ECAT-2094S

EtherCAT
4-Axis Stepper Motor
Controller/ Driver

User Manual

(Version 1.3.4)





WARRANTY

All products manufactured by ICP DAS are warranted against defective materials for a period of one year from the date of delivery to the original purchaser.

WARNING

ICP DAS assumes no liability for damages consequent to the use of this product. ICP DAS reserves the right to change this manual at any time without notice. The information furnished by ICP DAS is believed to be accurate and reliable. However, no responsibility is assumed by ICP DAS for its use, nor for any infringements of patents or other rights of third parties resulting from its use.

COPYRIGHT

Copyright © 2017 by ICP DAS. All rights are reserved.

TRADEMARK

Names are used for identification only and may be registered trademarks of their respective companies.

CONTACT US

If you have any questions, please feel free to contact us via email at:
service@icpdas.com
service.icpdas@gmail.com

Revision

Revision	Date	Description	Author
1.00	05.09.2018	Initial version	M.K.
1.0.1	13.05.2020	Updated specification	M.K.
1.0.2	19.05.2020	Modify table 5 (power supply for motor Z and U), Scope of delivery	M.K.
1.2.0	16.09.2020	• Acceleration/deceleration unit and type • Error Table • Vendor Specific Register • Modified motor voltage range	M.K.
1.2.1	25.01.2021	• Updated the "Open collector wiring diagram" • Acceleration/deceleration unit: ▪ Added "current to target" ▪ Update V-T graph • Updated Error Table • Added "Target overrun" • Added dynamic change of the relative position example	M.K.
1.2.2	20.07.2021	• Firmware version 1.6 • Added following objects: ▪ Target overrun ▪ Initialization error	M.K
1.3.0	25.05.2022	• Hardware modified, added following features: ▪ Alias rotary switch ▪ Firmware updare via FoE • Firmware version 2.0 • Added " Station Alias "	Eric Chen
1.3.1	10.03.2023	• Modify the PGND label of the connection interface	Eric Chen
1.3.2	08.06.2023	• Modify the description of “Save Configuration Data to Memory”	Eric Chen
1.3.3	08.04.2024	• Added attention: Enable DI function to perform emergency stop and cut off motor power, instead of disconnecting VM for STO • Modify the description of function of digital input, added	Eric Chen

		emergency stop functions • Modify the default value of "Power on motor current" to 0	
1.3.4	29.10.2026	• Fix SDO object enumeration errors	Eric Chen

Contents

1 Product Overview 4
1.1 Introduction 4
1.2 Technical Data 5
1.3 Hardware Specification 6
1.4 Dimensions 7
2 Scope of Delivery 8
3 Wiring 9
3.1 LED Definition 9
3.2 Alias Rotary Swtich 11
3.3 Connection Interfaces 11
3.4 Digital Input and Output Wiring 14
3.5 Stepper Motor Wiring 17
3.5.1 Four Lead Motor 17
3.5.2 Eight Lead Motor 18
3.5.3 Encoder Connection 19
4 Basics Communication 21
4.1 EtherCAT Cabling 21
4.2 EtherCAT State Machine 21
4.3 Synchronization Modes 23
4.3.1 Free Run Mode 23
4.3.2 Distributed Clocks (DC Mode) 24
5 Project Integration 27
5.1 ESI File 27
5.1.1 Import of ESI File 27
5.2 Device Setup and Configuration 27
5.2.1 Scanning of the EtherCAT Device 28
5.2.2 EtherCAT Slave Process Data Settings 30
5.2.3 Basic Stepper Driver Configuration 31
5.3 Firmware Update 33
6 Position Control Setting 35
6.1 Positioning Interface Types 35
6.2 Positioning Interface 35
6.2.1 Acceleration/Deceleration Unit Definition 45
6.2.1.1 Acceleration time from Vmin to Vmax 46
6.2.1.2 Acceleration time from Vmin to Vtarget 47
6.2.1.3 Acceleration [128*µStep/sec^2] 49
6.2.1.4 Acceleration time from Vcurrent to Vtarget 50
6.2.2 Acceleration/Deceleration Type 51
6.2.2.1 Start-Stop Phase Type 52
6.2.2.2 Standard Acceleration/Deceleration 53
6.3 Positioning Interface Compact 54

6.4 Position Control 58
7 Velocity Control Setting 62
8 CoE Interface 65
8.1 General Description 65
8.2 Save Configuration Data to Memory 66
8.3 Driver Tuning 69
9 Object Description and Parameterization 70
9.1 Standard Objects 70
9.2 RxPDO Mapping Objects 71
9.3 TxPDO Mapping Objects 74
9.4 Sync Manager Objects 78
9.5 Input Data 84
9.6 Output Data 87
9.7 Configuration Data 91
9.8 Driver Tuning Functions 97
9.9 Information and Diagnostic Data 98
9.10 Configuration Parameters Storage 100
9.11 Station alias Configuration 101
10 Error Table 101
11 Vendor Specific Register Definitions 102

1 Product Overview



1.1 Introduction

The ECAT-2094S stepper motor controller is a cost-effective, two-phase bipolar stepper driver. The ECAT-2094S simultaneously controls up to four stepper motors. A motor voltage range between 6 and 46V DC and a maximum motor coil current of 1.5A/phase is being supported. The running motor current, microstep resolution and other motion parameters are software selectable.

Two-phase bipolar stepper motors can be directly connected to the ECAT-2094S device. The device is designed to operate in a open loop. Configuration and motion control has to be done by the EtherCAT master and the application program. The torque and step control is done by an internal stepper motor driver IC. Each stepper motor is being independently controlled by a separated driver IC. The four driver ICs are not synchronized and work independently from each other. An integrated ramp generator automatically calculates the acceleration and deceleration distance. The motion controller drives the motor to the target position or accelerates the motor to the target velocity. All motion parameters can be changed on the fly. A minimum set of configuration data consists of acceleration, deceleration and maximum motion velocity. After receiving the target position the motor driver starts controlling the motion movement.

The ECAT-2094s has four integrated incremental encoder interfaces. Four 32 bit high frequency encoder counter counts the input signal of external incremental encoders. The encoder can for example be used for homing purposes and for consistency checks.

High resolution of up to 256 microsteps per full step is supported for a ensuring smooth and precise motor operation.

For each motor two digital input channels are provided. The digital inputs can be set to act as a simple DI, as a left and right hardware limit switch which automatically stops the motor when activated, or a latch trigger for latching the current motor and encoder position.

The module must be supplied by three power sources. Two motor supply and a 24Vdc control supply. Two motors share one power supply.

1.2 Technical Data

- Supports 4 stepper motor (2-phase bipolar)
- Stepper motor are controlled in an open loop operation
- Programmable coil current level: up to 1.5 A/phase
- Programmable microstep size: maximum 256 microsteps per full step
- Supported motor voltage range: 6 to 46V_{DC}
- 4 x Encoder interfaces (A, B, Z), differential
- 8 x Digital input. Two DI channels for each axis: reference switch input, latch input
- 2 x Digital output
- Automatic current reduction to reduce heat when motor is not moving
- Drive protection:
  - Over-temperature
  - Under voltage
  - Short circuit
- Optically isolated I/O
- LED indicators for I/O, EtherCAT and motion status
- Internal memory for storing configuration data
- EtherCAT:
  - 2 x RJ-45 bus interface
  - Distance between stations up to 100 m (100BASE-TX)
  - Support daisy chain connection
  - EtherCAT conformance test tool verified
  - Supports Free-Run, SM-Synchron and Distributed Clock (DC) operation modes
  - Supports CoE and FoE
- Removable terminal block connector
- Two 16-position rotary switch for station alias addressing

1.3 Hardware Specification

Item	Specification
Motor Outputs	
Number of outputs	4x stepper motor, 2 phases
Output current	1.5A/phase
Motor voltage range	6 to 46V_{DC}
Current controller frequency	24.5 kHz
Maximum step frequency	8.388 MHz
Microsteps per step	256, 128, 64, 32, 16, 8, 4, 2
Encoder inputs	
Number of encoder inputs	4x encoder counter (A, B, Z), differential
Maximum encoder pulse frequency	4 MHz
Digital Inputs	
Number of digital inputs	8 (2x limit position for each motor)
Wet contact	• ON voltage level: +10 to 30V_{DC} • OFF voltage level: +5V_{DC} MAX
Photo-Isolation	3750V_{DC}
Digital Output	
Number of digital outputs	2
Output type	Open collector
Load voltage	+5 to 30V_{DC}
Max. load current	100mA
Isolation voltage	3750 V_{DC}
LED Indicators	
Diagnostic LED	Power, EtherCAT status, Digital IO, driving, temperature warning, over-temperature error, phase A and B under-voltage
Communication Interface	
Connector	2 x RJ-45
Protocol	EtherCAT
Distance between stations	Max. 100 m (100BASE-TX)
Data transfer medium	Ethernet/EtherCAT Cable (Min. CAT 5), Shielded
Power	
Input voltage range	20V ~ 30V_{DC}
EMS Protection	
ESD (IEC 61000-4-2)	4 KV Contact for Each Channel
EFT (IEC 61000-4-4)	Signal: 1 KV Class A; Power: 1 KV Class A
Surge (IEC 61000-4-5)	1 KV Class A
Mechanism	
Installation	DIN-Rail
Dimensions (LxWxH) [mm]	181 x 123.5 x 37 (without connectors)
Case material	Metal
Environment	
Operating temperature	-25℃ ~ 40℃
Storage temperature	-30℃ ~ 80℃
Relative humidity	10 ~ 90%, No Condensation

Table 1: Technical data

1.4 Dimensions

All dimension units are in millimeter.

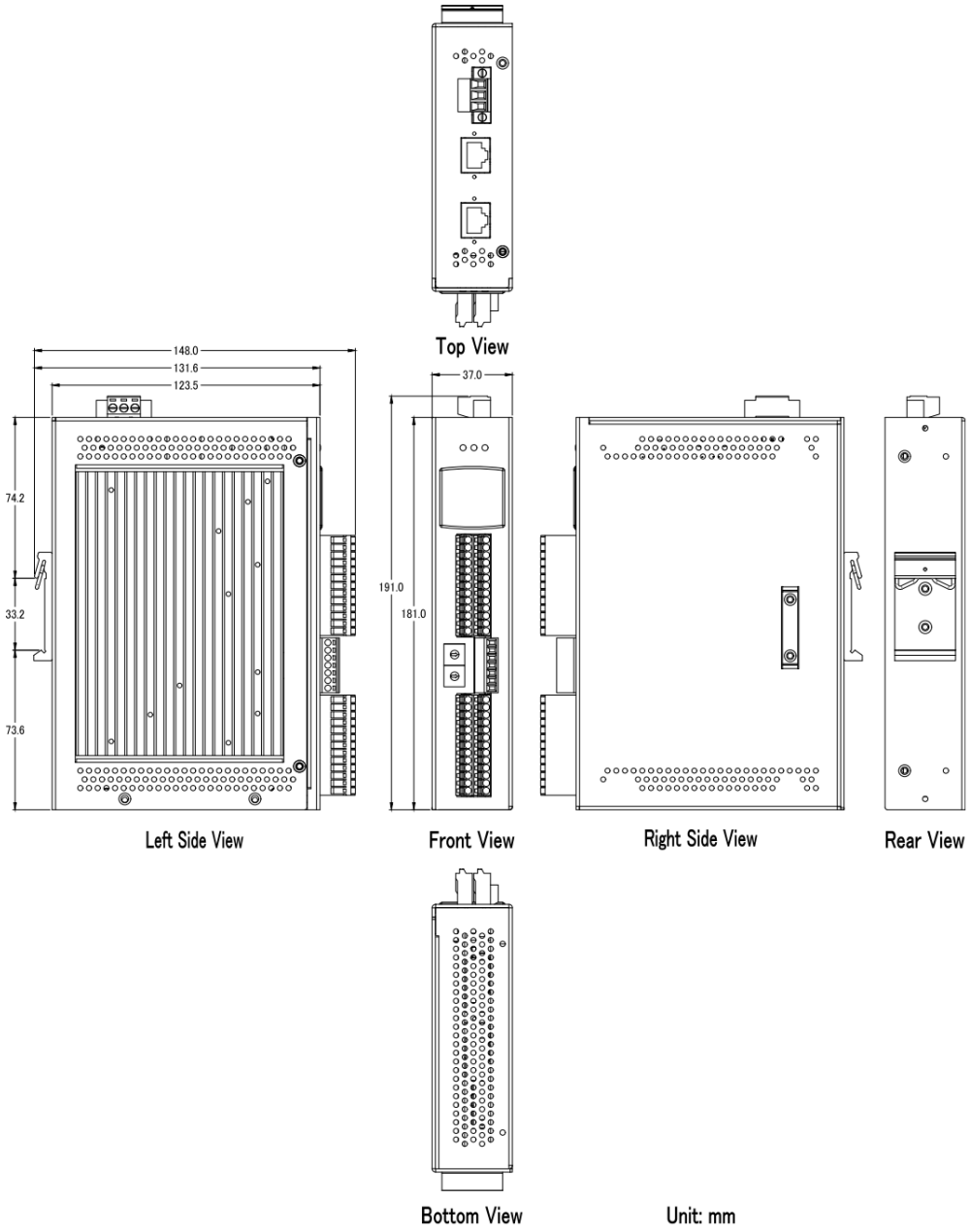


Figure 1: Dimension of the ECAT-2094S housing

2 Scope of Delivery

The shipping package includes the following items:

- 1 x ECAT-2094S
- 2 x Wall mount
- 4 x 13-pin plug-in connector
- 1 x 3-pin plug in connector (for power supply)
- 1 x 7-pin plug in connector (for motor power supply)
- 1 x Quick Start manual



Figure 2: ECAT-2094S module and Quick Start manual

Note:

If any of these items are missing or damaged, please contact your local distributor. Please keep the original retail box with all retail packaging (Styrofoam, inner boxes, fasteners, etc.) in case you need to return the product.

More information about the ECAT-2094S is available on the product website. Enter the product name in the search function of the main ICPDAS website to access the product website:

Step 1: Go to the ICPDAS main website
http://www.icpdas.com

Step 2: Enter "ECAT-2094S" in the search editor to access the ECAT-2094S website



The User Manual, Quick-Start, EtherCAT Slave Information (ESI) file and FAQ can be downloaded from the website.

