different position feedback and current sensing options, as well as a complete CANopen-over-EtherCAT firmware stack for the CiA DS402 device profile. TMC8670 is a building block that enables a servo controller with only a couple of components.



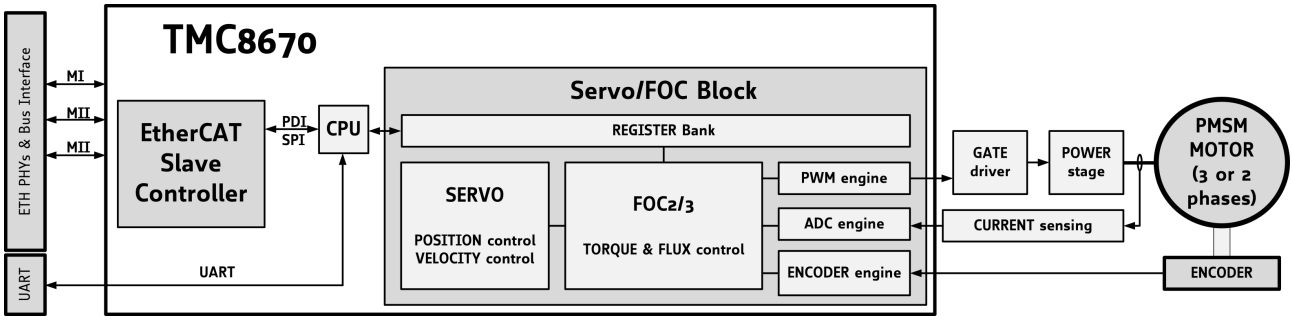
Features

- Field Oriented Control (FOC) Servo Controller
- Torque Control (FOC), Velocity Control, Position Control
- Sensor Engine (Hall analog/digital, Encoder analog/digital)
- Support for 3-Phase PMSM and 2-Phase Stepper Motors
- PWM Engine including SVPWM
- Integrated EtherCAT Slave Controller, CoE protocol CiA 402 drive profile
- UART interface

Applications

- Robotics
- Pick and Place Machines
- Semiconductor Handling
- Factory Automation
- E-Mobility
- Laboratory Automation
- Blowers
- Manufacturing
- Pumps

Simplified Block Diagram



Read entire documentation.

TRINAMIC
MOTION CONTROL
Now part of Maxim Integrated

Contents

1 CANopen-over-EtherCAT (CoE)

2 Preface

This document specifies objects and modes of operation of the Trinamic TMC8670 BLDC/PMSM motor control module with CANopen-over-EtherCAT (CoE) firmware. The CoE firmware is designed to fulfill the EtherCAT version of the CANopen DS402 standards. The EtherCAT conformance has also been tested. This manual assumes that the reader is already familiar with the basics of EtherCAT and the CoE protocol (especially DS402).

2.1 General Features of this CoE Implementation

Main Characteristics

- Communication according to EtherCAT standards
- Protocols: CoE

SDO Communication

- 1 server
- Expedited transfer
- Segmented transfer
- No block transfer

PDO Communication

- Producer
- Consumer
- RPDOs
  - Dynamic mapping with max. 9 mapping entries.
  - Default mappings: manufacturer specific.
- TPDOs
  - Dynamic mapping with max. 9 mapping entries.
  - Default mappings: manufacturer specific.

Sync managers

- Sync manager 0: receive mailbox used for SDO communication
- Sync manager 1: send mailbox used for SDO communication
- Sync manager 2: process data output (used for TPDO)
- Sync manager 3: process data input (used for RPDO)

Further Characteristics

- Emergency: producer

2.2 Abbreviations used in this Manual

Abbreviations	
CAN	Controller area network
CoE	CANopen over EtherCAT
CHGND	chassis ground / earth ground
COB	Communication object
FoE	File transfer over EtherCAT
FSA	Finite state automaton
FSM	Finite state machine
NMT	Network management
ID	Identifier
LSB	Least significant bit
MSB	Most significant bit
PDO	Process data object
PDS	Power drive system
RPDO	Receive process data object
SDO	Service data object
TPDO	Transmit process data object
EMCY	Emergency object
rw	Read and write
ro	Read only
hm	Homing mode
pp	Profile position mode
pv	Profile velocity mode
vm	Velocity mode

Table 1: Abbreviations used in this Manual

2.3 Firmware Update

The software running on the microprocessor consists of two parts, a boot loader and the CoE firmware itself. Whereas the boot loader is installed during production and testing at TRINAMIC and remains untouched throughout the whole lifetime, the CoE firmware can easily be updated by the user. The new firmware can be loaded into the module via the firmware update function of the TMCL-IDE, using the TTL-UART interface of the IC, or by EtherCAT FoE funktionality.

3 Communication

3.1 Reference Model

The application layer comprises a concept to configure and communicate real-time-data as well as the mechanisms for synchronization between devices. The functionality which the application layer offers to an application is logically divided over different service data objects (SDO) in the application layer. A service object offers a specific functionality and all the related services.

Applications interact by invoking services of a service object in the application layer. To realize these services this object exchanges data via the EtherCAT with peer service object(s) using a protocol.

The application and the application layer interact with service primitives.

Service Primitives	
Primitive	Definition
Request	Issued by the application to the application layer to request a service.
Indication	Issued by the application layer to the application to report an internal event detected by the application layer or indicate that a service is requested.
Response	Issued by the application to the application layer to respond to a previous received indication.
Confirmation	Issued by the application layer to the application to report the result of a previously issued request.

Table 2: Service Primitives

A service type defines the primitives that are exchanged between the application layer and the cooperating applications for a particular service of a service object. Unconfirmed and confirmed services are collectively called remote services.

Service Types	
Type	Definition
Local service	Involves only the local service object. The application issues a request to its local service object that executes the requested service without communicating with peer service object(s).
Unconfirmed service	Involves one or more peer service objects. The application issues a request to its local service object. This request is transferred to the peer service object(s) that each passes it to their application as an indication. The result is not confirmed back.
Confirmed service	Can involve only one peer service object. The application issues a request to its local service object. This request is transferred to the peer service object that passes it to the other application as an indication. The other application issues a response that is transferred to the originating service object that passes it as a confirmation to the requesting application.
Provider initiated service	Involves only the local service object. The service object (being the service provider) detects an event not solicited by a requested service. This event is then indicated to the application.

Table 3: Service Types

3.2 NMT State Machine

The finite state machine (FSM) or simply state machine is a model of behavior composed of a finite number of states, transitions between those states, and actions. It shows which way the logic runs when certain conditions are met.

Starting and resetting the device is controlled via the state machine. The NMT state machine consists of the states shown in figure 1.

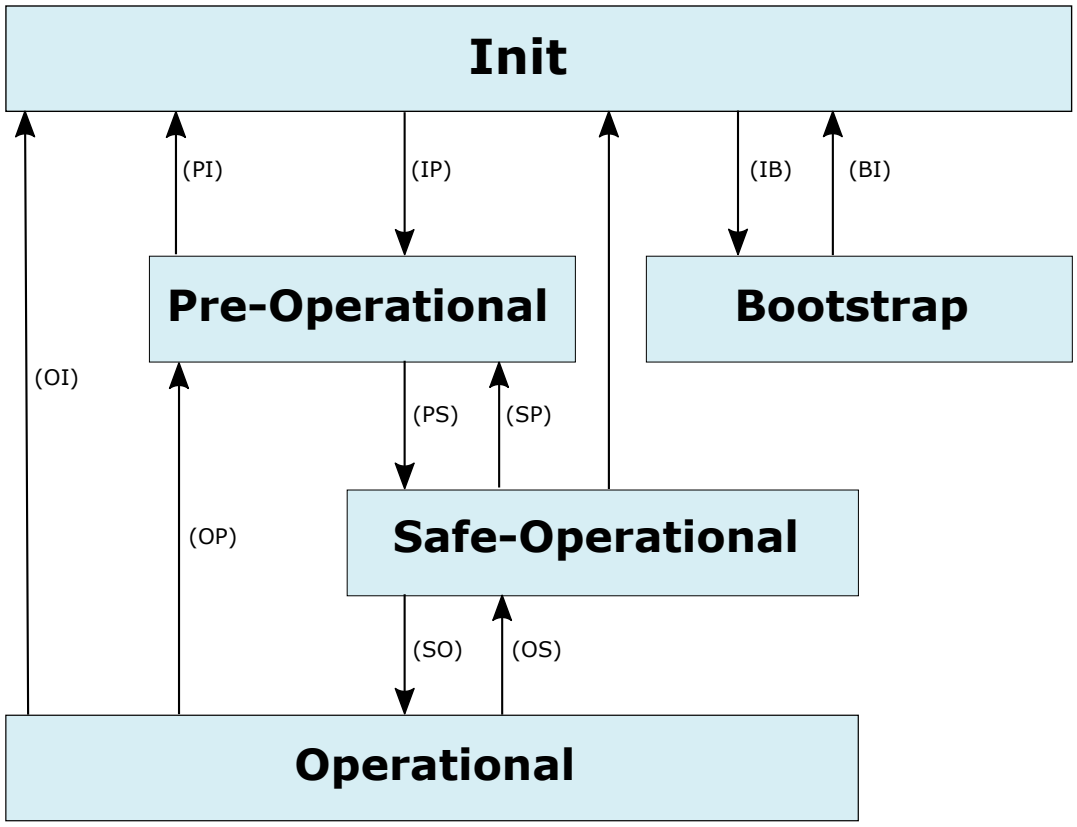


Figure 1: NMT State Machine

After power-on or reset the device enters the Initialization **(INIT)** state.