Technical support:
service@icpdas.com

3 Wiring

3.1 LED Definition

The ECAT-2094s provides on the frontside of the connection cap several diagnostic LEDs. Furthermore there are three LEDs to indicate the network status for EtherCAT. The exact meaning of the LED indication is specified in the following tables:

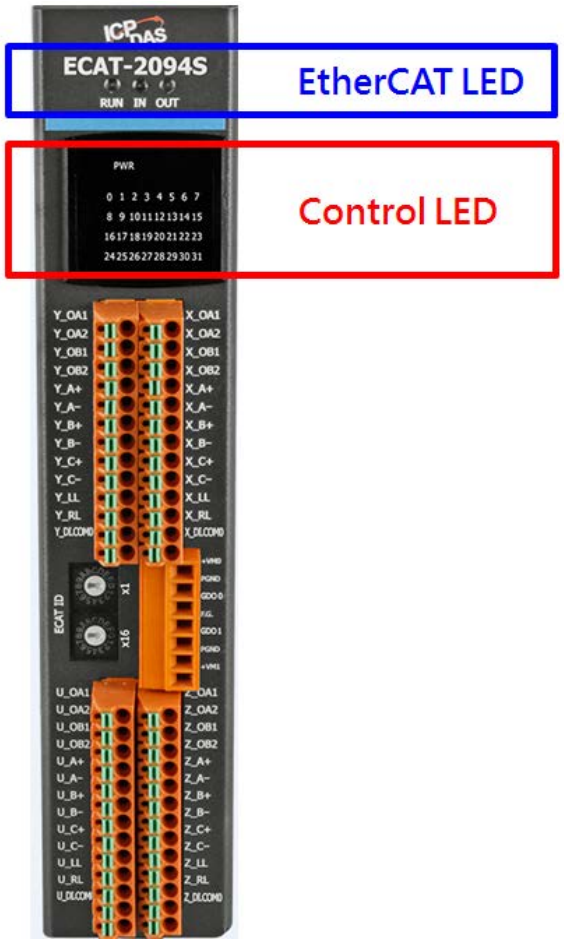


Figure 3: ECAT-2094S front and side view

EtherCAT LED	Color	State	Description
RUN	red		This LED indicates the operation state of the EtherCAT slave:
		Off	Device is in INIT state
		Flashing	Device is in PREOP state
		Single flash	Device is in SAFEOP state Outputs remain in safe state
		On	Device is in OP state

EtherCAT LED	Color	State	Description
IN	green		Indicates the communication status of the EtherCAT port IN
		Off	No connection
		Flashing	Link and activity (e.g. data exchange with the master)
		On	Link without any activity
OUT	green		Indicates the communication status of the EtherCAT port OUT. Further EtherCAT slave can be connected to the port OUT
		Off	No EtherCAT slaves are connected to port OUT
		Flashing	Link and activity (e.g. data exchange connected slaves)
		On	Link without any activity

Table 2: EtherCAT status indicator

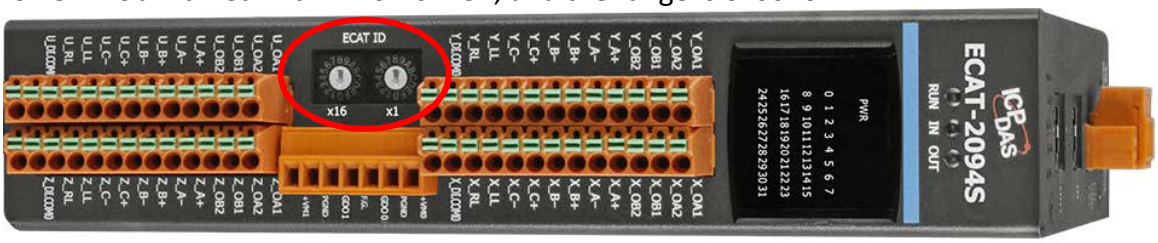
Control LED	Color	Description
*	red	- Power indicator
* * * * * * * * (first row) 0 1 2 3 4 5 6 7	green	- LED 0: Digital input channel 1 (X_LL) - Axis X - LED 1: Digital input channel 2 (X_RL) - Axis X - LED 2: Digital input channel 1 (Y_LL) - Axis Y - LED 3: Digital input channel 2 (Y_RL) - Axis Y - LED 4: Digital input channel 1 (Z_LL) - Axis Z - LED 5: Digital input channel 2 (Z_RL) - Axis Z - LED 6: Digital input channel 1 (U_LL) - Axis U - LED 7: Digital input channel 2 (U_RL) - Axis U
* * * * * * * * (second row) 8 9 10 11 12 13 14 15	green	- LED 8: Digital output (DO 0) - Axis X - LED 9: Digital output (DO 1) - Axis Y - LED 10: EEPROM access error - LED 11: reserved - LED 12: Driving output - Axis X - LED 13: Driving output - Axis Y - LED 14: Driving output - Axis Z - LED 15: Driving output - Axis U
* * * * * * * * (third row) 16 17 18 19 20 21 22 23	green	- LED 16: Motion error - Axis X - LED 17: Motion error - Axis Y - LED 18: Motion error - Axis Z - LED 19: Motion error - Axis U - LED 20: Over temperature error - Axis X - LED 21: Over temperature error - Axis Y - LED 22: Over temperature error - Axis Z - LED 23: Over temperature error - Axis U
* * * * * * * * (fourth row) 24 25 26 27 28 29 30 31	green	- LED 24: Short to ground error - Axis X - LED 25: Short to ground error - Axis Y - LED 26: Short to ground error - Axis Z - LED 27: Short to ground error - Axis U - LED 28: Over temperature/open load warning - Axis X

Control LED	Color	Description
		- LED 29: Over temperature/open load warning - Axis Y - LED 30: Over temperature/open load warning - Axis Z - LED 31: Over temperature/open load warning - Axis U

Table 3: Diagnostic LEDs

3.2 Alias Rotary Swtich

The Alias knob is divided into two part, the upper knob marked with “x1”is LSB, the lower knob marked with “x16” is MSB, and the range is 0x00~0xFF.



3.3 Connection Interfaces



F.G
GND
+Vs

IN
OUT

Figure 4: ECAT-2094S side view with power supply and EtherCAT connection

Name	Signal	Description
F.G	Frame ground	
GND	Power supply: Ground 0V (from negative power contact)	Feeding for ECAT-2094S
+Vs	Power supply: +24 V_{DC} (from positive power contact)	Feeding for ECAT-2094S
IN	EtherCAT signal input	Incoming EtherCAT cable
OUT	EtherCAT signal output	Outgoing EtherCAT cable

Table 4: ECAT-2094S power supply and EtherCAT interfaces

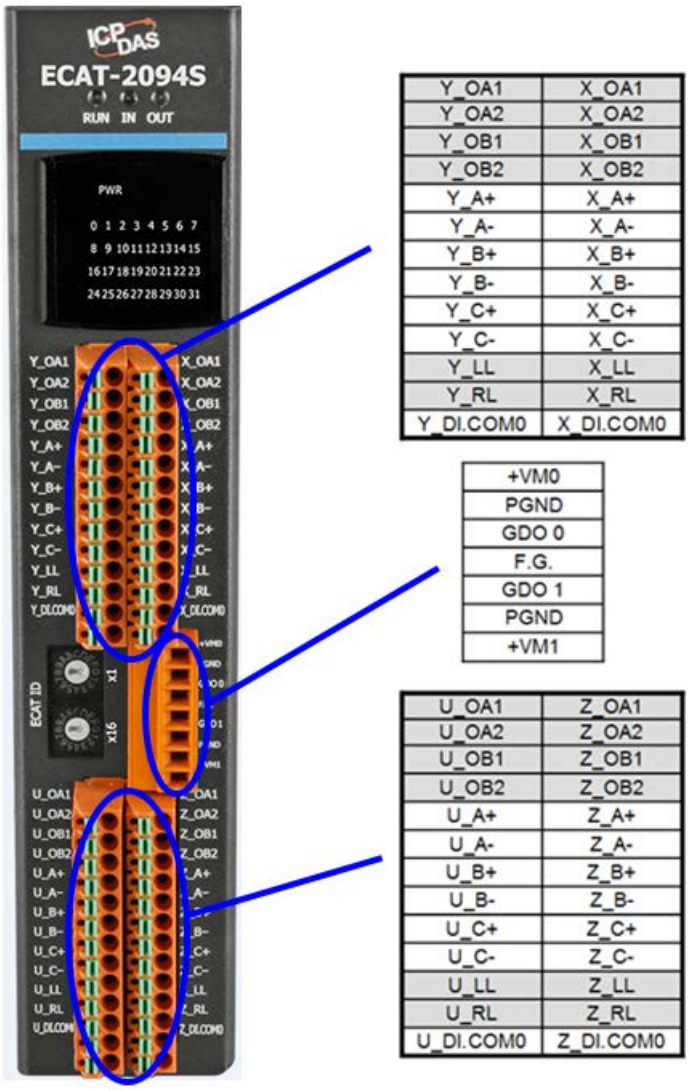


Figure 5: ECAT-2094S front view with motor and encoder in- and outputs

Name	Signal	Signal Description
+VM0	+6 to 46V_{DC} (from positive power contact)	Power supply for motor X and Y Power supply for motor
PGND	Ground 0V (from negative power contact)	 CAUTION: Automatic start of stepper motor! • Risk of death or serious injury for humans working in the machine. It can not ruled out that the stepper motor may perform unplanned movement during the ECAT-2094S setup and configuration
GDO 0	Output	General purpose digital output channel 0
F.G.		Frame ground
GDO 1	Output	General purpose digital output channel 1
PGND	Ground 0V (from negative power contact)	Power supply for motor Z and U Power supply for motor
+VM1	+6 to 46V_{DC} (from positive power contact)	 CAUTION: Automatic start of stepper motor! • Risk of death or serious injury for humans working in the machine. It can not ruled out that the stepper motor may perform unplanned movement during the ECAT-2094S setup and configuration

Table 5: Motor power supply and general purposes DO interfaces

Attention:
Since ECAT-2094S uses an integrated chip, it is not allowed to be powered off during use, so it is not recommended to disconnect the motor voltage (VM) supply for STO (Safe Torque off).
ECAT-2094S provides another solution: the user can enable the DI function to perform an emergency stop and turn off the motor when DI is triggered (see the "Function for input 1" (0x8n12:32) and "Function for input 2" (0x8n12:36) described in Chapter 9.7).

Name	Signal	Signal Description	
X_OA1	Output	Motor X winding A1	Motor X
X_OA2	Output	Motor X winding A2	
X_OB1	Output	Motor X winding B1	
X_OB2	Output	Motor X winding B2	
X_A+	Input	Encoder X input A+	Encoder X
X_A-	Input	Encoder X input A-	
X_B+	Input	Encoder X input B+	
X_B-	Input	Encoder X input B-	
X_C+	Input	Encoder X input C+	
X_C-	Input	Encoder X input C-	
X_LL	Input	Left limit switch for motor X	DI, limit switch or latch trigger for motor X
X_RL	Input	Right limit switch for motor X	
X_DI.COM0		Common DI X supply: +10 to +24V_{DC}	For X_LL and X_RL

Name	Signal	Signal Description	
Y_OA1	Output	Motor Y winding A1	Motor Y
Y_OA2	Output	Motor Y winding A2	
Y_OB1	Output	Motor Y winding B1	
Y_OB2	Output	Motor Y winding B2	
Y_A+	Input	Encoder Y input A+	Encoder Y
Y_A-	Input	Encoder Y input A-	
Y_B+	Input	Encoder Y input B+	
Y_B-	Input	Encoder Y input B-	
Y_C+	Input	Encoder Y input C+	
Y_C-	Input	Encoder Y input C-	
Y_LL	Input	Left limit switch for motor Y	DI, limit switch or latch trigger for motor Y
Y_RL	Input	Right limit switch for motor Y	
Y_DI.COM0		Common DI Y supply: +10 to +24V_{DC}	For Y_LL and Y_RL
Z_OA1	Output	Motor Z winding A1	Motor Z
Z_OA2	Output	Motor Z winding A2	
Z_OB1	Output	Motor Z winding B1	
Z_OB2	Output	Motor Z winding B2	
Z_A+	Input	Encoder Z input A+	Encoder Z
Z_A-	Input	Encoder Z input A-	
Z_B+	Input	Encoder Z input B+	
Z_B-	Input	Encoder Z input B-	
Z_C+	Input	Encoder Z input C+	
Z_C-	Input	Encoder Z input C-	
Z_LL	Input	Left limit switch for motor Z	DI, limit switch or latch trigger for motor Z
Z_RL	Input	Right limit switch for motor Z	
Z_DI.COM0		Common DI Z supply: +10 to +24V_{DC}	For Z_LL and Z_RL
U_OA1	Output	Motor U winding A1	Motor U
U_OA2	Output	Motor U winding A2	
U_OB1	Output	Motor U winding B1	
U_OB2	Output	Motor U winding B2	
U_A+	Input	Encoder U input A+	Encoder U
U_A-	Input	Encoder U input A-	
U_B+	Input	Encoder U input B+	
U_B-	Input	Encoder U input B-	
U_C+	Input	Encoder U input C+	
U_C-	Input	Encoder U input C-	
U_LL	Input	Left limit switch for motor U	DI, limit switch or latch trigger for motor U
U_RL	Input	Right limit switch for motor U	
U_DI.COM0		Common DI U supply: +10 to +24V_{DC}	For U_LL and U_RL

Table 6: Connection interfaces for the motor current outputs, encoder and digital inputs

3.4 Digital Input and Output Wiring

Digital Input		
Digital input channels		8 (2x limit position for each motor)
Input type		Wet
Wet contact	ON voltage level	+10 to 30 V_{DC}
	OFF voltage level	+5 V_{DC} MAX
Photo-isolation		3750 V_{DC}
Digital Output		
Digital output channels		2
Output type		Open collector
Load voltage		+5 to 30 V_{DC}
Max. load current		100mA
Isolation voltage		3750 V_{DC}

Table 7: Digital input and output specifications

The diagram for right (RL) and left (LL) limit switch wiring for axis X, Y, Z, U is shown below (Figure 6).
The digital input RL and LL can be used as a simple DI, a positive and negative limit switch and a position latch trigger. The DI channels can be set to simultaneously act as a limit switch and a position latch input. Each axis (X, Y, Z, U) is equipped with a pair of RL and LL digital inputs.
Example: The RL, LL, DI.COM symbols in the picture below represents the X_RL, X_LL and X_DI.COM for the X-motor.

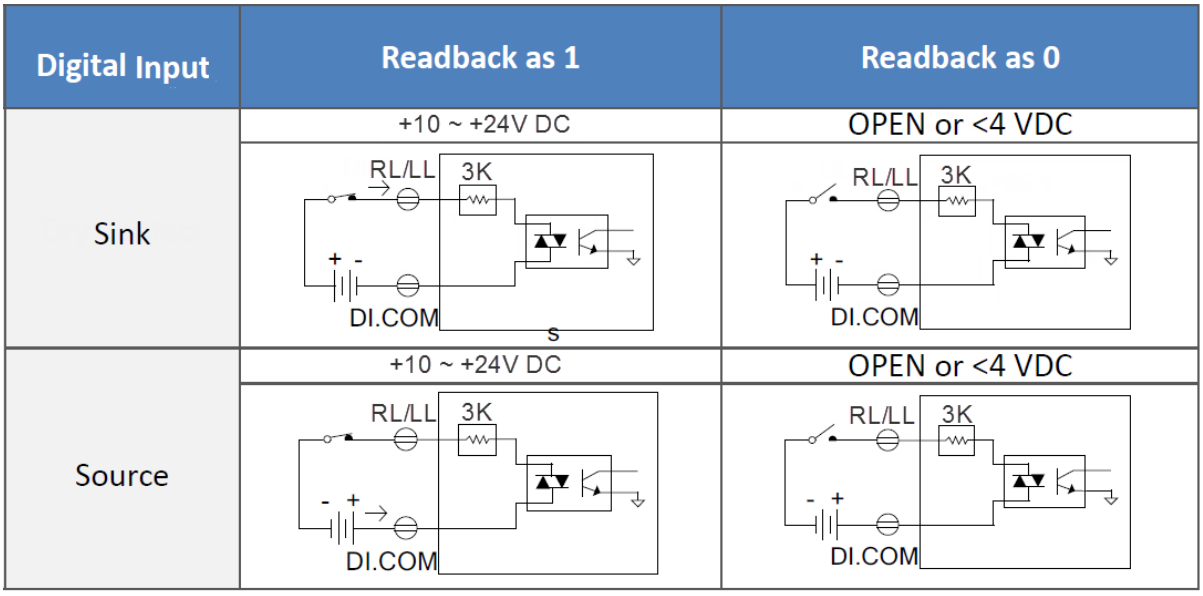


Figure 6: Digital inputs RL and LL for motor X, Y, Z, U

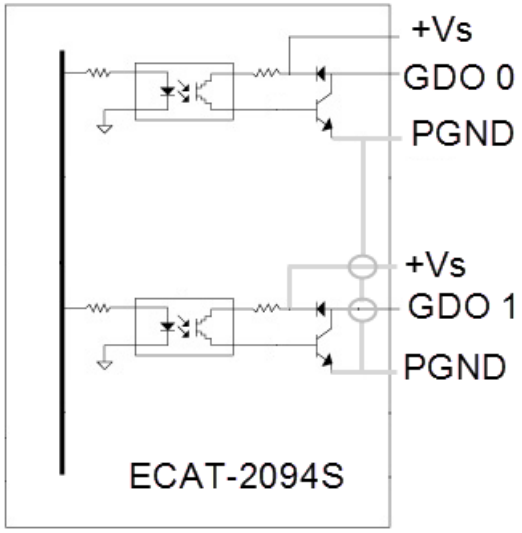


Figure 7: General purpose DO block diagram

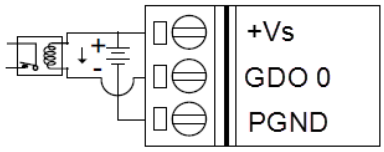
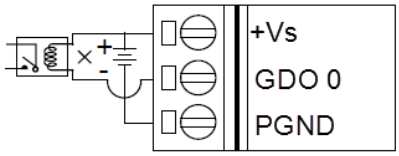
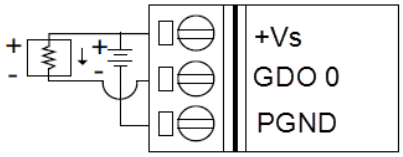
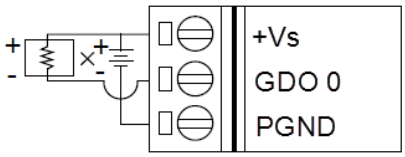
Output Type	ON State Readback as 1	OFF State Readback as 0
Driver Relay		
Resistance Load		

Figure 8: General purpose DO channel 0

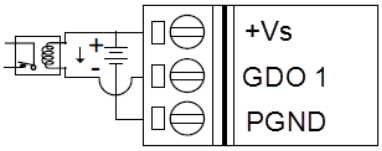
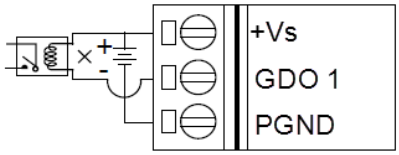
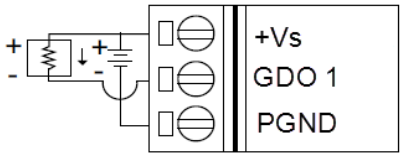
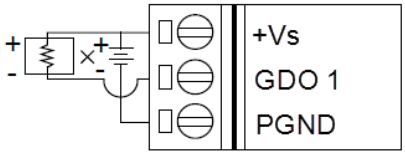
Output Type	ON State Readback as 1	OFF State Readback as 0
Driver Relay		
Resistance Load		

Figure 9: General purpose DO channel 1

3.5 Stepper Motor Wiring

3.5.1 Four Lead Motor

The Figure 10 below shows an example for a four lead two-phase motor connected to the X output of the ECAT-2094S.

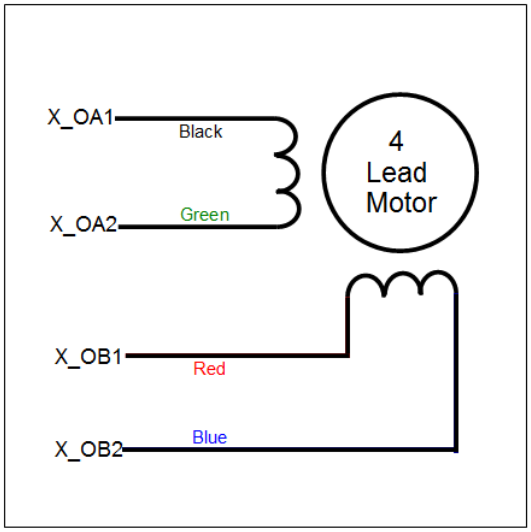


Figure 10: Four lead bipolar motor connected to the first axis output

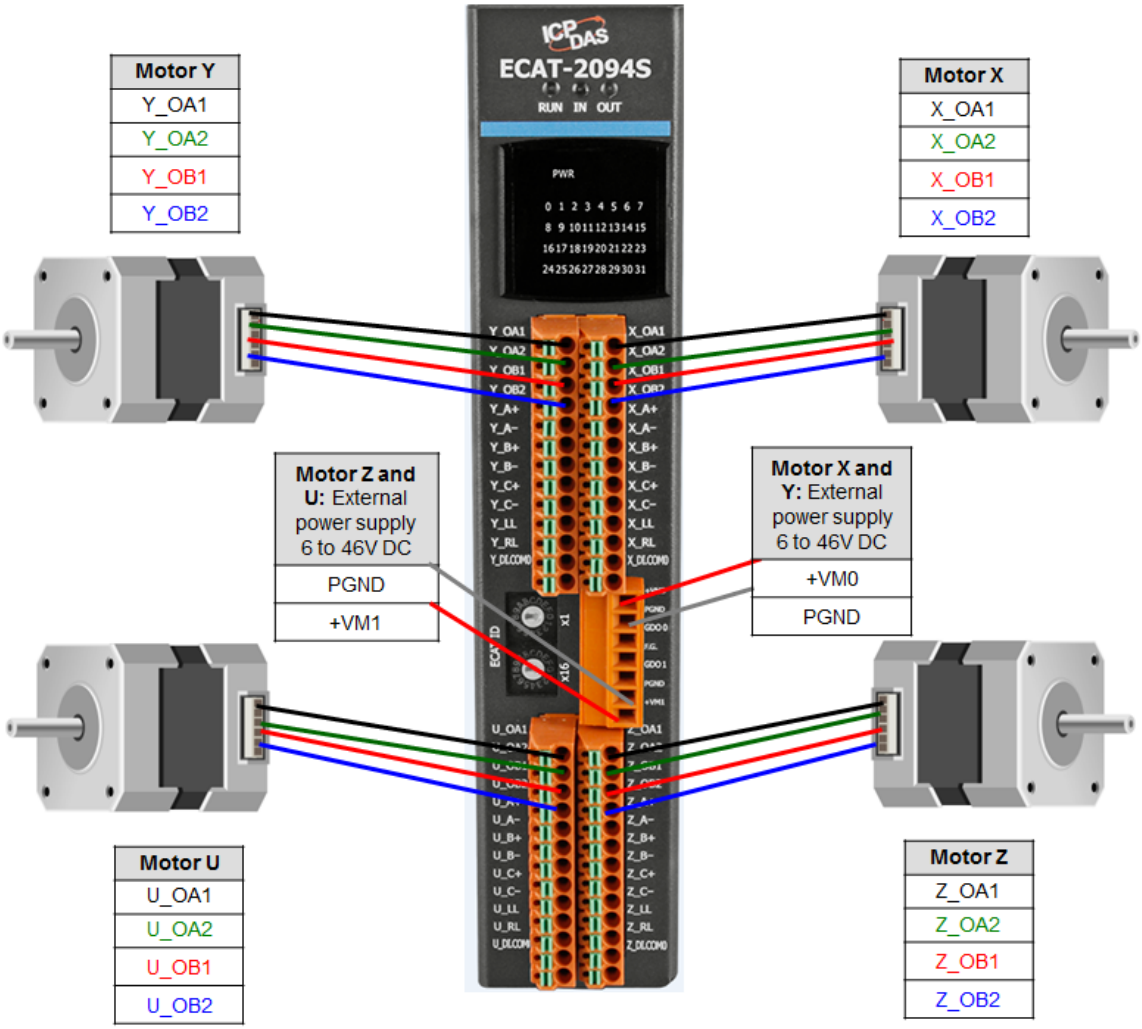


Figure 11: ECAT-2094S connected to four stepper motors

3.5.2 Eight Lead Motor

Eight lead motors can be connected in series or parallel. A series connected motor needs less current than one that is connected in parallel but it will not be able to run as fast.

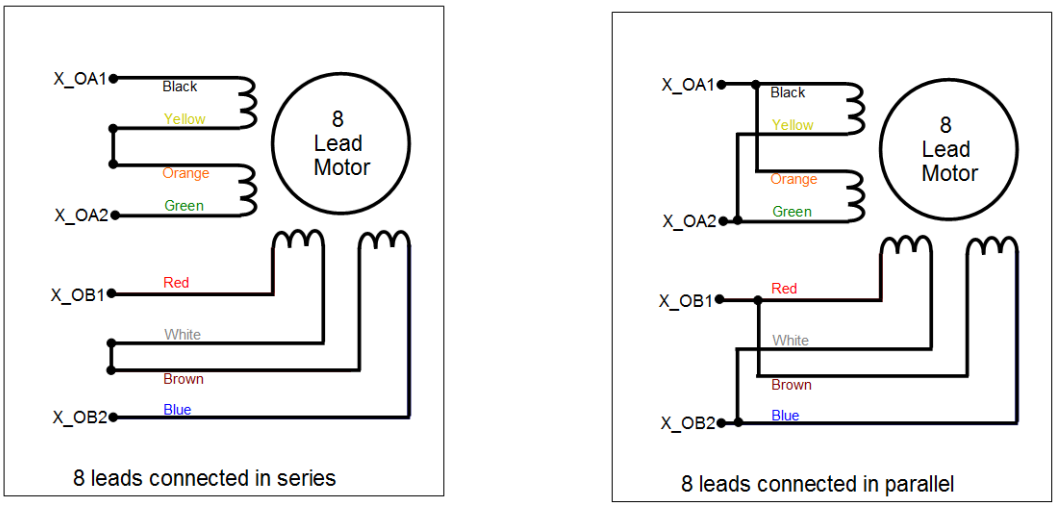


Figure 12: Eight lead bipolar motor connection (left: series, right: parallel)

3.5.3 Encoder Connection

Differential encoder:
The ECAT-2094S supports differential encoder by default.

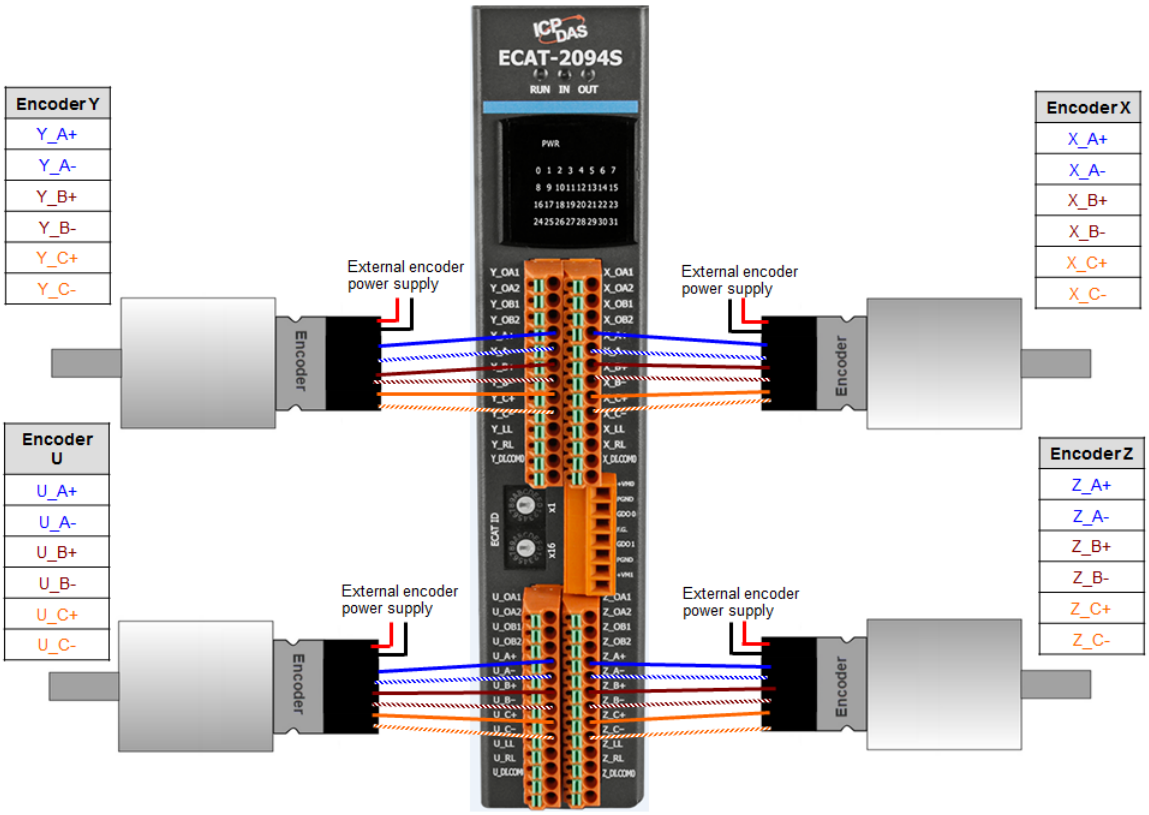


Figure 13: Connecting four differential encoder

Open collector type encoder:
For single-ended encoder connection refer to the Figure 14 which list the possible power supply values with the corresponding resistor sizes.

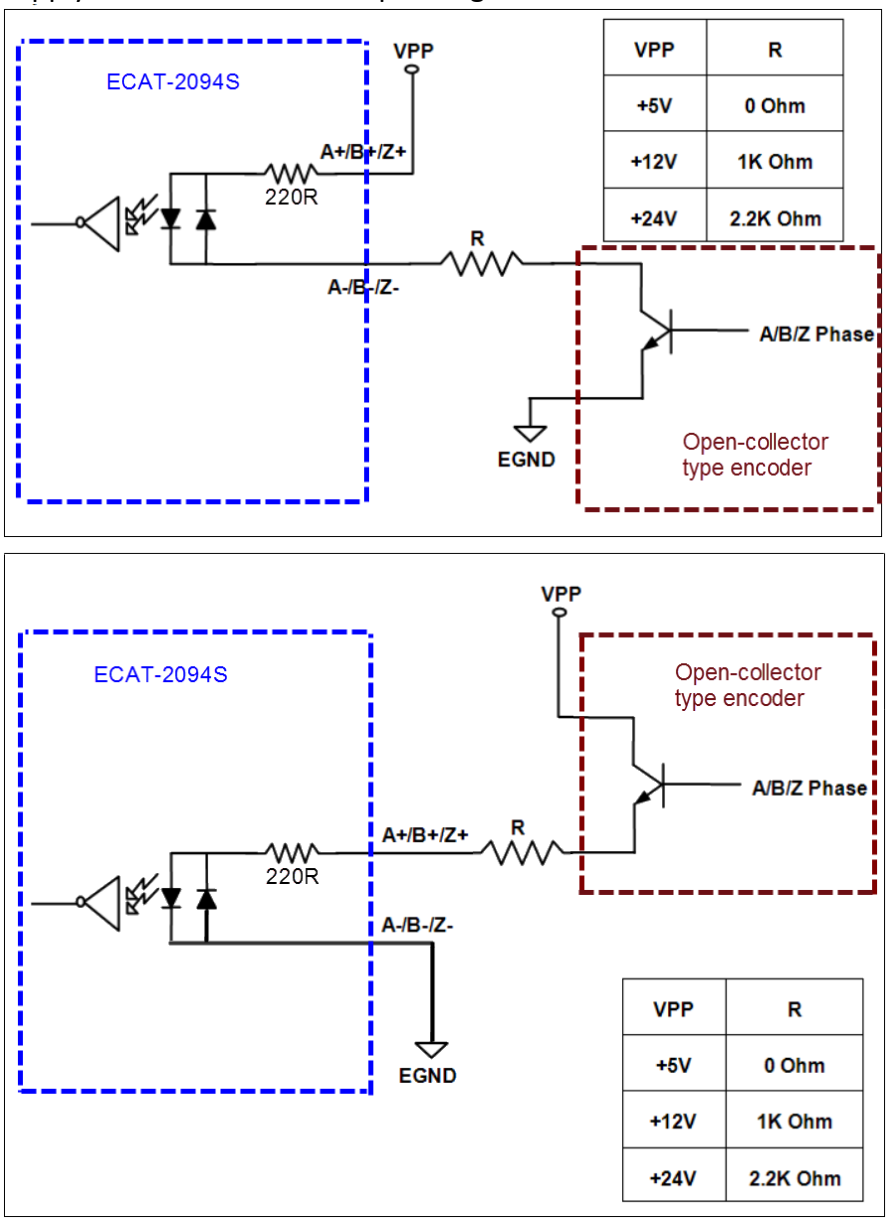


Figure 14: Open collector wiring diagram

4 Basics Communication

4.1 EtherCAT Cabling

The cable length between two EtherCAT devices must not exceed 100 m.