The master can then switch the device to Pre-Operational **(PRE-OP)** state. In this state, only SDO communication is possible. PDO communication is not possible.

In Safe-Operational **(SAFE-OP)** state, also PDO communication is possible. Inputs can be read, but outputs cannot be switched and the motor cannot be run.

In Operational **(OP)** state, all features of the module can be used. PDO communication is possible, outputs can be switched and the motor can be used. During Operational state the device can use all supported communication objects.

When switching from Operational to Safe-Operational state the motor will be stopped if it has been running. When the EtherCAT connection is lost during Operational state the device will also automatically

switch to Safe-Operational state.

The Bootstrap **(BOOT)** state is used for firmware updates via FoE. Before FoE can be used the device has to be switched to this state.

3.3 Device Model

A CoE device mainly consists of the following parts:

- *Communication:* This function unit provides the communication objects and the appropriate functionality to transport data items via the underlying network structure.
- *Object dictionary:* The object dictionary is a collection of all the data items which have an influence on the behavior of the application objects, the communication objects and the state machine used on this device.
- *Application:* The application comprises the functionality of the device with respect to the interaction with the process environment.

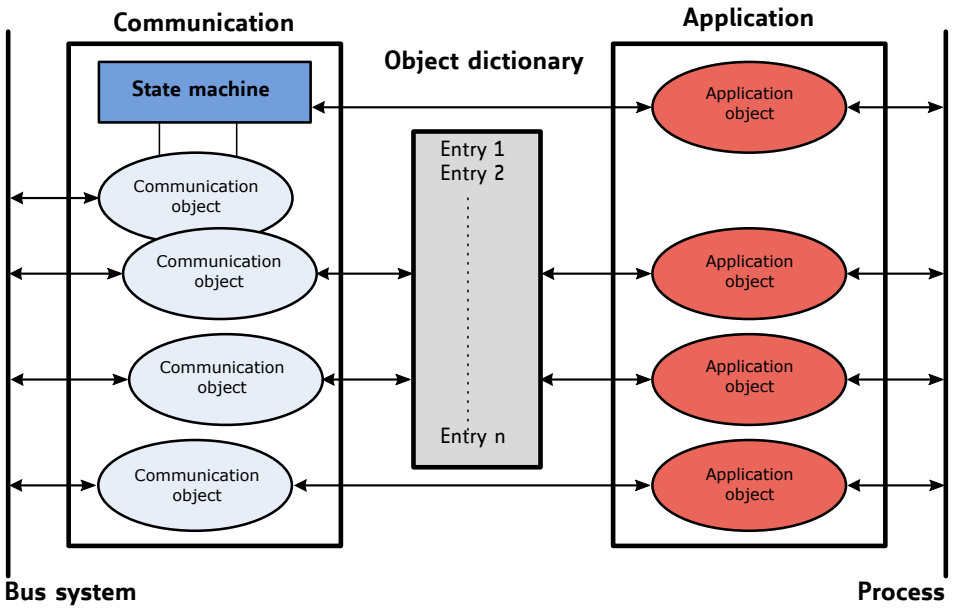


Figure 2: Device Model

3.4 Object Dictionary

The most important part of a device profile is the object dictionary description. The object dictionary is essentially a grouping of objects accessible via the network in an ordered pre-defined fashion. Each object within the dictionary is addressed using a 16-bit index. The overall layout of the standard object dictionary is shown in table 4:

Object Dictionary	
Index	Object
0000_h	Not used.
0001_h – $001F_h$	Static data types.
0020_h – $003F_h$	Complex data types.
0040_h – $005F_h$	Manufacturer specific complex data types.
0060_h – $007F_h$	Device profile specific static data types.
0080_h – $009F_h$	Device profile specific complex data types.
$00A0_h$ – $0FFF_h$	Reserved for further use.
1000_h – $1FFF_h$	Communication profile area.
2000_h – $5FFF_h$	Manufacturer specific profile area.
6000_h – $9FFF_h$	Standardized device profile area.
$A000_h$ – $BFFF_h$	Standardized interface profile area.
$C000_h$ – $FFFF_h$	Reserved for further use.

Table 4: Object Dictionary

The communication profile area at indices 1000_h through $1FFF_h$ contains the communication specific parameters for the CAN network. These entries are common to all devices.

The manufacturer segment at indices 2000_h through $5FFF_h$ contains manufacturer specific objects. These objects control the special features of the Trinamic TMC8670 motion control device.

The standardized device profile area at indices 6000_h through $9FFF_h$ contains all data objects common to a class of devices that can be read or written via the network. They describe the device parameters and the device functionality of the device profile.

4 Communication Area

The communication area contains all objects that define the communication parameters of the CoE device according to the EtherCAT standard.

4.1 Detailed Object Specifications

4.1.1 Object 1000_h: Device Type

This object contains information about the device type. The object 1000_h describes the type of device and its functionality. It is composed of a 16-bit field which describes the device profile that is used and a second 16-bit field which provides additional information about optional functionality of the device.

Object Description			
Index	Name	Object Type	Data Type
1000_h	Device type	Variable	UNSIGNED32

Table 5: Object Description (1000_h)

Entry Description				
Sub-index	Access	PDO Mapping	Value Range	Default Value
0	ro	no	UNSIGNED32	$FFFC0192_h$

Table 6: Entry Description (1000_h)

4.1.2 Object 1001_h: Error Register

This object contains error information. The CANopen device maps internal errors into object 1001_h. It is part of an emergency object.

Object Description			
Index	Name	Object Type	Data Type
1001_h	Error register	Variable	UNSIGNED8

Table 7: Object Description (1001_h)

Entry Description				
Sub-index	Access	PDO Mapping	Value Range	Default Value
0	ro	no	UNSIGNED8	0

Table 8: Entry Description (1001_h)

Error Register Bits	
Bit	Definition
0	Generic error
1	Current
2	Voltage
3	Temperature
4	Communication error
5	Device profile specific
6	Reserved (always 0)
7	Manufacturer specific

Table 9: Error Register Bits

4.1.3 Object 1008_h: Manufacturer Device Name

This object contains the name of the device as given by the manufacturer.

Object Description			
Index	Name	Object Type	Data Type
1008_h	Manufacturer Device Name	Variable	Visible String

Table 10: Object Description (1008_h)

Entry Description				
Sub-index	Access	PDO Mapping	Value Range	Default Value
0	ro	no	—	TMC8670

Table 11: Entry Description (1008_h)

4.1.4 Object 1009_h: Manufacturer Hardware Version

This object contains the hardware version description.

Object Description			
Index	Name	Object Type	Data Type
1009_h	Manufacturer Hardware Version	Variable	Visible String

Table 12: Object Description (1009_h)

Entry Description				
Sub-index	Access	PDO Mapping	Value Range	Default Value
0	ro	no	—	Depends on device, e.g. 1.0.

Table 13: Entry Description (1009_h)

4.1.5 Object $100A_h$: Manufacturer Software Version

This object contains the software version description.

Object Description			
Index	Name	Object Type	Data Type
$100A_h$	Manufacturer Software Version	Variable	Visible String

Table 14: Object Description ($100A_h$)

Entry Description				
Sub-index	Access	PDO Mapping	Value Range	Default Value
0	ro	no	—	Depends on device, e.g. 1.0.

Table 15: Entry Description ($100A_h$)

4.1.6 Object 1018_h: Identity Object

The object 1018_h contains general information about the device:

- The vendor ID (sub-index 01_h) contains a unique value allocated to each manufacturer. The vendor ID of Trinamic is 286_h.
- The manufacturer specific product code (sub-index 2_h) identifies a specific device version.
- The manufacturer specific revision number (sub-index 3_h) consists of a major revision number and a minor revision number.