Cables and connectors
For connecting EtherCAT devices only Ethernet connections (cables + plugs) that meet the requirements of at least category 5 (CAT5) according to EN 50173 or ISO/IEC 11801 should be used. EtherCAT uses 4 wires for signal transfer.
The pin assignment is compatible with the Ethernet standard (ISO/IEC 8802-3).

4.2 EtherCAT State Machine

The state of the EtherCAT master and slave is controlled via the EtherCAT State Machine (ESM) . The state determines which functions are accessible or executable in the EtherCAT slave. State changes are typically initiated by requests of the master and acknowledged by the slave after the successful initialization. In case of an internal error, the slave automatically changes to a lower state.

The ECAT-2094S supports four states:
- Init (state after Reset)
- Pre-Operational
- Safe-Operational
- Operational

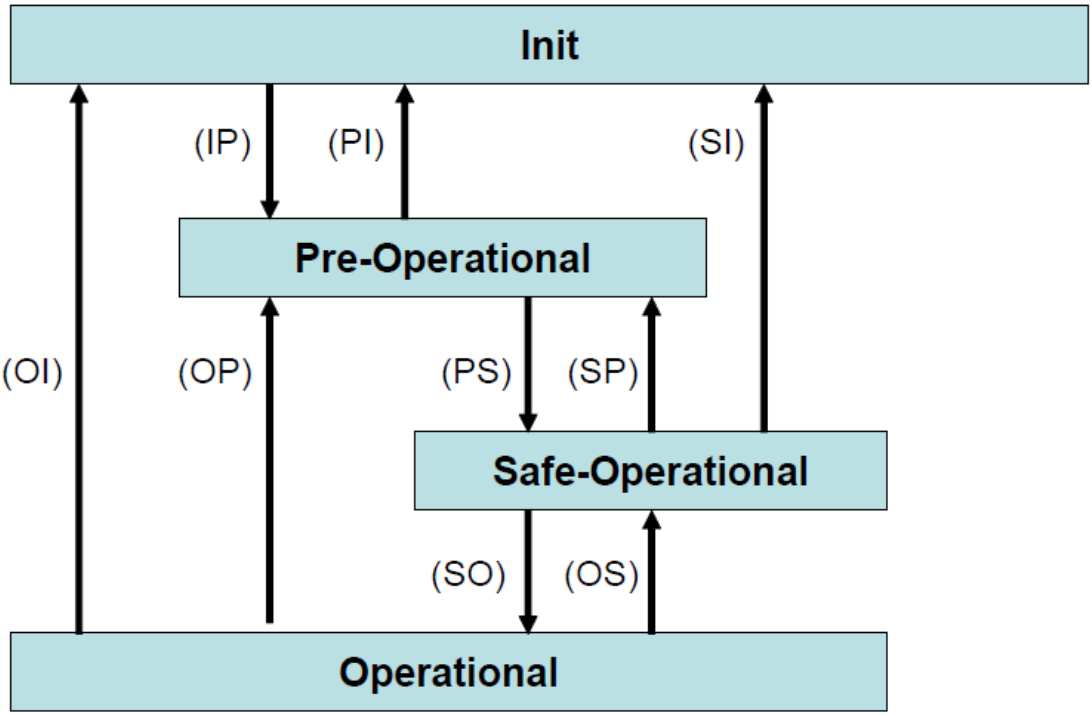


Figure 15: EtherCAT state machine

Init

After switch-on the EtherCAT slave is in the initial state. Only ESC register communication is possible, but no mailbox or process data communication. The slave initializes the service object data with default value or with values previously stored to the local memory. The EtherCAT master assigns the station address and configures the sync manager channels 0 and 1 for acyclic mailbox communication.

Pre-Operational (Pre-Op)

In Pre-Op state acyclic mailbox communication is possible, but not process data communication. In this state the EtherCAT master does the following configurations:

- Set the sync manager 2 and 3 of the ECAT-2094s for process data communication (from sync manager channel 2)
- The FMMU channels
- PDO mapping or the sync manager PDO assignment
- The user has the option to save motion control related configuration data (0x8000 - 0x8321) to a non-volatile memory.

Safe-Operational (Safe-Op)

In Safe-Op state both mailbox and process data communication are enabled, but the slave keeps its outputs in a safe state, while the input data are updated cyclically. The slave will ignore the output data sent by the master and just return the current input data (e.g. digital input, encoder value, etc.)

Outputs in Safe-Op state
The sync manager watchdog expires when the master application does not provide new output process data within the configured watchdog time. In this case the slave will automatically go from operational state to ERROR-SAFEOP state and set all the outputs in a safe state. The ECAT-2094S will stop the stepper motor, regulate the motor current to the configured safe level and switch the digital output to safe output values. All safe output value can be configured.

Operational (Op)
Here both the process data object (PDO) and service data object (SDP) are fully enabled. Master sends cyclic output data and read input data. The ECAT-2094S supports two type of Op modes: Free Run mode and Distributed Clock (DC) mode.

4.3 Synchronization Modes

ECAT-2094s devices support two different modes

- Free Run: The master cycle time and slave cycle time are independent and not synchronized.
- Distributed Clock (DC): The master cycle time and slave cycle time are synchronized.

4.3.1 Free Run Mode

The slave operates autonomously based on its own cycle and is not synchronized with the EtherCAT cycle. The master cycle time and the slave cycle time are fully independent which means each slave device reads/writes its own process data according to its local time, independent of the master's cycle time.

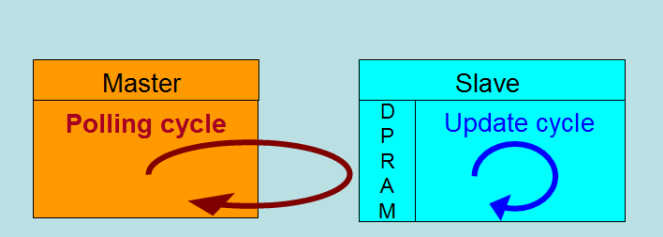


Figure 16: Master-slave cycle in Free Run mode

The following diagram shows the process timing of the slave in Free Run mode in detail:

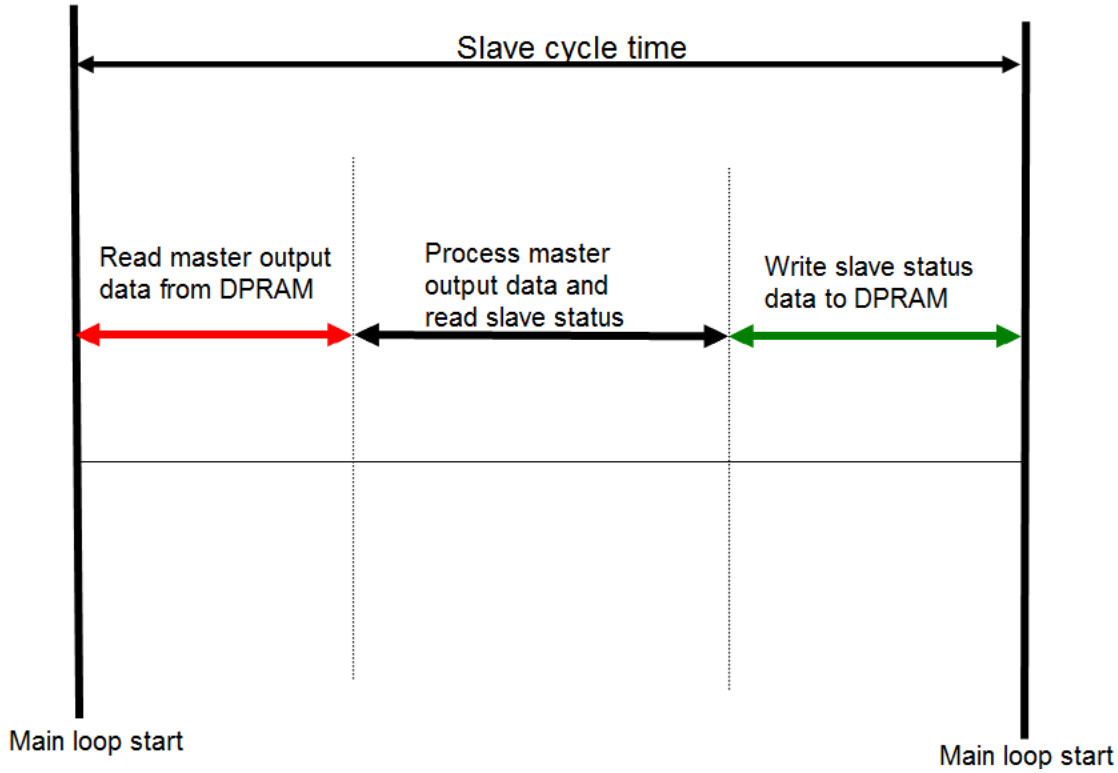


Figure 17: Slave processing sequence in Free-run mode

The slave firmware checks in each cycle time the memory of the EtherCAT slave chip (ESC) whether new output data has been received from the master. Newly received data will be processed, the motion path will be calculated and sent to the motion chip and digital output will be set. In the next step motion and digital input status are being read from motion chip. In the final step the read status are being written to the DPRAM, so that the master can retrieve the data ESC DPRAM in the next cycle time.

4.3.2 Distributed Clocks (DC Mode)

DC clock synchronization enables all EtherCAT devices (master and slaves) to share the same EtherCAT system time. The EtherCAT slaves in the network can be synchronized to each other. This enables the master to simultaneously set the output (e.g. digital output, pulse output) or to synchronously read inputs (e.g. digital input, encoder counter) of different slaves in the EtherCAT network.

For system synchronization all slaves are synchronized to one reference clock. Normally the first EtherCAT slave closest to the master with Distributed Clocks capability becomes the clock base for the master as well as for other DC slaves.

The EtherCAT slave is synchronized with the SYNC0 or SYNC1 event of the distributed

clock system. After the EtherCAT network has been set into DC communication mode by the master, the ESC (EtherCAT slave chip) of each slave generates fixed time hardware interrupt which triggers the slave firmware to process the PDO data received by the master. The master cycle time and the ESC hardware interrupt time interval are fully synchronized to the first slave in the network that is used as a reference clock with the SYNC0 signal.

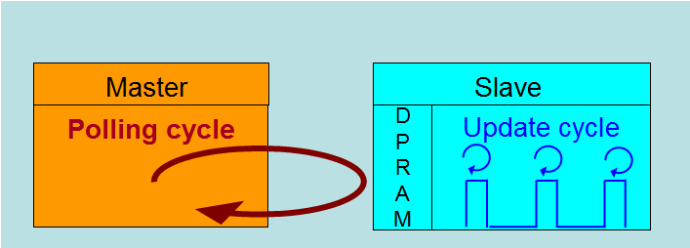


Figure 18: Master-slave cycle in DC mode

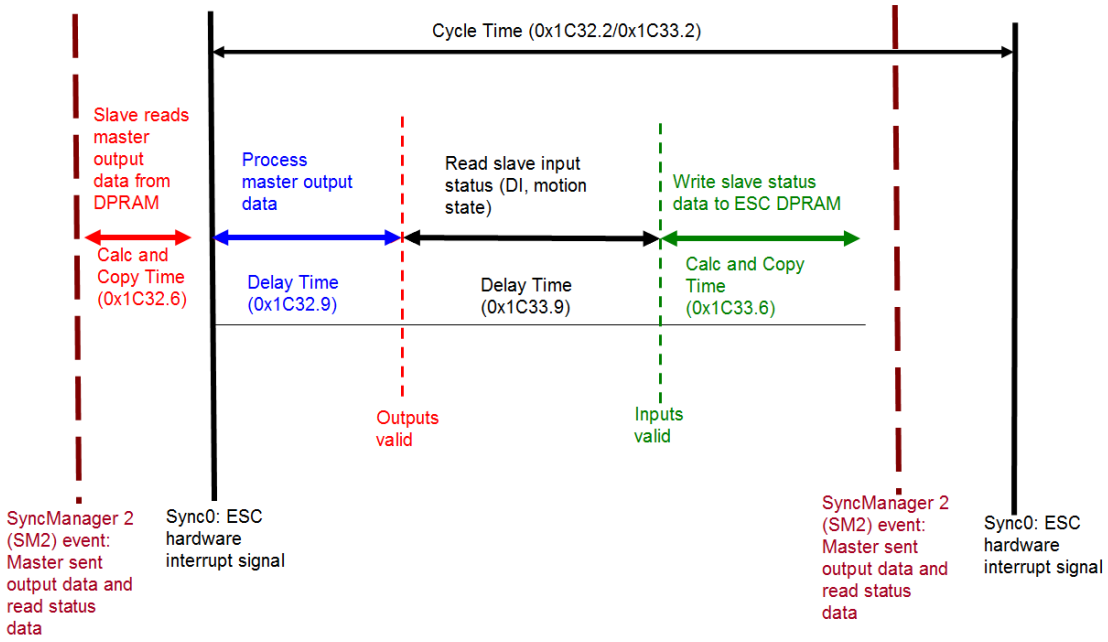


Figure 19: Internal slave processing sequence in DC mode

Once the slave receives process data (RxPDOs) from the master the SM2 event is triggered which causes the firmware to read the data from the ESC memory. The ESC interrupts the firmware at fixed time interval to process the data received from the master and write the status data to the ESC memory. Every time when the master fails to sent process data within the DC cycle time the internal sync error counter is being increase by three counts. This error counter is being decreased by one count for every successful DC cycle. Once the error counter reached the maximum count (default 4) a sync error will be generated and the slave goes into Safe OP mode (Sync Error 0x1C32:20 TRUE). The maximum count value can be set by changing the default value of

the "Sync Error Counter Limit" (0x10F1:02).

Index	Name	Flags	Value
10F1:0	Error Settings		> 2 <
10F1:01	Local Error Reaction	RW	0x00000001 (1)
10F1:02	Sync Error Counter Limit	RW	0x0004 (4)

Figure 20: Sync error counter limit object

The setting of the sync manager for the output and input data is available at the TwinCAT "CoE online" tab.

Index	Name	Flags	Value
1C32:0	SM output parameter		> 32 <
1C32:01	Synchronization Type	RW	0x0002 (2)
1C32:02	Cycle Time	RO	0x00000000 (0)
1C32:04	Synchronization Types supported	RO	0x401F (16415)
1C32:05	Minimum Cycle Time	RO	0x001E8480 (2000000)
1C32:06	Calc and Copy Time	RO	0x0007A120 (500000)
1C32:08	Get Cycle Time	RW	0x0001 (1)
1C32:09	Delay Time	RO	0x000927C0 (600000)
1C32:0A	Sync0 Cycle Time	RW	0x005B8D80 (6000000)
1C32:0B	SM-Event Missed	RO	0x0000 (0)
1C32:0C	Cycle Time Too Small	RO	0x0000 (0)
1C32:20	Sync Error	RO	FALSE

Figure 21: SyncManager 2 parameters

SyncManager parameter description (time unit: nanosecond):

- Calc and Copy Time (0x1C32.6 / 0x1C33.6): Required time to copy the process data from the ESC to the local memory and calculate the output value.
- Delay Time (0x1C32.9 / 0x1C33.9): Delay from receiving the trigger to set the output or latch the input.
- Cycle Time (0x1C32.2 / 0x1C33.2): The current cycle time for the application. When using DC synchronization the value is read from register 0x9A0:0x9A3.
- 0x1C32.5 / 0x1C33.5 (Min Cycle Time): Minimum cycle time for the application. It is the total execution time of all slave application related operations.

5 Project Integration

In this chapter the integration of the ECAT-2094S device into a TwinCAT controlled EtherCAT network is being described. In general the ECAT-2094S is a standard EtherCAT slave which can be controlled by any standard EtherCAT master (e.g. Acontis, CODESYS, etc.).

5.1 ESI File

A ESI file describes the properties and functions supported by the ECAT-2094S. By using the ESI file an easy and abstract integration of an EtherCAT device in a project tool is realized. With the help of the ESI file a detailed knowledge of EtherCAT is not required to configure the device. The TwinCAT EtherCAT master/System Manager needs the device description files in order to generate device configuration in online or offline mode.

5.1.1 Import of ESI File

Copy the XML description file "ECAT-2094S.xml" of the ECAT-2094S device into the TwinCAT system directory and restart the TwinCAT system.
For TwinCat 3.1 copy the ESI file "ECAT-2094S.xml" in the following directory:
C:\TwinCAT\3.1\Config\Io\EtherCAT

Software	Default directory path
Beckhoff EtherCAT Configuration	C:\EtherCAT Configurator\EtherCAT
Beckhoff TwinCAT 3.x	C:\TwinCAT\3.x\Config\Io\EtherCAT
Beckhof TwinCAT 2.x	C:\TwinCAT\Io\EtherCAT

Table 8: ESI file target directory

5.2 Device Setup and Configuration

In this manual only the online configuration of the slave module will be discussed. For offline configuration procedure please consult the TwinCAT user manual.



CAUTION:
Automatic start of stepper motor!

- Risk of death or serious injury for humans working in the machine.
- It can not ruled out that the stepper motor may perform unplanned movement during the ECAT-2094S setup and configuration
- Make sure that, even if the drive starts to move unintentionally, no danger can result for personnel or machinery. The measures you must take in this regard for your task are based on the risk assessment of the application.

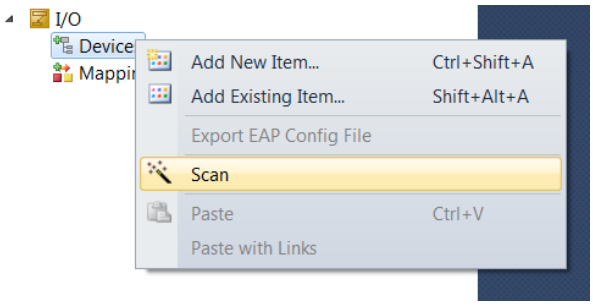
The following conditions must be met before a configuration can be set up:

- The ECAT-2094S slave devices must be connected via EtherCAT cables to the EtherCAT master. In this manual TwinCAT 3.1 version is being used as the EtherCAT master and configuration tool
- The ECAT-2094S devices has to be connected to the power supply and ready for communication
- Set the TwinCAT in CONFIG mode.

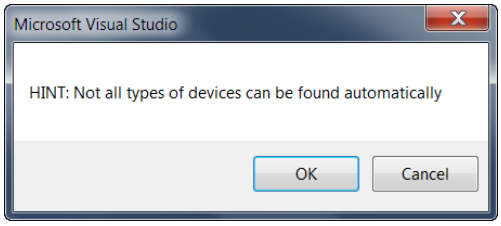
5.2.1 Scanning of the EtherCAT Device

After the TwinCAT has been set into CONFIG mode the online device search can be started.

Step 1: Right-click the “Devices” in the configuration tree to open the scan dialog. Click "Scan" to search the ECAT-2094S device.



Step 2: Select "OK"



Step 3: Select the Ethernet device (Ethernet chip) to which the ECAT-2094S is connected to. Confirm the selection with "OK".

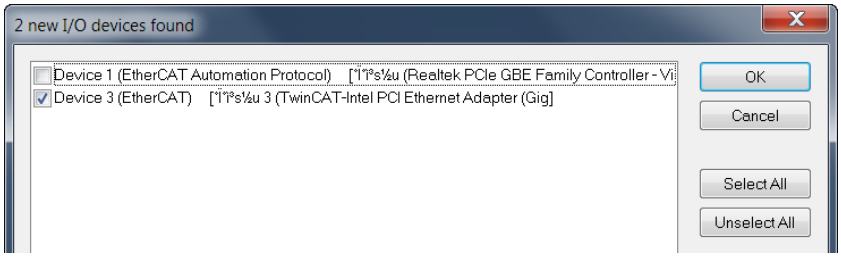
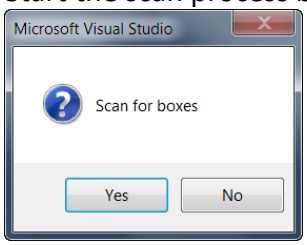
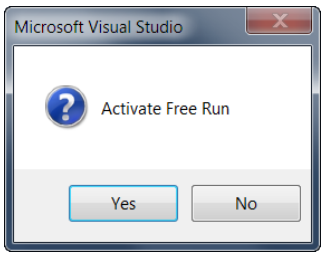


Figure 22:List of Ethernet chips detected on the EtherCAT master PC

Step 4: Start the scan process by clicking "Yes"



Step 5: Set the ECAT-2094S into Free-Run mode by clicking "Yes"



The ECAT-2094S is by default in the velocity mode. All the parameter used by the velocity control mode is being displayed in the tree view:

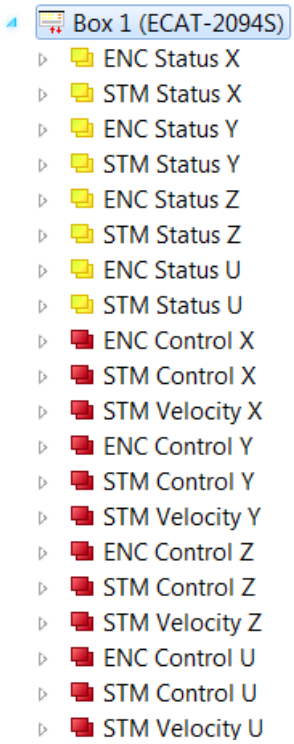


Figure 23: Default parameter selection for the velocity control mode

5.2.2 EtherCAT Slave Process Data Settings

The user has to select the process data which is being transferred between the EtherCAT master and slave during each cycle (Process Data Objects, PDOs). The process data exist of two parts:

- TxPDO: Data which is being read by the master (e.g. motion status)
- RxPDO: data or parameters which is being sent to the slave (e.g. target position of the stepper motor) .

The process data image is determined by the application program and will be updated cyclically.

The ECAT-2094S basically support four types of motion modes:

- Velocity control
- Position control
- Position interface compact
- Position interface

By selecting one of the motion mode from the list box (Figure 24) all the relevant parameters are automatically assigned and mapped to the process data objects (TxPDO,

RxPDO). If required, additional objects can be assigned to the process data by selecting the object listed under "PDO Assignment (0x1C12)" and "PDO Assignment (0x1C13)".

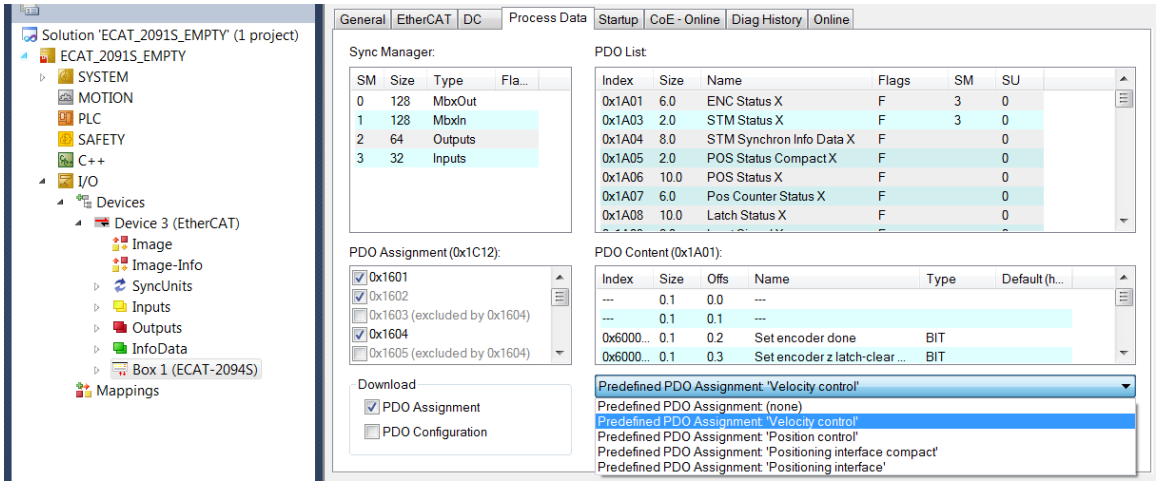


Figure 24: Predefined PDO assignment selection

Download the new PDO assignment to the Sync manager of the slave by clicking "Restart TwinCAT (Config Mode)" in the drop down menu.

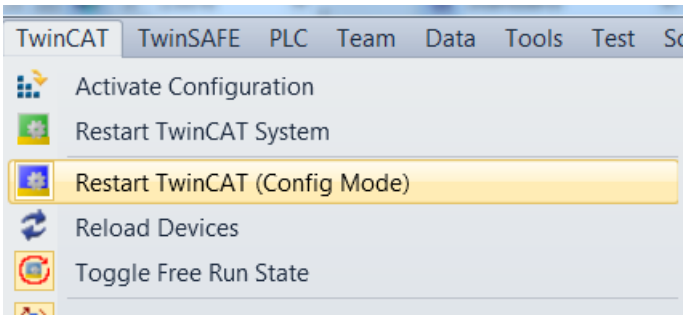


Figure 25: Download PDO assignment and restart TwinCAT

5.2.3 Basic Stepper Driver Configuration

Motion parameters which do only need to be configured once before the actual motion control starts are listed in the "CoE online" tab. These parameters have to be accessed via the CANopen over EtherCAT (CoE) protocol. The CoE protocol has a lower priority than the cyclic process data object (PDO) communication. Therefore CoE motion parameters will not be updated in every cycle but only when the master has spare time.

Motion relevant CoE parameter are

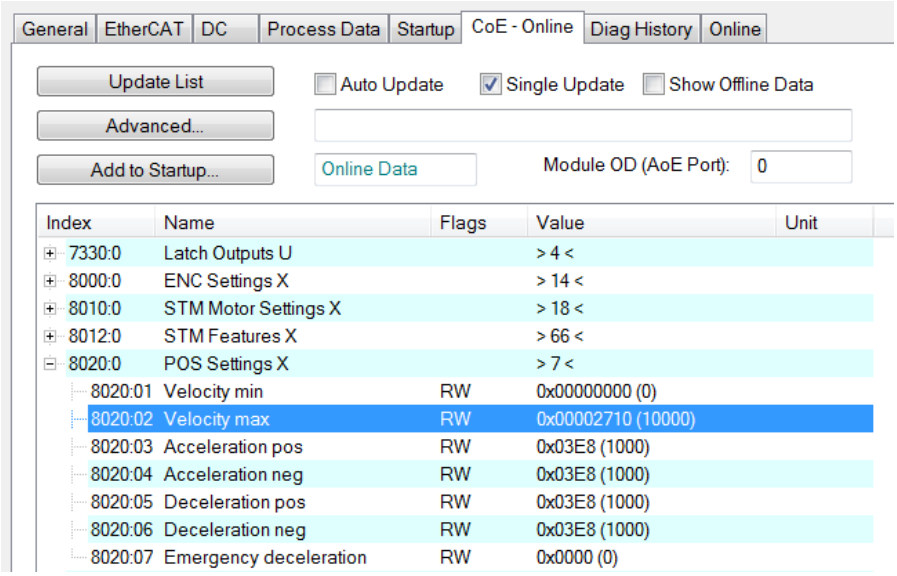
- Encoder setting (Index 8n00)
- Stepper motor setting (Index 8n10)
- Stepper motor features (Index 8n12)

- POS setting (Index 8n20)
- POS features (Index 8n21)

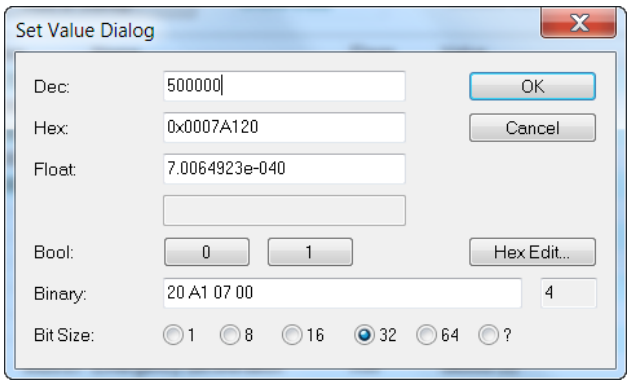
Whereby "n" represents the motor number (0 to 3)

Example of setting the maximum allowable speed for motor X:

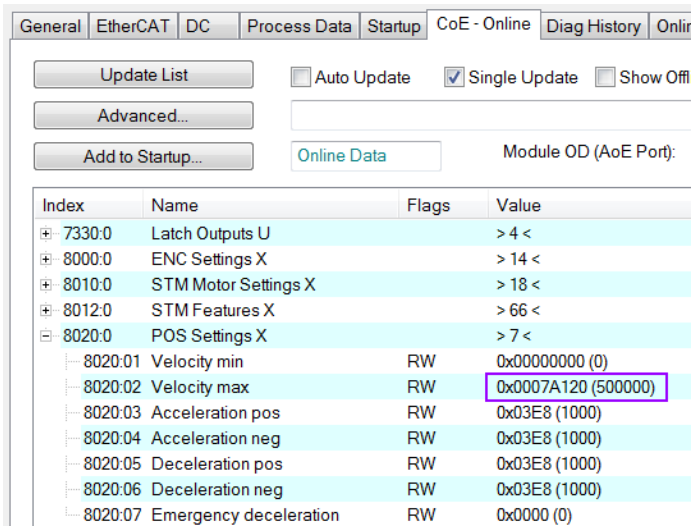
Step 1: Go to POS Setting X. Extend the index tree and double click "Velocity max" with the index 8020:02.