Object Description			
Index	Name	Object Type	Data Type
1018_h	Identity object	Record	Identity

Table 16: Object Description (1018_h)

Entry Description					
Sub-index	Description	Access	PDO Mapping	Value Range	Default Value
00_h	Number of entries	ro	no	0…3	3
01_h	Vendor ID	ro	no	UNSIGNED32	0286_h
02_h	Product code	ro	no	UNSIGNED32	8670
03_h	Revision number	ro	no	UNSIGNED32	e.g. 20003_h for version 2.3

Table 17: Entry Description (1018_h)

4.1.7 Object 1600_h: Receive PDO Mapping Parameter

This object contains the mapping parameters for the RPDO the device is able to receive. The sub-index 00_h contains the number of valid entries within the mapping record. This number of entries is also the number of the application variables which shall be received with the corresponding RPDO. The sub-indices from 01_h to the number of entries contain the information about the mapped application variables. These entries describe the PDO contents by their index, sub-index and length.

Object Description			
Index	Name	Object Type	Data Type
1600_h	Receive PDO mapping parameter	RECORD	PDO Mapping

Table 18: Object Description (1600_h)

Entry Description				
Sub-index	Description	Access	Value Range	Default Value
00_h	Number of mapped application objects in PDO	rw	0…9	Index 1600_h: 4
01_h	Mapping entry 1	rw	UNSIGNED32	60400010_h
02_h	Mapping entry 3	rw	UNSIGNED32	$607A0020_h$
03_h	Mapping entry 4	rw	UNSIGNED32	60710010_h
04_h	Mapping entry 5	rw	UNSIGNED32	$60FF0020_h$
05_h	Mapping entry 2	rw	UNSIGNED32	0_h
06_h	Mapping entry 6	rw	UNSIGNED32	0_h
07_h	Mapping entry 7	rw	UNSIGNED32	0_h
08_h	Mapping entry 8	rw	UNSIGNED32	0_h
09_h	Mapping entry 9	rw	UNSIGNED32	0_h

Table 19: Entry Description (1600_h)

4.1.8 Objects $1A00_h$: Transmit PDO Mapping Parameter

This object contains the mapping parameters for the TPDO the device is able to transmit. The sub-index 00_h contains the number of valid entries within the mapping record. This number of entries is also the number of the application variables which shall be transmitted with the corresponding TPDO. The sub-indices from 01_h to the number of entries contain the information about the mapped application variables. These entries describe the PDO contents by their index, sub-index and length.

Object Description			
Index	Name	Object Type	Data Type
$1A00_h$	Transmit PDO mapping parameter	RECORD	PDO Mapping

Table 20: Object Description ($1A00_h$)

Entry Description				
Sub-index	Description	Access	Value Range	Default Value
00_h	Number of mapped aapli-cation objects in PDO	rw	0...9	6
01_h	Mapping entry 1	rw	UNSIGNED32	60410010_h
02_h	Mapping entry 2	rw	UNSIGNED32	60610008_h
03_h	Mapping entry 3	rw	UNSIGNED32	60640020_h
04_h	Mapping entry 4	rw	UNSIGNED32	60770010_h
05_h	Mapping entry 5	rw	UNSIGNED32	$606C0020_h$
06_h	Mapping entry 6	rw	UNSIGNED32	$60FD0020_h$
07_h	Mapping entry 7	rw	UNSIGNED32	0_h
08_h	Mapping entry 8	rw	UNSIGNED32	0_h
09_h	Mapping entry 9	rw	UNSIGNED32	0_h

Table 21: Entry Description ($1A00_h$)

4.1.9 Objects $1C00_h$: Sync Manager Communication Type

This object describes the communication types of the EtherCAT sync managers. The types of the first four synch managers are normally fixed and should not be changed. Sync managers can have the following for communication types:

Sync Manager Communication Types	
Type	Description
1	Mailbox receive
2	Mailbox send
3	Process data input
4	Process data output

Table 22: Sync Manager Communication Types

Object Description			
Index	Name	Object Type	Data Type
$1C00_h$	Sync manager communication type	RECORD	UNSIGNED8

Table 23: Object Description ($1C00_h$)

Entry Description				
Sub-index	Description	Access	Value Range	Default Value
00_h	Number of entries	rw	0…3	4
01_h	Communication type sync manager 1	rw	UNSIGNED8	1
02_h	Communication type sync manager 2	rw	UNSIGNED8	2
03_h	Communictaion type sync manager 3	rw	UNSIGNED8	3
04_h	Communictaion type sync manager 4	rw	UNSIGNED8	4

Table 24: Entry Description ($1C00_h$)

4.1.10 Objects $1C12_h$: Sync Manager 2 PDO Assignment

This object contains the index of the PDO definiton object that is assigned to sync manager 2. Normally, the RPDO objects are assigned to sync manager 2. Under most cicumstances there is no need to change this setting.

Object Description			
Index	Name	Object Type	Data Type
$1C12_h$	Sync manager 2 PDO assignment	RECORD	PDO assignment

Table 25: Object Description ($1C12_h$)

Entry Description				
Sub-index	Description	Access	Value Range	Default Value
00_h	Number of assigned PDOs	rw	0...1	1
01_h	PDO mapping index of assigned RPDO	rw	UNSIGNED16	1600_h

Table 26: Entry Description ($1C12_h$)

4.1.11 Objects $1C13_h$: Sync Manager 3 PDO Assignment

This object contains the index of the PDO definiton object that is assigned to sync manager 3. Normally, the TPDO objects are assigned to sync manager 3. Under most cicumstances there is no need to change this setting.

Object Description			
Index	Name	Object Type	Data Type
$1C13_h$	Sync manager 3 PDO assignment	RECORD	PDO assignment

Table 27: Object Description ($1C13_h$)

Entry Description				
Sub-index	Description	Access	Value Range	Default Value
00_h	Number of assigned PDOs	rw	0...1	1
01_h	PDO mapping index of assigned TPDO	rw	UNSIGNED16	$1A00_h$

Table 28: Entry Description ($1C13_h$)

5 Manufacturer specific area

The manufacturer segment contains manufacturer specific objects. These objects control the special features of the Trinamic Motion Control device TMC8670.

5.1 Detailed object specifications

5.1.1 Object 2000_h: Device Info

Object Description			
Index	Name	Object Type	Data Type
2000_h	Device Info	Variable	Record

Table 29: Object Description (2000_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
1	MC_Type	no	0	4294967295	0		R
2	MC_Version	no	0	4294967295	0		R
3	MC_Date	no	0	4294967295	0		R
4	MC_Time	no	0	4294967295	0		R
5	MC_Variant	no	0	4294967295	0		R

Table 30: Entry Description (2000_h)

5.1.2 Object 2001_h: Device State

Object Description			
Index	Name	Object Type	Data Type
2001_h	Device State	Variable	Record

Table 31: Object Description (2001_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
1	Status_Flags	no	0	4294967295	0		R
2	Warning_Mask	no	0	4294967295	0		RW
3	Error_Mask	no	0	4294967295	0		RW
4	Enable_Driver	no	0	255	0		RW

Table 32: Entry Description (2001_h)

5.1.3 Object 2010_h: PWM Settings

Object Description			
Index	Name	Object Type	Data Type
2010_h	PWM Settings	Variable	Record

Table 33: Object Description (2010_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
1	Polarities	no	0	255	0		RW
2	MAX_Count	no	0	65535	0		RW
3	PWM_BBM_L	no	0	255	0		RW
4	PWM_BBM_H	no	0	255	0		RW
5	PWM_Chop	no	0	255	0		RW
6	PWM_SV	no	0	255	0		RW

Table 34: Entry Description (2010_h)

5.1.4 Object 2020_h: Motor Settings

Object Description			
Index	Name	Object Type	Data Type
2020_h	Motor Settings	Variable	Record

Table 35: Object Description (2020_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
1	Motor_Type	no	0	255	0		RW
2	N_Pole_Pairs	no	0	65535	0		RW

Table 36: Entry Description (2020_h)

5.1.5 Object 2030_h: ADC Config

Object Description			
Index	Name	Object Type	Data Type
2030_h	ADC Config	Variable	Record

Table 37: Object Description (2030_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
1	ADC_Select	no	0	65535	0		RW
2	ADC_I0	no	0	65535	0		R
3	ADC_I1	no	0	65535	0		R
4	ADC_I0_Offset	no	-32768	32767	0		RW
5	ADC_I0_Scale	no	-32768	32767	0		RW
6	ADC_I1_Offset	no	-32768	32767	0		RW
7	ADC_I1_Scale	no	-32768	32767	0		RW
8	ADC_I0_Scaled	no	-32768	32767	0		R
9	ADC_I1_Scaled	no	-32768	32767	0		R
10	ADC_I2_Scaled	no	-32768	32767	0		R
11	Real_World_Factor	no	1	65535	256		RW

Table 38: Entry Description (2030_h)

5.1.6 Object 2040_h: Open Loop Mode Settings

Object Description			
Index	Name	Object Type	Data Type
2040_h	Open Loop Mode Settings	Variable	Record

Table 39: Object Description (2040_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
1	Mode	no	0	65535	0		RW
2	Acceleration	no	0	4294967295	0	[rpm/s]	RW
3	Velocity_Target	no	-2147483648	2147483647	0	[rpm/s]	RW
4	Velocity_Actual	no	-2147483648	2147483647	0	[rpm]	R
5	PHI	no	-32768	32767	0		RW
0	UD_Ext	no	-32768	32767	0		RW
0	UQ_Ext	no	-32768	32767	0		RW

Table 40: Entry Description (2040_h)

5.1.7 Object 2041_h: Torque Mode Settings

Object Description			
Index	Name	Object Type	Data Type
2041_h	Torque Mode Settings	Variable	Record

Table 41: Object Description (2041_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
1	Torque_Target	no	-2147483648	2147483647	0	[mA]	R
2	Torque_Actual	no	-2147483648	2147483647	0	[mA]	R
3	Flux_Target	no	-2147483648	2147483647	0	[mA]	RW
4	Flux_Actual	no	-2147483648	2147483647	0	[mA]	R
5	Torque_P	no	0	32767	0		RW
6	Torque_I	no	0	32767	0		RW
7	Flux_P	no	0	32767	0		RW
8	Flux_I	no	0	32767	0		RW
9	PID_Torque_Error	no	-2147483648	2147483647	0		R
10	PID_Torque_Error_Sum	no	-2147483648	2147483647	0		R
11	PID_Flux_Error	no	-2147483648	2147483647	0		R
12	PID_Flux_Error_Sum	no	-2147483648	2147483647	0		R
13	PHI_E	no	-32768	32767	0		R

Table 42: Entry Description (2041_h)

5.1.8 Object 2042_h: Velocity Mode Settings

Object Description			
Index	Name	Object Type	Data Type
2042_h	Velocity Mode Settings	Variable	Record

Table 43: Object Description (2042_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
1	Velocity_Target	no	-2147483648	2147483647	0	[rpm]	R
2	Velocity_Actual	no	-2147483648	2147483647	0	[rpm]	R
3	Velocity_P	no	0	32767	0		RW
4	Velocity_I	no	0	32767	0		RW
5	PID_Velocity_Error	no	-2147483648	2147483647	0		R
6	PID_Velocity_Error_Sum	no	-2147483648	2147483647	0		R

Table 44: Entry Description (2042_h)

5.1.9 Object 2043_h: Position Mode Settings

Object Description			
Index	Name	Object Type	Data Type
2043_h	Position Mode Settings	Variable	Record

Table 45: Object Description (2043_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
1	Position_Target_ABS	no	-2147483648	2147483647	0		R
2	Position_Target_REL	no	-2147483648	2147483647	0		R
3	Position_Actual	no	-2147483648	2147483647	0		RW
4	Position_P	no	0	32767	0		RW
5	Position_I	no	0	32767	0		RW
6	PID_Position_Error	no	-2147483648	2147483647	0		R
7	PID_Position_Eroor_Sum	no	-2147483648	2147483647	0		R

Table 46: Entry Description (2043_h)

5.1.10 Object 2050_h: Limits

Object Description			
Index	Name	Object Type	Data Type
2050_h	Limits	Variable	Record

Table 47: Object Description (2050_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
1	UQ_UD_Limits	no	0	32767	23169		RW
2	Torque_Flux_DDT_Limits	no	0	32767	32767		RW
3	Torque_Flux_Limits_Raw	no	0	32767	32767		R
4	Torque_Flux_Limits	no	0	2147483647	2147483647	[mA]	RW
5	Acceleration_Limit	no	0	2147483647	2147483647	[rpm/s]	RW
6	Velocity_Limit	no	0	2147483647	2147483647	[rpm]	RW
7	Position_Limit_Low	no	-2147483648	2147483647	-2147483648		RW
8	Position_Limit_High	no	-2147483648	2147483647	2147483647		RW

Table 48: Entry Description (2050_h)

5.1.11 Object 2060_h: Selections

Object Description			
Index	Name	Object Type	Data Type
2060_h	Selections	Variable	Record

Table 49: Object Description (2060_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
1	PHI_E_Selection	no	0	255	0		RW
2	Velocity_Selection	no	0	255	0		RW
3	Position_Selection	no	0	255	0		RW
4	Motion_Mode	no	0	255	0		RW
5	Ramp_Mode	no	0	255	0		RW

Table 50: Entry Description (2060_h)

5.1.12 Object 2070_h: External Assignments

Object Description			
Index	Name	Object Type	Data Type
2070_h	External Assignments	Variable	Record

Table 51: Object Description (2070_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
1	PHI_E_Ext	no	-32768	32767	0		RW
2	PHI_M_Ext	no	-32768	32767	0		RW
3	Position_Ext	no	-2147483648	2147483647	0		RW

Table 52: Entry Description (2070_h)

5.1.13 Object 2080_h: Digital Hall

Object Description			
Index	Name	Object Type	Data Type
2080_h	Digital Hall	Variable	Record

Table 53: Object Description (2080_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
1	Mode	no	0	65535	0		RW
2	Hall_Position_000	no	-32768	32767	0		RW
3	Hall_Position_060	no	-32768	32767	0		RW
4	Hall_Position_120	no	-32768	32767	0		RW
5	Hall_Position_180	no	-32768	32767	0		RW
6	Hall_Position_240	no	-32768	32767	0		RW
7	Hall_Position_300	no	-32768	32767	0		RW
8	PHI_E	no	-32768	32767	0		R
9	PHI_E_Offset	no	-32768	32767	0		RW
10	PHI_E_Interpolated	no	-32768	32767	0		R
11	dPHI_Max	no	0	65535	0		RW

Table 54: Entry Description (2080_h)

5.1.14 Object 2081_h: ABN Encoder 1

Object Description			
Index	Name	Object Type	Data Type
2081_h	ABN Encoder 1	Variable	Record

Table 55: Object Description (2081_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
1	Mode	no	0	65535	0		RW
2	PPR	no	0	4294967295	0		RW
3	Count	no	0	16777215	0		RW
4	Count_N	no	0	16777215	0		RW
5	PHI_E	no	-32768	32767	0		R
6	PHI_E_Offset	no	-32768	32767	0		RW
7	PHI_M	no	-32768	32767	0		R
8	PHI_M_Offset	no	-32768	32767	0		RW
9	Init_Mode	no	0	2	0		RW
10	Init_Voltage	no	0	32767	6000		RW
11	Init_Time	no	0	65535	1000		RW

Table 56: Entry Description (2081_h)

5.1.15 Object 2082_h: ABN Encoder 2

Object Description			
Index	Name	Object Type	Data Type
2082_h	ABN Encoder 2	Variable	Record

Table 57: Object Description (2082_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
1	Mode	no	0	65535	0		RW
2	PPR	no	0	4294967295	0		RW
3	Count	no	0	16777215	0		RW
4	Count_N	no	0	16777215	0		RW
5	PHI_M	no	-32768	32767	0		R
6	PHI_M_Offset	no	-32768	32767	0		RW

Table 58: Entry Description (2082_h)

5.1.16 Object 2100_h: Analog Inputs

Object Description			
Index	Name	Object Type	Data Type
2100_h	Analog Inputs	Variable	Record

Table 59: Object Description (2100_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
1	ADC_I0_Raw	no	0	65535	0		R
2	ADC_I1_Raw	no	0	65535	0		R
3	ADCSPI_I0_Raw	no	0	65535	0		R
4	ADCSPI_I1_Raw	no	0	65535	0		R
5	ADCSD_I0_Raw	no	0	65535	0		R
6	ADCSD_I1_Raw	no	0	65535	0		R
7	ADC_I0_Ext	no	0	65535	0		RW
7	ADC_I1_Ext	no	0	65535	0		RW

Table 60: Entry Description (2100_h)

5.1.17 Object 2110_h: Digital Inputs

Object Description			
Index	Name	Object Type	Data Type
2110_h	Digital Inputs	Variable	Record

Table 61: Object Description (2110_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
1	Inputs_Raw	no	0	4294967295	0		R

Table 62: Entry Description (2110_h)

5.1.18 Object 2120_h: Digital Outputs

Object Description			
Index	Name	Object Type	Data Type
2120_h	Digital Outputs	Variable	Record

Table 63: Object Description (2120_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
1	Outputs_Raw	no	0	4294967295	0		RW

Table 64: Entry Description (2120_h)

5.1.19 Object 2130_h: Evaluation

Object Description			
Index	Name	Object Type	Data Type
2130_h	Evaluation	Variable	Record

Table 65: Object Description (2130_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
1	Disable_PWM	no	0	255	123		RW
2	Load_BLDC_Defaults	no	0	255	123		RW
3	Load_Stepper_Defaults	no	0	255	123		RW
4	Start_Open_Loop_Test	no	0	255	123		RW
5	Load_Hall_Config	no	0	255	123		RW
6	Load_Hall_Invert_Config	no	0	255	123		RW
7	Start_ABN_Init_Mode_0	no	0	255	123		RW
8	Start_ABN_Init_Mode_1	no	0	255	123		RW
9	Start_ABN_Init_Mode_2	no	0	255	123		RW
10	Load_ABN_Config	no	0	255	123		RW

Table 66: Entry Description (2130_h)

6 Profile specific area

The profile segment contains CiA-402 standard motion control objects. These objects control the motion control functions of the TMC8670. Since it is not possible to operate the modes in parallel, the user is able to activate the required function by selecting a mode of operation. The control device writes to the modes of operation object in order to select the operation mode. The drive device provides the modes of operation display object to indicate the actual activated operation mode. Controlword, statusword, and set-points are used mode-specific. This implies the responsibility of the control device to avoid inconsistencies and erroneous behavior.