Step 2: Enter a new value for the maximum allowable velocity [steps/second] and click "OK"



Step 3: Once the value has been successfully sent to the slave it will be displayed in the CoE online parameter list:



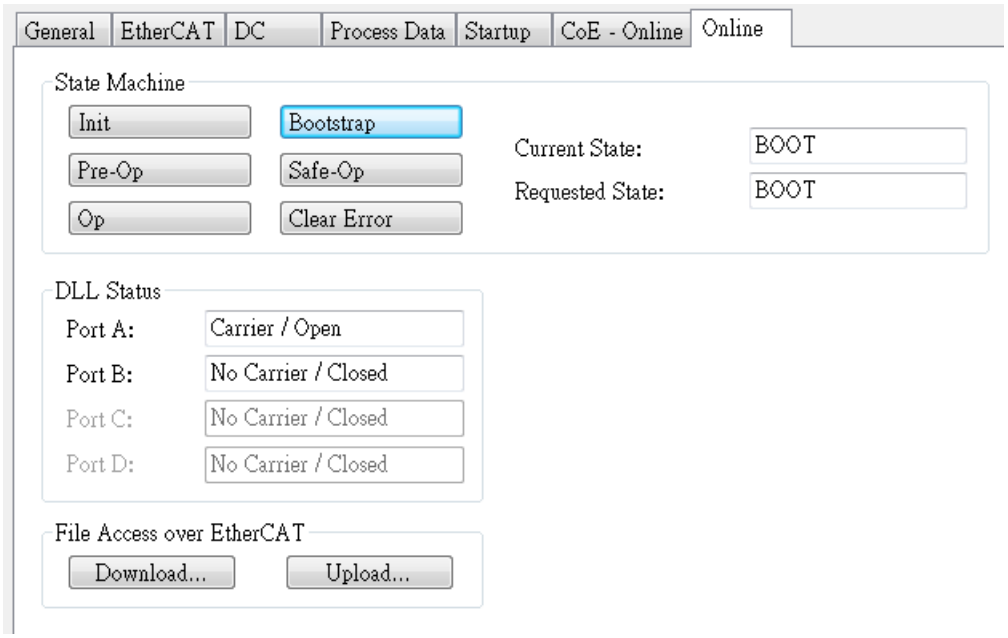
This value needs only to be set once and therefore does not have to be sent in every cycle time.

All the relevant motion parameters have to be set first before the actual real time motion control starts. Once the parameters are set, the four motors are basically ready for operation.

5.3 Firmware Update

This section describes the firmware update for ECAT-2094S (with version 2.0 or later) via FoE (File access over EtherCAT).

Set the state machine of module to Bootstrasp in CONFIG mode in TwinCAT software. After the state has been set into BOOT, click “Download” button in the "File Access over EtherCAT" groupbox, select the update file "ECAT-2094S_v2.x.efw", and then click “OK” to process update.


General
EtherCAT
DC
Process Data
Startup
CoE - Online
Online
State Machine
Init
Bootstrap
Pre-Op
Safe-Op
Op
Clear Error
Current State:
BOOT
Requested State:
BOOT
DLL Status
Port A:
Carrier / Open
Port B:
No Carrier / Closed
Port C:
No Carrier / Closed
Port D:
No Carrier / Closed
File Access over EtherCAT
Download...
Upload...

6 Position Control Setting

The position interface allows the user to set a target position and the motion controller automatically drives the motor to the specified position. The basic motion configuration data such as the acceleration and deceleration values and the maximum motion velocity have to be set before motion control execution can be started.

6.1 Positioning Interface Types

Two predefined PDO assignment types for the position interface are provided:

- Positioning interface
- Positioning interface compact

The predefined PDO assignment enables a simplified selection of the process data.

The "Positioning interface" type activates all the position control PDOs required to execute point to point motion. The created process data image is quite large because it contains motion parameters needed to control the four motors. This large process data is being transmitted in every cycle and slows down the system. If communication speed and a small process data image are a criteria for the system setup then the "Positioning interface compact" type should be activated. Here most motion parameter values are not send at a fixed, deterministic cycle but set via CoE. In an application where the motion parameters (velocity, acceleration, deceleration etc.) only need to be set once in a while the "Positioning interface compact" type is the better option.

In the following the parameter settings for both positioning interface types will be discussed in details.

6.2 Positioning Interface

The sequence of executing and controlling a travel command in "Positioning interface" mode is shown in the following flow diagram (Figure 26). The diagram shows the sequence of parameter setting and status checking during the execution of a position command. The configuration parameter setting has to be done beforehand.

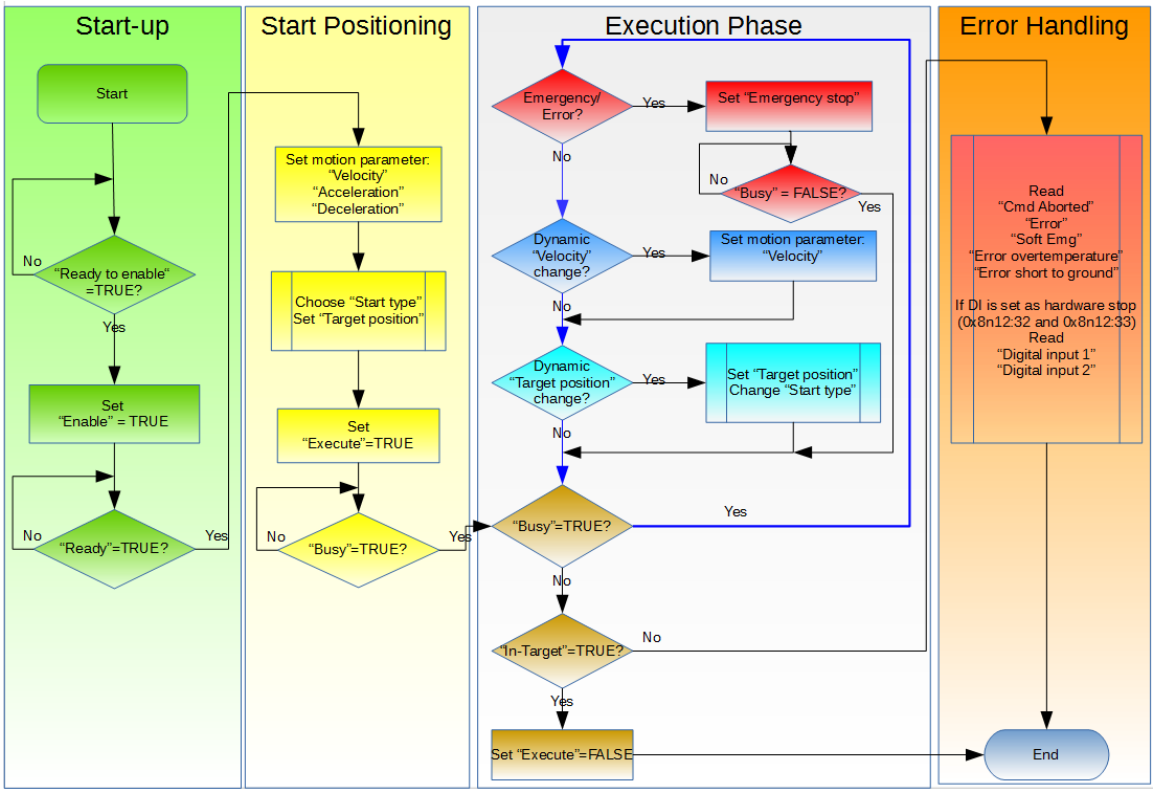


Figure 26: Flow diagram for position interface

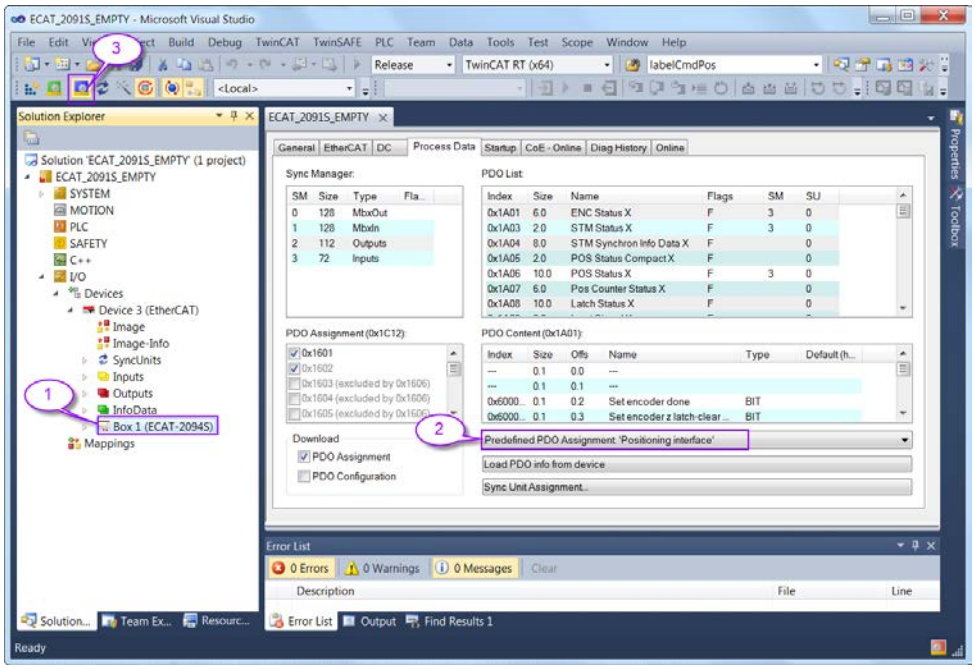
Execution procedure:

Step 1: PDO assignment

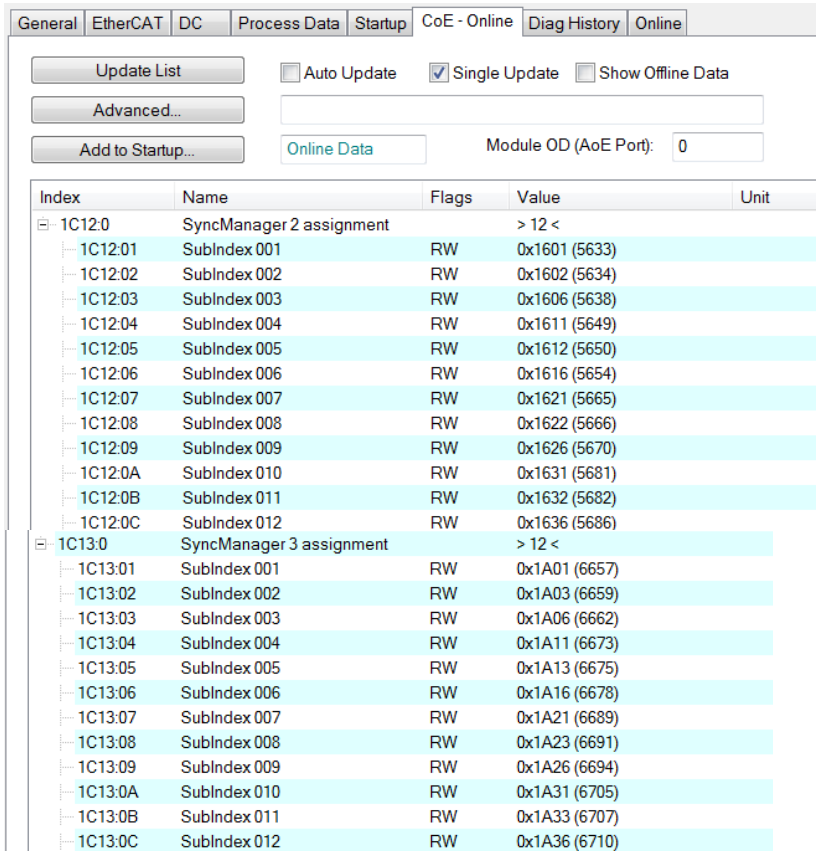
Select the function "Positioning interface" in the lower part of the "Process data" tab. As a result, all necessary PDOs are automatically activated and the unnecessary PDOs are deactivated.

Procedure:

1. Select the "Process Data" tab of the ECAT-2094S
2. Select the "Predefined PDO Assignment: "Position interface" from the combo box
3. Send the PDO assignment to the slave by clicking the "Reload I/O device" button



The SyncManager 2 and 3 in the "CoE-Online" tab displays the new PDO assignment:



Index	Name	Flags	Value	Unit
1C12:0	SyncManager 2 assignment		> 12 <	
1C12:01	SubIndex 001	RW	0x1601 (5633)	
1C12:02	SubIndex 002	RW	0x1602 (5634)	
1C12:03	SubIndex 003	RW	0x1606 (5638)	
1C12:04	SubIndex 004	RW	0x1611 (5649)	
1C12:05	SubIndex 005	RW	0x1612 (5650)	
1C12:06	SubIndex 006	RW	0x1616 (5654)	
1C12:07	SubIndex 007	RW	0x1621 (5665)	
1C12:08	SubIndex 008	RW	0x1622 (5666)	
1C12:09	SubIndex 009	RW	0x1626 (5670)	
1C12:0A	SubIndex 010	RW	0x1631 (5681)	
1C12:0B	SubIndex 011	RW	0x1632 (5682)	
1C12:0C	SubIndex 012	RW	0x1636 (5686)	
1C13:0	SyncManager 3 assignment		> 12 <	
1C13:01	SubIndex 001	RW	0x1A01 (6657)	
1C13:02	SubIndex 002	RW	0x1A03 (6659)	
1C13:03	SubIndex 003	RW	0x1A06 (6662)	
1C13:04	SubIndex 004	RW	0x1A11 (6673)	
1C13:05	SubIndex 005	RW	0x1A13 (6675)	
1C13:06	SubIndex 006	RW	0x1A16 (6678)	
1C13:07	SubIndex 007	RW	0x1A21 (6689)	
1C13:08	SubIndex 008	RW	0x1A23 (6691)	
1C13:09	SubIndex 009	RW	0x1A26 (6694)	
1C13:0A	SubIndex 010	RW	0x1A31 (6705)	
1C13:0B	SubIndex 011	RW	0x1A33 (6707)	
1C13:0C	SubIndex 012	RW	0x1A36 (6710)	

Step 2: Set the motor torque:

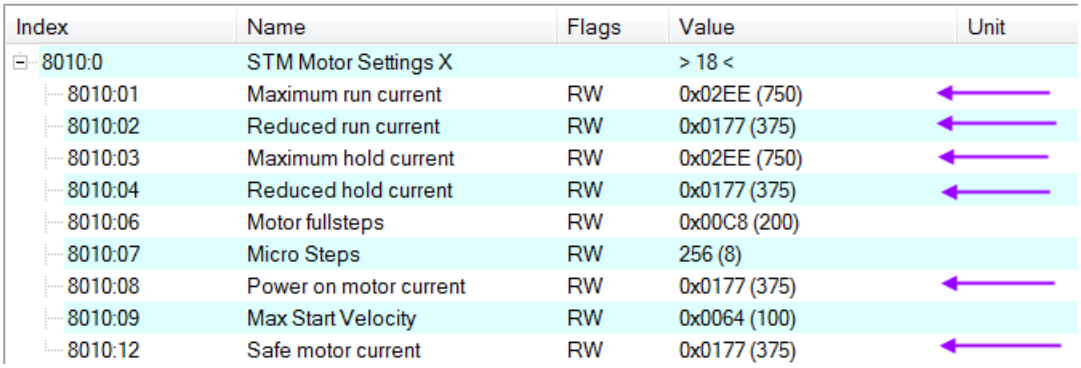
Be careful when setting the torque. Motor current fine tuning is required to lower motor temperature and reduce the current to save power.