The following operating modes (selectable via object 6060_h, please see 6.1.1) are implemented on the TMC8670:

- Homing mode (hm)
- Cyclic synchronous position mode (csp)
- Cyclic synchronous velocity mode (csv)
- Cyclic synchronous torque mode (cst)

6.1 Detailed object specifications

The following text offers detailed object specifications. For a better understanding, it is necessary to see how the state machine works.

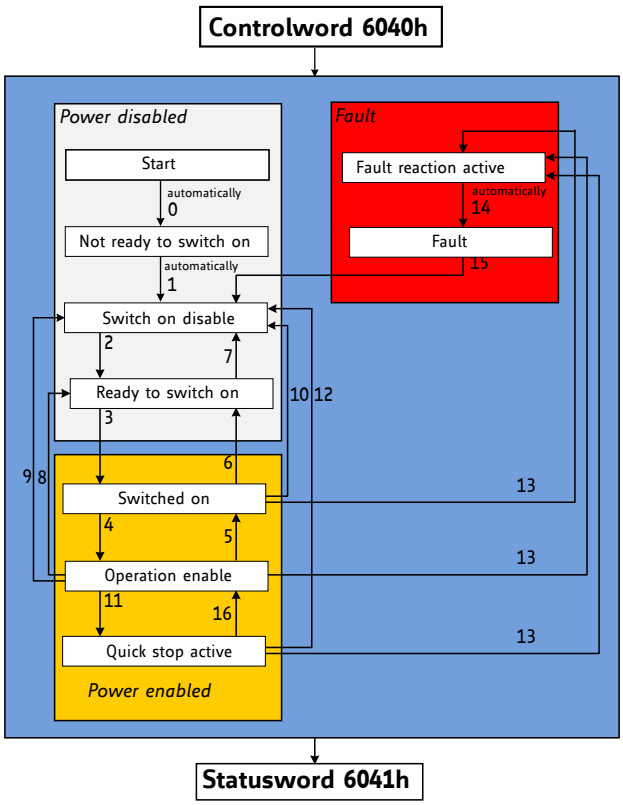


Figure 3: DS402 Finite State Machine

Notes on state transitions:

- Commands directing a change in state are processed completely and the new state achieved before additional state change commands are processed.
- Transitions 0 and 1 occur automatically at drive power-on or reset. Transition 14 occurs automatically, too. All other state changes must be directed by the host.
- Drive function disabled indicates that no current is being supplied to the motor.
- Drive function enabled indicates that current is available for the motor and profile position and profile velocity reference values may be processed.

6.1.1 Object 6060_h: Modes of operation

This object indicates the requested operation mode. Supported operating modes are:

Value Definition	
Value	Mode
0	No mode
6	Homing mode (hm)
8	Cyclic synchronous position mode (csp)
9	Cyclic synchronous velocity mode (csv)
10	Cyclic synchronous torque mode (cst)

Table 67: Value Description (6060_h)

The motor will not run when the operating mode is set to 0. 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.

Object Description			
Index	Name	Object Type	Data Type
6060_h	Modes of operation	Variable	SIGNED8

Table 68: Object Description (6060_h)

Entry Description				
Sub-index	Access	PDO Mapping	Value Range	Default Value
0	rw	refer to CiA-402	0/6/8/9/10	0

Table 69: Entry Description (6060_h)

6.1.2 Object 6061_h: Modes of Operation Display

This object shows the operating mode that is currently set.

Value Definition	
Value	Mode
0	No mode
6	Homing mode (hm)
8	Cyclic synchronous position mode (csp)
9	Cyclic synchronous velocity mode (csv)
10	Cyclic synchronous torque mode (cst)

Table 70: Value Description (6061_h)

The motor will not run when the operating mode is set to 0. 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.

Object Description			
Index	Name	Object Type	Data Type
6061_h	Modes of operation display	Variable	SIGNED8

Table 71: Object Description (6061_h)

Entry Description				
Sub-index	Access	PDO Mapping	Value Range	Default Value
0	rw	refer to CiA-402	0/6/8/9/10	0

Table 72: Entry Description (6061_h)

6.1.3 Object 6502_h: Supported Drive Modes

This object provides information on the supported drive modes (0: not supported, 1: supported). This object is organized bit-wise. The bits have the following meaning:

Bit 0: profile position mode
Bit 1: velocity mode
Bit 2: profile velocity mode
Bit 3: profile torque mode
Bit 4: reserved
Bit 5: homing mode
Bit 6: interpolated position mode
Bit 7: cyclic synchronous position mode
Bit 8: cyclic synchronous velocity mode
Bit 9: cyclic synchronous torque mode
Bit 10-15: reserved
Bit 16-31: manufacturer-specific

Object Description			
Index	Name	Object Type	Data Type
6502_h	Supported Drive Modes	Variable	UNSIGNED32

Table 73: Object Description (6502_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	Supported Drive Modes	no	0	4294967295	0		R

Table 74: Entry Description (6502_h)

7 Profile Position Mode

A target position is applied to the trajectory generator. It is generating a position demand value for the position control loop described in the position control function.

Please refer to object 6060_h (section 6.1.1) for information about how to choose an operation mode. Object 6061_h (section 6.1.2) shows the operation mode that is set.

7.1 Detailed Object Specifications

The following text offers detailed object specifications. For a better understanding, it is necessary to see how the state machine works.

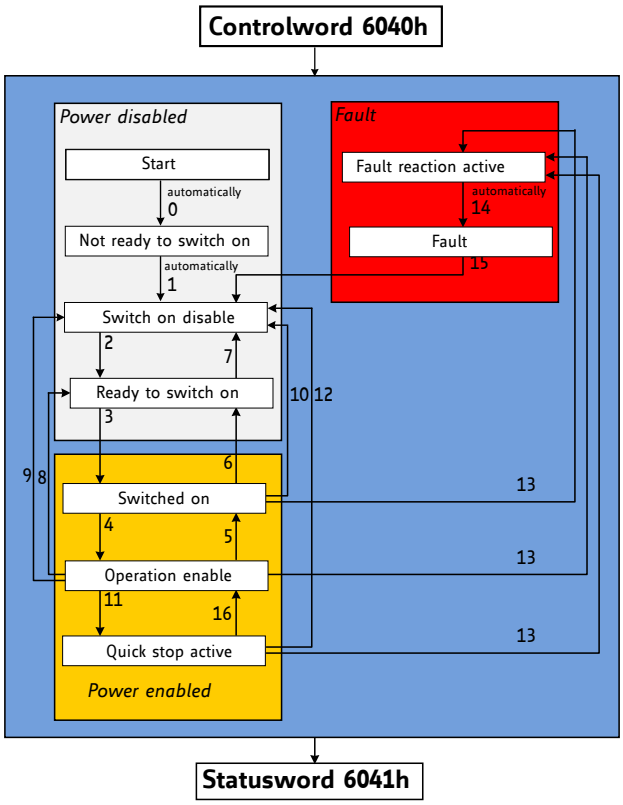


Figure 4: DS402 Finite State Machine

Notes on state transitions:

- Commands directing a change in state are processed completely and the new state achieved before additional state change commands are processed.
- Transitions 0 and 1 occur automatically at drive power-on or reset. Transition 14 occurs automatically, too. All other state changes must be directed by the host.
- Drive function disabled indicates that no current is being supplied to the motor.
- Drive function enabled indicates that current is available for the motor and profile position and profile velocity reference values may be processed.

7.2 Detailed Object Specifications

7.2.1 Object 6040_h: Control Word

This object indicates the received command controlling the power drive system finite state automaton (PDS FSA). The CiA-402 state machine can be controlled using this object. Please refer to figure 4 for detailed information.

Structure of the Control Word										
15 11	10	9	8	7	6 4	3	2	1	0	
nu	r	oms	h	fr	oms	eo	qs	ev	so	
MSB										LSB

Legend: nu=not used; r=reserved; oms=operation mode specific; h=halt; fr=fault reset; eo=enable operation; qs=quick stop; ev=enable voltage; so=switch on.

Table 75: Structure of the Control Word in pp Mode

Operation Mode specific Bits in pp Mode		
Bit	Name	Definition
4	New set point	0-to-1: the next positioning will be started.
5	Change immediately	Not supported.
6	Absolute / relative	0: New position is absolute. 1: New position is relative.
9	Change set point	Not supported.

Table 76: Operation Mode specific Bits in pp Mode

Command Coding						
Command	Bits of Control Word					Transitions
	Bit 7	Bit 3	Bit 2	Bit 1	Bit 0	
Shutdown	0	x	1	1	0	2,6,8
Switch on	0	0	1	1	1	3
Switch on & enable operation	0	1	1	1	1	3, 4
Disable voltage	0	x	x	0	x	7,9,10,12
Quick stop	0	x	0	1	x	7,10,11
Disable operation	0	0	1	1	1	5
Enable operation	0	1	1	1	1	4, 16
Fault reset	0-to-1	x	x	x	x	15

Table 77: Command Coding

Object Description			
Index	Name	Object Type	Data Type
6040_h	Controlword	Variable	UNSIGNED16

Table 78: Object Description (6040_h in pp Mode)

Entry Description				
Sub-index	Access	PDO Mapping	Value Range	Default Value
0	rw	see CiA402-3	See command coding above.	

Table 79: Entry Description (6040_h in pp Mode)

7.2.2 Object 6041_h: Status Word

This object provides the status of the PDS FSA. It reflects the status of the CiA-402 state machine. Please refer to figure 4 for detailed information. The object is structured as defined below.
For more information about the coding please refer to the CANopen Drives and motion control device profile, part 2.

Structure of the Status Word															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
dir	mot	oms		ila	tr	rm	ms	w	sod	qs	ve	f	oe	so	rtso
MSB															LSB

Legend: nu=not used; r=reserved; oms=operation mode specific; h=halt; fr=fault reset; oe=operation enable; qs=quick stop; ve=voltage enable; so=switch on.

Table 80: Structure of the Staus Word in pp Mode

Trinamic Specific Bits		
Bit	Name	Definition
14	Motor activity	0: Motor stands still. 1: Motor rotates.
15	Direction of rotation	This bit shows the direction of rotation.

Table 81: Trinamic Specific Bits

Operation Mode specific Bits in pp Mode		
Bit	Name	Definition
10	Target reached	Set when the motor is within the position window.
12	Set point acknowledged	0: Set point processed. 1: Set point still in process.
13	Following error	Not supported.

Table 82: Operation Mode specific Bits in pp Mode

State Coding	
Status word	FSA state
xxxx xxxx x0xx 0000_h	Not ready to switch on
xxxx xxxx x1xx 0000_h	Switch on disabled
xxxx xxxx x01x 0001_h	Ready to switch on
xxxx xxxx x01x 0011_h	Switched on
xxxx xxxx x01x 0111_h	Operation enabled
xxxx xxxx x00x 0111_h	Quick stop active
xxxx xxxx x0xx 1111_h	Fault reaction active
xxxx xxxx x0xx 1000_h	Fault

Table 83: State Coding

Object Description			
Index	Name	Object Type	Data Type
6041_h	Controlword	Variable	UNSIGNED16

Table 84: Object Description (6041_h in pp Mode)

Entry Description				
Sub-index	Access	PDO Mapping	Value Range	Default Value
0	rw	see CiA402-3	See state coding above.	

Table 85: Entry Description (6041_h in pp Mode)

7.2.3 Object 6062_h: Position Demand Value

This object provides the demanded position value. The value is given in hall or encoder steps. Object 6062h indicates the actual position that the motor should have. It is not to be confused with objects 6063h and 6064h.

Object Description			
Index	Name	Object Type	Data Type
6062_h	Position Demand Value	Variable	SIGNED32

Table 86: Object Description (6062_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	Position Demand Value	yes	-2147483648	2147483647	0		R

Table 87: Entry Description (6062_h)

7.2.4 Object 6063_h: Position Actual Internal Value

This object provides the actual position value of the motor.

Object Description			
Index	Name	Object Type	Data Type
6063_h	Position Actual Internal Value	Variable	SIGNED32

Table 88: Object Description (6063_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	Position Actual Internal Value	no	-2147483648	2147483647	0		R

Table 89: Entry Description (6063_h)

7.2.5 Object 6064_h: Position Actual Value

This object provides the actual value of the position measurement device. It always contains the same value as object 6063h.

Object Description			
Index	Name	Object Type	Data Type
6064_h	Position Actual Value	Variable	SIGNED32

Table 90: Object Description (6064_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	Position Actual Value	yes	-2147483648	2147483647	0		R

Table 91: Entry Description (6064_h)

7.2.6 Object 6067_h: Position Window

This object indicates the configured symmetrical range of accepted positions relative to the target position. If the actual value of the position encoder is within the position window, this target position is regarded as having been reached. The value is given in increments. If the value of the position window is FFFFFFFFh, the position window control is switched off. If this object is set to zero, the target reached event will be signaled when the demand position (6062h) has reached the target position (6064h). When the position window is set to a value greater than zero, the target reached event will be signaled when the actual encoder position value (6064h) is within (target_position - position_window) and (target_position + position_window).

Object Description			
Index	Name	Object Type	Data Type
6067_h	Position Window	Variable	UNSIGNED32

Table 92: Object Description (6067_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	Position Window	no	0	4294967295	4294967295		RW

Table 93: Entry Description (6067_h)

7.2.7 Object $606C_h$: Velocity Actual Value

This object shows the actual velocity value derived from the velocity sensor.

Object Description			
Index	Name	Object Type	Data Type
$606C_h$	Velocity Actual Value	Variable	SIGNED32

Table 94: Object Description ($606C_h$)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	Velocity Actual	no	-2147483648	2147483647	0	[rpm]	R

Table 95: Entry Description ($606C_h$)

7.2.8 Object $607A_h$: Target Position

The target position is the position that the drive should move to in profile position mode using the actual settings of motion control parameters (such as velocity, acceleration, deceleration, etc.). The value of this object is interpreted as absolute or relative depending on the abs/rel flag in the controlword.

Object Description			
Index	Name	Object Type	Data Type
$607A_h$	Target Position	Variable	SIGNED32

Table 96: Object Description ($607A_h$)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	Target Position	yes	-2147483648	2147483647	0		RW

Table 97: Entry Description ($607A_h$)

7.2.9 Object $607D_h$: Software Position Limit

This object indicates the configured maximal and minimal software position limits. These parameters define the absolute position limits for the position demand value and the position actual value. Every new target position is checked against these limits. The limit positions are always relative to the machine home position. Before being compared with the target position, they are corrected internally by the home offset as follows:

Corrected min position limit = min position limit - home offset
Corrected max position limit = max position limit - home offset

Object Description			
Index	Name	Object Type	Data Type
$607D_h$	Software Position Limit	Array	SIGNED32

Table 98: Object Description ($607D_h$)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
1	Min Position Limit	no	-2147483648	2147483647	-2147483648		RW
2	Max Position Limit	no	-2147483648	2147483647	2147483647		RW

Table 99: Entry Description ($607D_h$)

7.2.10 Object $607F_h$: Max Profile Velocity

Max. absolute velocity for velocity and positioning mode.

Object Description			
Index	Name	Object Type	Data Type
$607F_h$	Max Profile Velocity	Variable	UNSIGNED32

Table 100: Object Description ($607F_h$)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	MaxVelocity	no	0	200000	4000	[rpm]	RW

Table 101: Entry Description ($607F_h$)

7.2.11 Object 6081_h: Profile Velocity

This object indicates the configured velocity normally attained at the end of the acceleration ramp during a profiled motion and is valid for both directions of motion. The profile velocity is the maximum velocity used when driving to a new position.

Object Description			
Index	Name	Object Type	Data Type
6081_h	Profile Velocity	Variable	UNSIGNED32

Table 102: Object Description (6081_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	Profile Velocity	no	0	200000	0	[rpm]	RW

Table 103: Entry Description (6081_h)

7.2.12 Object 6082_h: End Velocity

This object indicates the configured velocity normally attained at the end of the deceleration ramp during a profiled motion and is valid for both directions of motion. The end velocity is the velocity used when reaching the new position.

Object Description			
Index	Name	Object Type	Data Type
6082_h	End Velocity	Variable	SIGNED32

Table 104: Object Description (6082_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	End Velocity	no	-200000	200000	0	[rpm]	RW

Table 105: Entry Description (6082_h)

7.2.13 Object 6083_h: Profile Acceleration

This object indicates the configured acceleration. Object 6083h sets the maximum acceleration to be used in profile positioning mode, and profile velocity mode.