Torque produced by the stepper motors is directly proportional to the current, but the amount of heat generated is roughly proportional to the *square* of the current. If the motor is operated at 90% of rated current, 90% of the rated torque will be outputted. But the motor will produce approximately 81% as much heat compared to the maximum torque output. At 70% current, the torque is reduced to 70% and the heating to about 50%.

Attention:
If the motor current is set at or above 1.1A for increased periods of time the ECAT-2094S will heat up and emit increasing heat as the resistive power dissipation raises with the square of the motor current.

Four torque settings have to be done. The valid range for the motor current setting is 0 to 1500 mA. The unit for the motor current parameters is milliamperes [mA].

1. The "Maximal run current" sets the motor driving current. This torque setting will be applied once the motion execution flag (0x7n10:01 - Enable) has been activated
2. "Reduce run current" output is triggered once the "Reduce torque" Boolean has been set to true (0x7n10:03 - Reduced torque).
3. The "Maximal hold current" sets the motor standstill current. This torque setting will be applied once the motion execution flag (0x7n10:01 - Enable) has been activated
4. "Reduce hold current" output is triggered once the "Reduce torque" Boolean has been set to true (0x7n10:03 - Reduced torque).
5. The purpose of the "Power on motor current" (0x7n10:08) variable is to apply a torque to the driver directly after power on to prevent freewheeling.
6. In case the EtherCAT communication is interrupted or when the EtherCAT master sets the ECAT-2094s from OP mode into a non-OP mode while a motion command is being executed then the ECAT-2094s uses the "Safe motor current" (0x7n10:12) setting to prevent the motor from freewheeling.



Index	Name	Flags	Value	Unit
8010:0	STM Motor Settings X		> 18 <	
8010:01	Maximum run current	RW	0x02EE (750)	
8010:02	Reduced run current	RW	0x0177 (375)	
8010:03	Maximum hold current	RW	0x02EE (750)	
8010:04	Reduced hold current	RW	0x0177 (375)	
8010:06	Motor fullsteps	RW	0x00C8 (200)	
8010:07	Micro Steps	RW	256 (8)	
8010:08	Power on motor current	RW	0x0177 (375)	
8010:09	Max Start Velocity	RW	0x0064 (100)	
8010:12	Safe motor current	RW	0x0177 (375)	

Step 3: Set the number of micro-steps per full step (8n10:07). The motor runs smoother and with less vibration with higher micro-steps value setting, but also requires a higher step pulse frequency to achieve maximum speed.

Step 4: Set the motion parameters for the system: max velocity, max acceleration, etc.

1. Set the start velocity (unit: steps/second) (0x8n10:09)
2. Set the velocity range of the system (unit: steps/second) (0x8n20:01 and 0x8n20:02). The maximal velocity "Velocity max" ensures that under no circumstances the motor velocity will exceed this maximal value. The minimal velocity "Velocity min" defines the lowest velocity of the system and is being applied when changing the velocity during driving.
3. Set the "Acceleration unit" (0x8n20:08) and "Acc-Dec parameter definition" (0x8n20:09). The parameter definitions a described in section 6.2.1 and 6.2.2.
4. "Emergency deceleration" determines how fast the motor decelerates to stop once the emergency stop flag has been raised (0x7n20:02 - Emergency stop)

Index	Name	Flags	Value	Unit
8010:0	STM Motor Settings		> 18 <	
8010:01	Maximum run current	RW	0x02EE (750)	
8010:02	Reduced run current	RW	0x0177 (375)	
8010:03	Maximum hold current	RW	0x02EE (750)	
8010:04	Reduced hold current	RW	0x0177 (375)	
8010:06	Motor fullsteps	RW	0x00C8 (200)	
8010:07	Micro Steps	RW	256 (8)	
8010:08	Power on motor current	RW	0x0177 (375)	
8010:09	Max Start Velocity	RW	0x0064 (100)	
8010:12	Safe motor current	RW	0x0177 (375)	
8012:0	STM Features		> 67 <	
8020:0	POS Settings		> 13 <	
8020:01	Velocity min	RW	0x00000000 (0)	
8020:02	Velocity max	RW	0x00002710 (10000)	
8020:03	Acceleration pos	RW	0x03E8 (1000)	
8020:04	Acceleration neg	RW	0x03E8 (1000)	
8020:05	Deceleration pos	RW	0x03E8 (1000)	
8020:06	Deceleration neg	RW	0x03E8 (1000)	
8020:07	Emergency deceleration	RW	0x0000 (0)	
8020:08	Acceleration unit	RW	Acceleration time from Vmin to Vmax [ms] (0)	
8020:09	Acc-Dec parameter definition	RW	Acceleration-->Start phase & Deceleration-->Stop Phase (0)	
8020:0D	Hardlimit deceleration	RW	0x0000 (0)	
8021:0	POS Features		> 1 <	

Step 5: Motion execution procedure:

- STM Control X
  - Enable
  - Reset
  - Reduce torque
  - Digital output1
- POS Control X
  - Execute
  - Emergency stop
  - Target position
  - Velocity
  - Start type
  - Acceleration
  - Deceleration

1. Activate the Enable (0x7n10:01)
2. Set the motion parameters: acceleration and deceleration, the target velocity (steps/seconds) and target position (steps).
   - i. Set the target velocity (unit: steps/second)
   - ii. Set the "Acceleration" and "Deceleration" values. The unit and behavior of both parameters are determined by the SDO variables "Acceleration unit" (0x8n20:08) and "Acc-Dec parameter definition" (0x8n20:09) which are described in the next section 6.2.1 and 6.2.2.
   - iii. The "Start type" (0x7n20:22) describes whether the target position is a relative or absolute position. In addition it is possible to set with the start type parameter whether a running motion command can be overwritten.

Name	Command	Description

Name	Command	Description
ABSOLUTE	0x0001	The motor travels from the current position to the target position. The distance to travel depends on the distance difference between the current and target position
RELATIVE	0x0002	A specified position difference is added to the current position
ENDLESS_PLUS	0x0003	Endless travel in the positive direction of rotation
ENDLESS_MINUS	0x0004	Endless travel in the negative direction of rotation
ADDITIVE	0x0006	A specified position difference is added to the last target position **Note:** The RELATIVE and ADDITIVE type are similar when the last command was completed successfully. In this case both types will travel the same position because both start positions are the same. If an error occurred during the execution of the previous command (e.g. motor stall, emergency stop) then the current position is arbitrary. Now the RELATIVE type will use the current arbitrary position as the start position but the ADDITIVE type will use the last target position as the start position. By selecting the ADDITIVE type the user has the advantage that he can use the last target position for determining the next target position. Therefore no home search needs to be done in case of an error.
ABSOLUTE_CHANGE	0x1001	Change of the target position on the fly: Dynamic change of the target position during a travel command to a new absolute position
RELATIVE_CHANGE	0x1002	Dynamic change of the target position during a travel command to a new relative position (the current changing position value is used here also) Attention: Due to propagation delays it is not possible to determine exactly the actual position of the running motor. Reading the current position takes time and during this time the motor has already move to a new position. Therefore, there will be a difference between the desired target position and the actual target position.
ADDITIVE_CHANGE	0x1006	Dynamic change of the target position during a travel command to a new additive position (the last target position is used here)

Table 9: Start type definition

iv. Set the target position (unit: steps). The target position can be a relative

distance or a absolute position. The behavior of this parameter is being determined by the "Start type" setting.

3. Start motion execution by setting the "Execute"-variable to true (0x7n20:01).
4. If an emergency stop (0x7n20:02) has been activated during driving, then the "Emergency stop"-variable has to be set to false and "Execute" back to false before the next command can be executed
5. Error: If an error occurred during driving (overheating, EtherCAT communication failed, Master sets slave from OP to none OP mode, etc.) the error flag is activated (0x6n10:04 Error). In order to clear this flag the "Reset" variable has to be activated (0x7n10:02 - Reset) for one cycle time.

Example 1:

Dynamic change of the target position

Time	POS Control X Outputs	POS Status X Inputs	Descriptions
t1	Execute = 1 Target position = 250000 Velocity = 10000 Start type = 0x0001 Acceleration = 1000 Deceleration = 1000	Busy = 1 Accelerate = 1 Deceleration = 0 In-Target = 0	• Set the motion parameters: - Absolute position = 250000 - Acceleration unit and type has to be set via 0x8n20:08 and (0x8n20:09) • Start executing the motion command by triggering the 'Execute' input (FALSE --> TRUE) ▪ Acceleration phase
t2		Busy = 1 Accelerate = 0 Deceleration = 0 In-Target = 0	• Target velocity has been reached
t3	Target position = 220000 Velocity = 8000 Start type = 0x1001 Acceleration = 500 Deceleration = 500	Busy = 1 Accelerate = 0 Deceleration = 1 In-Target = 0	• Change target position, velocity and acc/dec on the fly
t4		Busy = 1 Accelerate = 0 Deceleration = 0 In-Target = 0	• New target velocity has been reached
t5		Busy = 1 Accelerate = 0 Deceleration = 1 In-Target = 0	• Start the deceleration phase to the target position
T6	Execute = 0	Busy = 0 Accelerate = 0 Deceleration = 0	• Target position has been reached • Set Execute to false

Time	POS Control X Outputs	POS Status X Inputs	Descriptions
		In-Target = 1	

Table 10: Change the target position on the fly

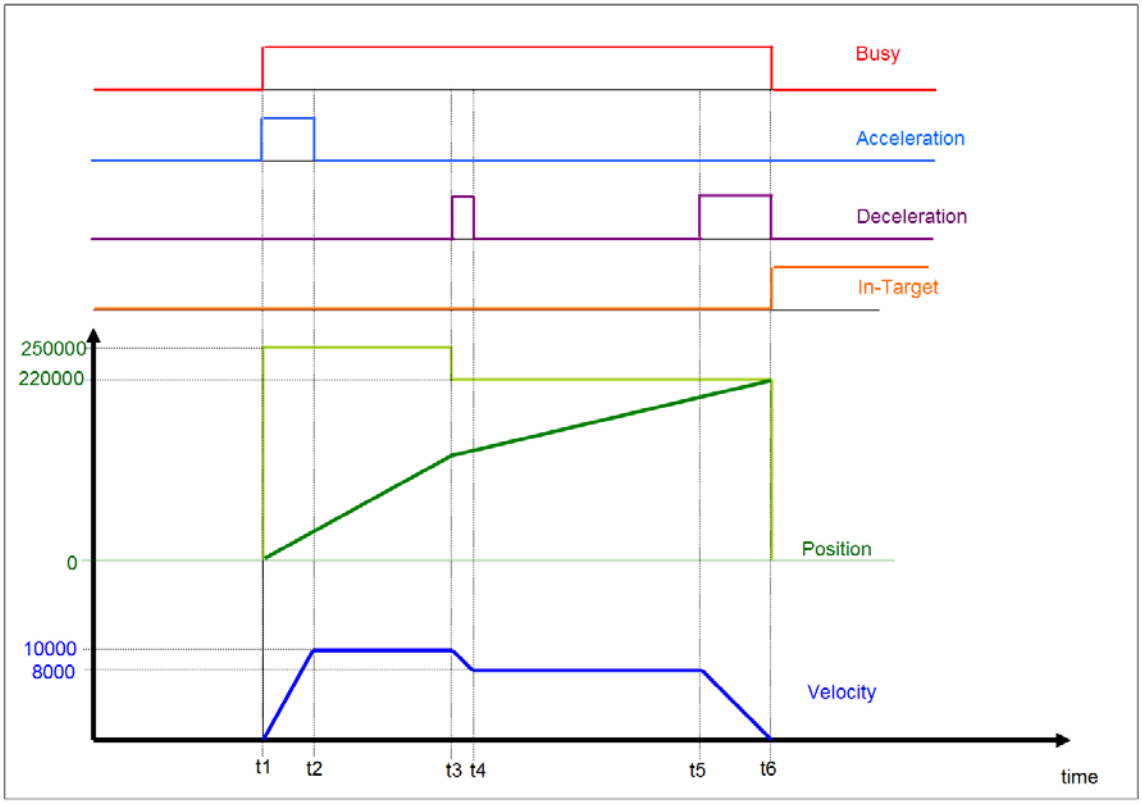


Figure 27: Output and input parameters on a time graph for changing the target position on the fly

Example 2:

Dynamic change of the relative position:

Time	POS Control Outputs	POS Status Inputs	Descriptions
t1	Execute = 1 Target position = 300000 Velocity = 11000 Start type = 0x0002 Acceleration = 1000 Deceleration = 1000	Busy = 1 Accelerate = 1 Deceleration = 0 In-Target = 0	• Set the motion parameters: - Relative position = 300000 - Acceleration unit and type has to be set via 0x8n20:08 and 0x8n20:09 • Start executing the motion command by triggering the 'Execute' input (FALSE --> TRUE) ▪ Acceleration phase
t2		Busy = 1 Accelerate = 0 Deceleration = 0 In-Target = 0	• Target velocity has been reached
t3	Target position = 50000	Busy = 1	• Change the relative position,

Time	POS Control Outputs	POS Status Inputs	Descriptions
	Velocity = 7000 Start type = 0x1002 Acceleration = 500 Deceleration = 500	Accelerate = 0 Deceleration = 1 In-Target = 0	velocity and acc/dec on the fly
t4		Busy = 1 Accelerate = 0 Deceleration = 0 In-Target = 0	• New target velocity has been reached
t5	Target position = 30000 Start type = 0x0000	Busy = 1 Accelerate = 0 Deceleration = 0 In-Target = 0	Procedure for changing relative position on the fly: • Set the 'Start type' to zero • Set the new relative distance to travel (Target position) • In the next EtherCAT cycle set 'Start type' back to 0x1002 After the 'Start type' has been set to back to 0x1002 the relative distance will be added to the current position. A new relative distance can be assigned even if the movement has ended. Once the new distance has been set ('Start type'=0; 'Start type'=0x1002) the motor accelerates from standstill to the new position.
t6	Start type = 0x1002	Busy = 1 Accelerate = 0 Deceleration = 0 In-Target = 0	
t7		Busy = 1 Accelerate = 0 Deceleration = 1 In-Target = 0	• Start the deceleration phase to the target position
t8	Execute = 0	Busy = 0 Accelerate = 0 Deceleration = 0 In-Target = 1	• Target position has been reached • Set Execute to false

Table 11: Change the relative distance, velocity, acceleration, deceleration on the fly