Object Description			
Index	Name	Object Type	Data Type
6083_h	Profile Acceleration	Variable	UNSIGNED32

Table 106: Object Description (6083_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	Profile Acceleration	no	0	100000	2000	[rpm/s]	RW

Table 107: Entry Description (6083_h)

7.2.14 Object 6084_h: Profile Deceleration

This object indicates the configured deceleration.

Object Description			
Index	Name	Object Type	Data Type
6084_h	Profile Deceleration	Variable	UNSIGNED32

Table 108: Object Description (6084_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	Profile Deceleration	no	0	100000	2000	[rpm/s]	RW

Table 109: Entry Description (6084_h)

7.2.15 Object 6085_h: Quick Stop Deceleration

This object indicates the configured deceleration used to stop the motor when the quick stop function is activated and the quick stop code object 605Ah is set to 2 (or 6).

Object Description			
Index	Name	Object Type	Data Type
6085_h	Quick Stop Deceleration	Variable	UNSIGNED32

Table 110: Object Description (6085_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	Quick Stop Deceleration	no	0	100000	2000	[rpm/s]	RW

Table 111: Entry Description (6085_h)

7.3 How to move a Motor in pp Mode

Here is a little example that shows how to get a motor running in pp mode. In this little example we assume that the module has been reset (and then switched to pre-operational or operational) by NMT commands before. Please note that the values are decimal.

- Select pp mode by writing 1 to object 6060_h (Modes_of_Operation).
- Write 6 to object 6040_h (Controlword) to switch to READY_TO_SWITCH_ON state.
- Write 7 to object 6040_h to switch to SWITCHED_ON state.
- Write the desired target position (e.g. 50000) to object $607A_h$ (Target_Position).
- Write 15 to object 6040_h to switch to OPERATION_ENABLED state. The motor now accelerates to the target position.

8 Profile Velocity Mode

The profile velocity mode is used to control the velocity of the drive without a special regard of the position. It contains limit functions and trajectory generation.

The profile velocity mode covers the following sub-functions:

- Demand value input via trajectory generator.
- Monitoring of the profile velocity using a window-function.
- Monitoring of velocity actual value using a threshold.

The operation of the reference value generator and its input parameters include:

- Profile velocity
- Profile acceleration
- Motion profile type

8.1 Detailed Object Specifications

8.1.1 Object 6040_h: Control Word

This object indicates the received command controlling the power drive system finite state automaton (PDS FSA). The CiA-402 (www.can-cia.org/can-knowledge/canopen/cia402) state machine can be controlled using this object. Please refer to figure 4 for detailed information.

In pv mode the control word does not contain any operation mode specific bits.

Structure of the Control Word											
15	11	10	9	8	7	6	4	3	2	1	0
nu		r	r	h	fr	r		eo	qs	ev	so
MSB											LSB

Legend: nu=not used; r=reserved; h=halt; fr=fault reset; eo=enable operation; qs=quick stop; ev=enable voltage; so=switch on.

Table 112: Structure of the Control Word in pv Mode

Command Coding						
Command	Bits of Control Word					Transitions
	Bit 7	Bit 3	Bit 2	Bit 1	Bit 0	
Shutdown	0	x	1	1	0	2,6,8
Switch on	0	0	1	1	1	3
Switch on & enable operation	0	1	1	1	1	3, 4
Disable voltage	0	x	x	0	x	7,9,10,12
Quick stop	0	x	0	1	x	7,10,11
Disable operation	0	0	1	1	1	5
Enable operation	0	1	1	1	1	4, 16
Fault reset	0-to-1	x	x	x	x	15

Table 113: Command Coding

Object Description			
Index	Name	Object Type	Data Type
6040_h	Controlword	Variable	UNSIGNED16

Table 114: Object Description (6040_h in pv Mode)

Entry Description				
Sub-index	Access	PDO Mapping	Value Range	Default Value
0	rw	see CiA402-3	See command coding above.	

Table 115: Entry Description (6040_h in pv Mode)

8.1.2 Object 6041_h: Status Word

This object provides the status of the PDS FSA. It reflects the status of the CiA-402 state machine. Please refer to figure 4 for detailed information. The object is structured as defined below.
For more information about the coding please refer to the CANopen Drives and motion control device profile, part 2.

Structure of the Status Word															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
dir	mot	oms		ila	tr	rm	ms	w	sod	qs	ve	f	oe	so	rtso
MSB															LSB

Legend: nu=not used; r=reserved; oms=operation mode specific; h=halt; fr=fault reset; oe=operation enable; qs=quick stop; ve=voltage enable; so=switch on.

Table 116: Structure of the Status Word in pv Mode

Trinamic Specific Bits		
Bit	Name	Definition
14	Motor activity	0: Motor stands still. 1: Motor rotates.
15	Direction of rotation	This bit shows the direction of rotation.

Table 117: Trinamic Specific Bits

Operation Mode specific Bits in pv Mode		
Bit	Name	Definition
10	Target reached	Indicates that the target speed has been reached.
12	Speed	Not supported.
13	Max. slippage error	Not supported.

Table 118: Operation Mode specific Bits in pv Mode

State Coding	
Status word	FSA state
xxxx xxxx x0xx 0000_h	Not ready to switch on
xxxx xxxx x1xx 0000_h	Switch on disabled
xxxx xxxx x01x 0001_h	Ready to switch on
xxxx xxxx x01x 0011_h	Switched on
xxxx xxxx x01x 0111_h	Operation enabled
xxxx xxxx x00x 0111_h	Quick stop active
xxxx xxxx x0xx 1111_h	Fault reaction active
xxxx xxxx x0xx 1000_h	Fault

Table 119: State Coding

Object Description			
Index	Name	Object Type	Data Type
6041_h	Controlword	Variable	UNSIGNED16

Table 120: Object Description (6041_h in pv Mode)

Entry Description				
Sub-index	Access	PDO Mapping	Value Range	Default Value
0	rw	see CiA402-3	See state coding above	

Table 121: Entry Description (6041_h in pv Mode)

8.1.3 Object $606B_h$: Velocity Demand Value

This objects provides the output value of the trajectory generator.

Object Description			
Index	Name	Object Type	Data Type
$606B_h$	Velocity Demand Value	Variable	SIGNED32

Table 122: Object Description ($606B_h$)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	Velocity Demand Value	yes	-2147483648	2147483647	0		R

Table 123: Entry Description ($606B_h$)

8.1.4 Object $606C_h$: Velocity Actual Value

This object shows the actual velocity value derived from the velocity sensor.

Object Description			
Index	Name	Object Type	Data Type
$606C_h$	Velocity Actual Value	Variable	SIGNED32

Table 124: Object Description ($606C_h$)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	Velocity Actual	no	-2147483648	2147483647	0	[rpm]	R

Table 125: Entry Description ($606C_h$)

8.1.5 Object 6083_h: Profile Acceleration

This object indicates the configured acceleration. Object 6083h sets the maximum acceleration to be used in profile positioning mode, and profile velocity mode.

Object Description			
Index	Name	Object Type	Data Type
6083_h	Profile Acceleration	Variable	UNSIGNED32

Table 126: Object Description (6083_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	Profile Acceleration	no	0	100000	2000	[rpm/s]	RW

Table 127: Entry Description (6083_h)

8.1.6 Object $60FF_h$: Target Velocity

This object indicates the configured target velocity and is used as input for the trajectory generator. Object 60FFh 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 6083h and 6084h.

Object Description			
Index	Name	Object Type	Data Type
$60FF_h$	Target Velocity	Variable	SIGNED32

Table 128: Object Description ($60FF_h$)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	Target Velocity	yes	-200000	200000	0	[rpm]	RW

Table 129: Entry Description ($60FF_h$)

8.2 How to move a Motor in pv Mode

Here is a little example that shows how to get a motor running in pv mode. In this little example we assume that the module has been reset (and then switched to pre-operational or operational) by NMT commands before.

- Select pv mode by writing 3 to object 6060_h (Modes_of_Operation).
- Write 6 to object 6040_h (Controlword) to switch to READY_TO_SWITCH_ON state.
- Write 7 to object 6040_h to switch to SWITCHED_ON state.
- Write the desired target velocity (e.g. 2000) to object $60FF_h$ (Target_Velocity).
- Write 15 to object 6040_h to switch to OPERATION_ENABLED state. The motor now accelerates to the target velocity.
- Stop the motor by writing 0 to object $60FF_h$.

9 Cyclic synchronous Torque Mode

The cyclic synchronous torque mode is used to directly control the torque of the motor, without the need for position or velocity control. It contains limit functions, but not a trajectory generator.

The cyclic synchronous torque mode covers the following sub-functions:

- Demand value input directly via an object.
- Monitoring and limiting the torque.

9.1 Detailed Object Specifications

9.1.1 Object 6040_h: Control Word

This object indicates the received command controlling the power drive system finite state automaton (PDS FSA). The CiA-402 state machine can be controlled using this object. Please refer to figure 4 for detailed information. The cyclic synchronous torque mode does not use any mode specific bits of the control word.

Structure of the Control Word									
15	9	8	7	6	4	3	2	1	0
nu		h	fr	nu		eo	qs	ev	so
MSB									LSB

Legend: nu=not used; h=halt; fr=fault reset; eo=enable operation; qs=quick stop; ev=enable voltage; so=switch on.

Table 130: Structure of the Control Word in cst Mode

Command Coding						
Command	Bits of Control Word					Transitions
	Bit 7	Bit 3	Bit 2	Bit 1	Bit 0	
Shutdown	0	x	1	1	0	2,6,8
Switch on	0	0	1	1	1	3
Switch on & enable operation	0	1	1	1	1	3, 4
Disable voltage	0	x	x	0	x	7,9,10,12
Quick stop	0	x	0	1	x	7,10,11
Disable operation	0	0	1	1	1	5
Enable operation	0	1	1	1	1	4, 16
Fault reset	0-to-1	x	x	x	x	15

Table 131: Command Coding

Object Description			
Index	Name	Object Type	Data Type
6040_h	Controlword	Variable	UNSIGNED16

Table 132: Object Description (6040_h in cst Mode)

Entry Description				
Sub-index	Access	PDO Mapping	Value Range	Default Value
0	rw	see CiA402-3	See command coding above.	

Table 133: Entry Description (6040_h in cst Mode)

9.1.2 Object 6041_h: Status Word

This object provides the status of the PDS FSA. It reflects the status of the CiA-402 state machine. Please refer to figure 4 for detailed information. The object is structured as defined below.

For more information about the coding please refer to the CANopen Drives and motion control device profile, part 2.

Structure of the Status Word															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
dir	mot	oms		ila	r	rm	ms	w	sod	qs	ve	f	oe	so	rtso
MSB															LSB

Legend: nu=not used; r=reserved; oms=operation mode specific; h=halt; fr=fault reset; oe=operation enable; qs=quick stop; ve=voltage enable; so=switch on.

Table 134: Structure of the Status Word in cst Mode

Trinamic Specific Bits		
Bit	Name	Definition
14	Motor activity	0: Motor stands still. 1: Motor rotates.
15	Direction of rotation	This bit shows the direction of rotation.

Table 135: Trinamic Specific Bits

Operation Mode specific Bits in cst Mode		
Bit	Name	Definition
10	Reserved	Not used.
12	Target torque ignored	0: Target torque ignored. 1: Target torque used as input to control loop.
13	Reserved	Not used.

Table 136: Operation Mode specific Bits in cst Mode

State Coding	
Status word	FSA state
xxxx xxxx x0xx 0000_h	Not ready to switch on
xxxx xxxx x1xx 0000_h	Switch on disabled
xxxx xxxx x01x 0001_h	Ready to switch on
xxxx xxxx x01x 0011_h	Switched on
xxxx xxxx x01x 0111_h	Operation enabled
xxxx xxxx x00x 0111_h	Quick stop active
xxxx xxxx x0xx 1111_h	Fault reaction active
xxxx xxxx x0xx 1000_h	Fault

Table 137: State Coding

Object Description			
Index	Name	Object Type	Data Type
6041_h	Controlword	Variable	UNSIGNED16

Table 138: Object Description (6041_h in cst Mode)

Entry Description				
Sub-index	Access	PDO Mapping	Value Range	Default Value
0	rw	see CiA402-3	See state coding above	

Table 139: Entry Description (6041_h in cst Mode)

9.1.3 Object 6071_h: Target Torque

This object gives the target motor current.

Object Description			
Index	Name	Object Type	Data Type
6071_h	Target Torque	Variable	SIGNED32

Table 140: Object Description (6071_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	Target Torque	yes	-90000	90000	0	[mA]	RW

Table 141: Entry Description (6071_h)

9.1.4 Object 6072_h: Max Torque

Max. allowed absolute motor current. *This value can be temporarily exceeded marginal due to the operation of the current regulator.

Object Description			
Index	Name	Object Type	Data Type
6072_h	Max Torque	Variable	UNSIGNED32

Table 142: Object Description (6072_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	Max Torque	no	0	90000	4000	[mA] (peak)	RW

Table 143: Entry Description (6072_h)

9.1.5 Object 6074_h: Torque Demand Value

This object gives the internal target motor current.

Object Description			
Index	Name	Object Type	Data Type
6074_h	Torque Demand Value	Variable	SIGNED32

Table 144: Object Description (6074_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	Torque Demand Value	yes	-90000	90000	0	[mA]	R

Table 145: Entry Description (6074_h)

9.1.6 Object 6077_h: Torque Actual Value

The actual motor current.

Object Description			
Index	Name	Object Type	Data Type
6077_h	Torque Actual Value	Variable	SIGNED32

Table 146: Object Description (6077_h)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	Torque Actual Value	yes	-2147483648	2147483647	0		R

Table 147: Entry Description (6077_h)

9.1.7 Object $60B2_h$: Torque offset

The actual set torque offset.

Object Description			
Index	Name	Object Type	Data Type
$60B2_h$	Torque offset	Variable	SIGNED32

Table 148: Object Description ($60B2_h$)

Entry Description							
Sub-index	Name	PDO Mapping	Min	Max	Default	Unit	Access
0	Torque offset	no	-2147483648	2147483647	0		RW

Table 149: Entry Description ($60B2_h$)

9.2 How to move a Motor in cst Mode

Here is a little example that shows how to get a motor running in cst mode. In this little example we assume that the module has been reset (and then switched to start) by NMT commands before.

- Select cst mode by writing 10 to object 6060_h (Modes_of_Operation).
- Write 6 to object 6040_h (Controlword) to switch to READY_TO_SWITCH_ON state.
- Write 7 to object 6040_h to switch to SWITCHED_ON state.
- Write the desired torque (e.g. 1000) to object 6071_h (Target_Torque).
- Write 15 to object 6040_h to switch to OPERATION_ENABLED state to start the motor.
- To stop the motor, write 0 to object 6071_h.

10 Supplemental Directives

10.1 Producer Information

10.2 Copyright

10.3 Trademark Designations and Symbols

Trademark designations and symbols used in this documentation indicate that a product or feature is owned and registered as trademark and/or patent either by TRINAMIC or by other manufacturers, whose products are used or referred to in combination with TRINAMIC's products and TRINAMIC's product documentation.

This Datasheet is a non-commercial publication that seeks to provide concise scientific and technical user information to the target user. Thus, trademark designations and symbols are only entered in the Short Spec of this document that introduces the product at a quick glance. The trademark designation /symbol is also entered when the product or feature name occurs for the first time in the document. All trademarks and brand names used are property of their respective owners.

10.4 Target User

The documentation provided here, is for programmers and engineers only, who are equipped with the necessary skills and have been trained to work with this type of product.

The Target User knows how to responsibly make use of this product without causing harm to himself or others, and without causing damage to systems or devices, in which the user incorporates the product.

10.5 Disclaimer: Life Support Systems

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.

Information given in this document is believed to be accurate and reliable. However, no responsibility is assumed for the consequences of its use nor for any infringement of patents or other rights of third parties which may result from its use. Specifications are subject to change without notice.

10.6 Disclaimer: Intended Use

The data specified in this user manual is intended solely for the purpose of product description. No representations or warranties, either express or implied, of merchantability, fitness for a particular purpose

or of any other nature are made hereunder with respect to information/specification or the products to which information refers and no guarantee with respect to compliance to the intended use is given.

In particular, this also applies to the stated possible applications or areas of applications of the product. TRINAMIC products are not designed for and must not be used in connection with any applications where the failure of such products would reasonably be expected to result in significant personal injury or death (safety-Critical Applications) without TRINAMIC's specific written consent.

TRINAMIC products are not designed nor intended for use in military or aerospace applications or environments or in automotive applications unless specifically designated for such use by TRINAMIC. TRINAMIC conveys no patent, copyright, mask work right or other trade mark right to this product. TRINAMIC assumes no liability for any patent and/or other trade mark rights of a third party resulting from processing or handling of the product and/or any other use of the product.

10.7 Collateral Documents & Tools

This product documentation is related and/or associated with additional tool kits, firmware and other items, as provided on the product page at: www.trinamic.com.

11 Figures Index

12 Tables Index

13 Revision History

13.1 IC Revision

Version	Date	Author	Description
V1.00	2016-NOV-21	LL/SL	First IC version.

Table 150: IC Revision

13.2 Firmware Revision

Version	Date	Author	Description
V1.00	2016-NOV-21	ED	First CoE version.

Table 151: Firmware Revision

13.3 Document Revision

Version	Date	Author	Description
V1.0	2016-NOV-21	ED	Initial version

Table 152: Document Revision