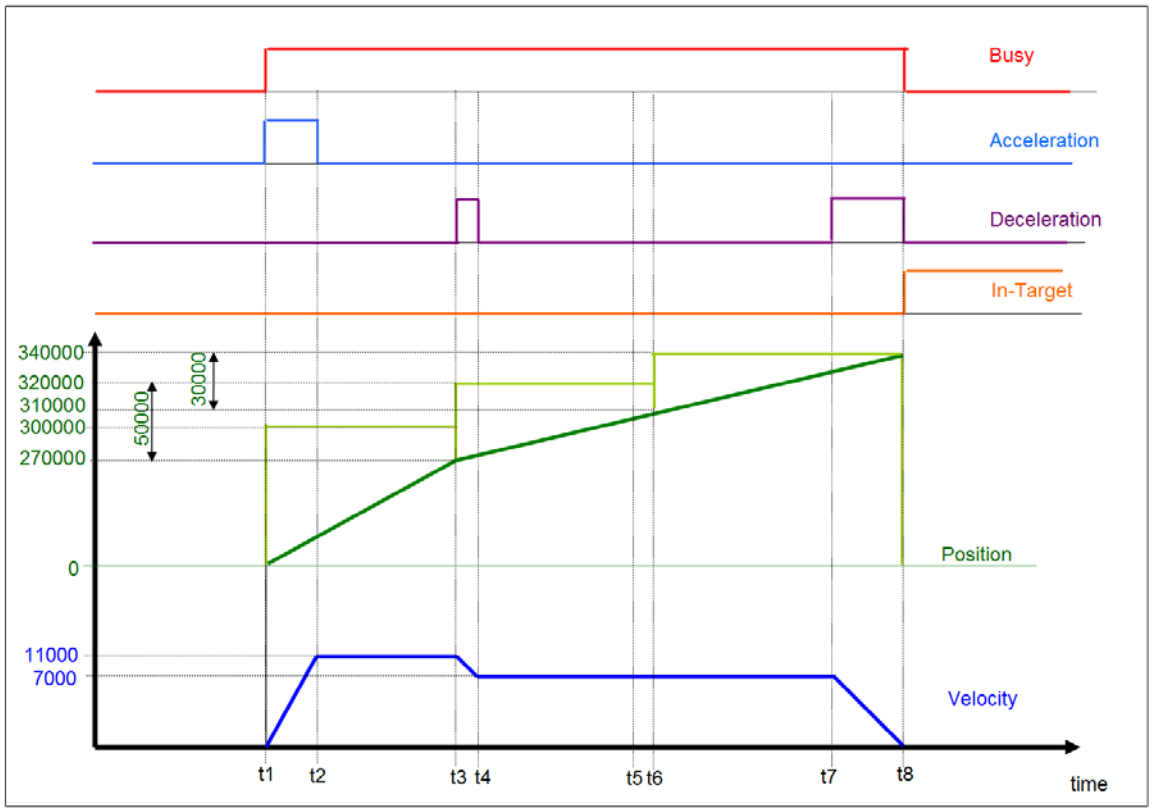


Figure 28: Output and input parameters on a time graph for changing the relative distance on the fly

6.2.1 Acceleration/Deceleration Unit Definition

The acceleration/deceleration unit has to be set via PDO (0x8n20:08). The units are supported (Table 12):

Name	Value
Acceleration time from Vmin to Vmax [ms]	0
Acceleration time from Vmin to Vtarget [ms]	1
Acceleration [128*uStep/sec^2]	2
Acceleration time from Vcurrent to Vtarget [ms]	3

Table 12: DT0814EN02 object

The unit definitions are described in more details below.

6.2.1.1 Acceleration time from Vmin to Vmax

The acceleration/deceleration gradient is determined by the "Velocity min" (0x8n20:01) and "Velocity max" (0x8n20:02) parameters setting (Figure 29). The "Acceleration" (0x7n20:23) and "Deceleration" (0x7n20:24) parameters sets the time [ms] to accelerate or decelerate the motor from "Velocity min" (0x8n20:01) to "Velocity max" (0x8n20:02).

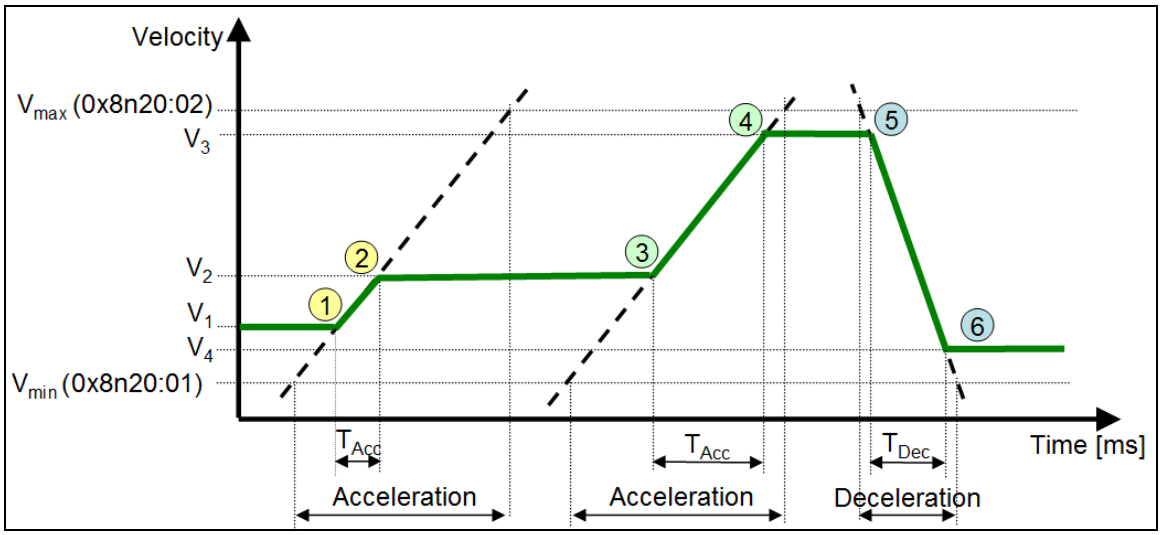
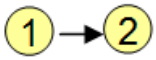


Figure 29: Acceleration unit - " Acceleration time [ms] from Vmin to Vmax "



Accelerate from the current V_1 to the target V_2 (0x7n20:21):

- The parameter "Acceleration" (0x7n20:23) sets the acceleration time [ms] to change the velocity from V_{min} (0x8n20:01) to V_{max} (0x8n20:02)
- T_{Acc}: The actual time [ms] to accelerate from the current V_1 to the target V_2.
  - $T_{Acc} = |V_2 - V_1| / Acc$
    with:
    $Acc = (V_{max} - V_{min}) / (Acceleration\ (0x7n20:23))$
- $T_{DecStop}$: The actual time [ms] to decelerate to standstill (V_2 to V_{min}).
  - $T_{DecStop} = (V_2 - V_{min}) / Dec$
    with:
    $Dec = (V_{max} - V_{min}) / (Deceleration\ (0x7n20:24))$



Accelerate from the current V_2 to the target V_3 (0x7n20:21):

- The parameter "Acceleration" (0x7n20:23) sets the acceleration time [ms] to change the velocity from V_{min} (0x8n20:01) to V_{max} (0x8n20:02)
- T_{Acc}: The actual time [ms] to accelerate from the current V_2 to the target V_3.
  - $T_{Acc} = (V_3 - V_2) / Acc$

with:

$Acc = (V_{max} - V_{min}) / (Acceleration\ (0x7n20:23))$

- $T_{DecStop}$: The actual time [ms] to decelerate to standstill (V_3 to V_{min}).
  - $T_{DecStop} = (V_3 - V_{min}) / Dec$

    with:

    $Dec = (V_{max} - V_{min}) / (Deceleration\ (0x7n20:24))$



Decelerate from the current V_3 to the new target V_4 (0x7n20:21):

- Depending on the deceleration type either the parameter "Acceleration" (0x7n20:23) (type 0) or the parameter "Deceleration" (0x7n20:24) (type 1) determines the time [ms] to reduce the speed from V_{max} to V_{min} .
- T_{Dec}: The actual time [ms] to decelerate from current velocity V_3 to the new target velocity V_4.
  - $T_{Dec} = |V_3 - V_4| / Dec$

    with:
    - Type 0: $Dec = (V_{max} - V_{min}) / (Acceleration\ (0x7n20:23))$
    - Type 1: $Dec = (V_{max} - V_{min}) / (Deceleration\ (0x7n20:24))$
- $T_{DecStop}$: The actual time [ms] to decelerate to standstill (V_4 to V_{min}).
  - $T_{DecStop} = |V_4 - V_{min}| / Dec$

    with:

    $Dec = (V_{max} - V_{min}) / (Deceleration\ (0x7n20:24))$

6.2.1.2 Acceleration time from Vmin to Vtarget

The acceleration/deceleration gradient is determined by the "Velocity min" (0x8n20:01) and target velocity (0x7n20:21) as shown in Figure 30.

Important:

- This mode should not be used to change the velocity while the axis is moving as this may cause a target overshoot due to too low acceleration. This situation may occur when changing the velocity from a high value to a very low value while moving.
- After the "Execute" has been triggered do not change the "Velocity", "Acceleration" and "Deceleration" parameters.

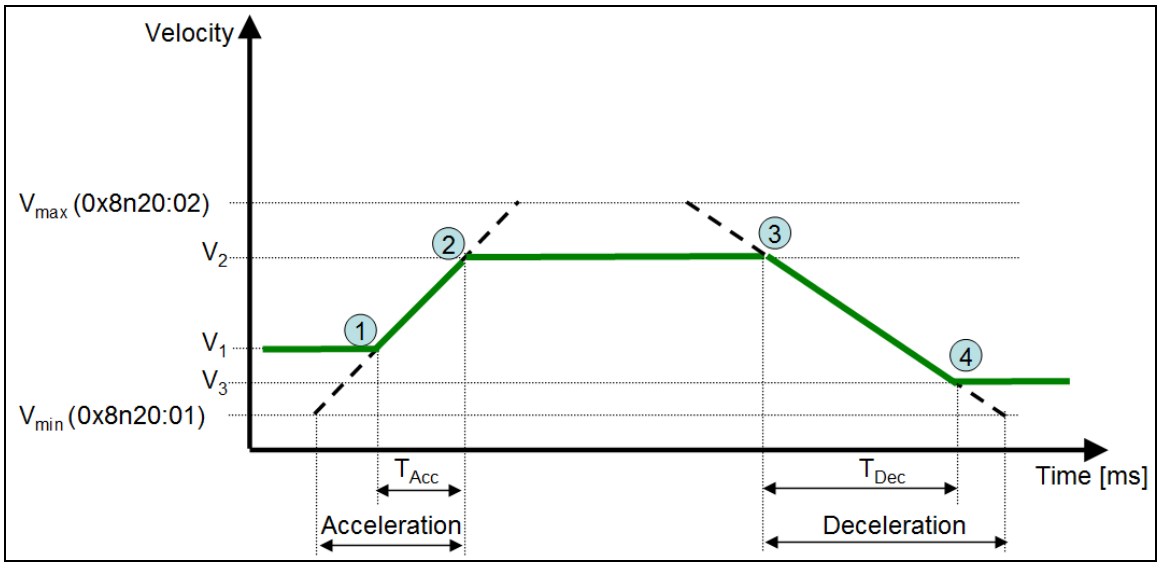


Figure 30: Acceleration unit - " Acceleration time [ms] from Vmin to Vtarget "



Accelerate from the current V_1 to the target V_2 (0x7n20:21):

- "Acceleration" (0x7n20:23): Acceleration time [ms] from V_{min} to V_2
- T_{Acc}: The actual time [ms] to accelerate from the current V_1 to the target V_2.
  - $T_{Acc} = |V_2 - V_1| / Acc$
    with:
    $Acc = |V_2 - V_{min}| / (Acceleration\ (0x7n20:23))$
- $T_{DecStop}$: The actual time [ms] to decelerate to standstill ("V_2" to the target "V_{min}").
  - $T_{DecStop} = |V_2 - V_1| / Dec$
    with:
    $Dec = |V_2 - V_{min}| / (Deceleration\ (0x7n20:24))$



Decelerate from the current V_2 to the new target V_3 (0x7n20:21):

- Depending on the deceleration type either the parameter "Acceleration" (0x7n20:23) (type 0) or the parameter "Deceleration" (0x7n20:24) (type 1) determines the time [ms] to reduce the speed from V_3 to V_{min} .
- T_{Dec}: The actual time [ms] to decelerate from current velocity V_2 to V_3.
  - $T_{Dec} = |V_2 - V_3| / Dec$
    with:
    - Type 0: $Dec = (V_3 - V_{min}) / (Acceleration\ (0x7n20:23))$
    - Type 1: $Dec = (V_3 - V_{min}) / (Deceleration\ (0x7n20:24))$
- $T_{DecStop}$: The actual time [ms] to decelerate to standstill (V_3 to V_{min}).
  - $T_{DecStop} = |V_3 - V_{min}| / Dec$
    with:

$Dec = (V_3 - V_{min}) / (Deceleration\ (0x7n20:24))$

Attention:
If the new target velocity V_3 is close to V_{min} the then the Dec acceleration is set to a very low number which causes a very long deceleration time. It is therefore not suggested to use this unit setting when target velocity V_3 is close to V_{min}.

6.2.1.3 Acceleration [128*µStep/sec^2]

The acceleration/deceleration value are directly set via the Acceleration (0x7n20:23) and the Deceleration (0x7n20:24) parameters (Figure 31). The unit for both parameters are [128* μ Step/sec2].

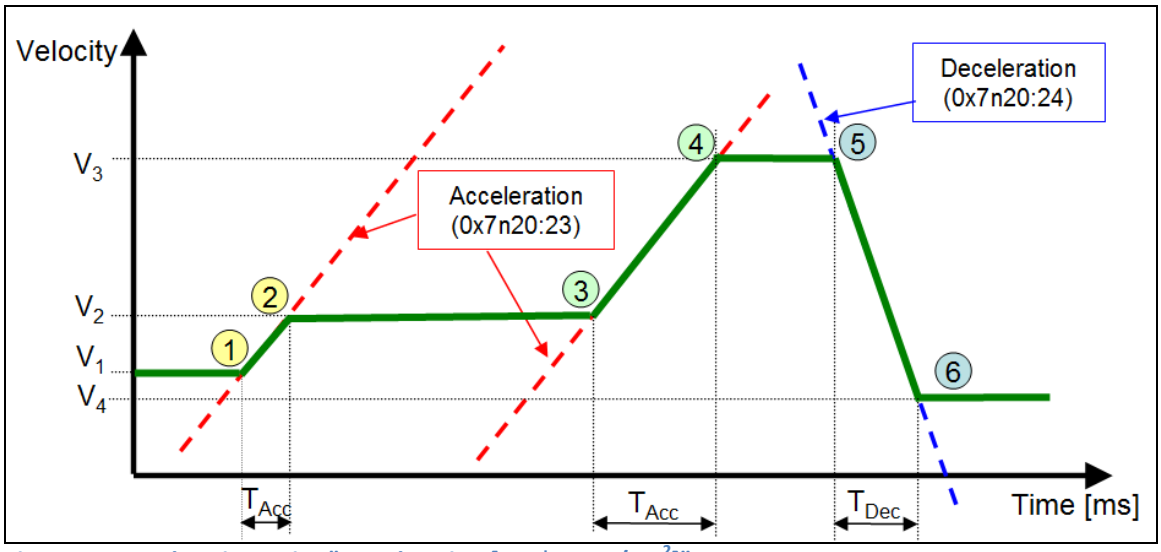


Figure 31: Acceleration unit - " Acceleration [128*µStep/sec^2]"

1 → 2

Accelerate from the current "V_1" to the target "V_2" (0x7n20:21):

- Acceleration (0x7n20:23): Acceleration [128* μ Step/sec2] from "V_1" to target "V_2"
- $T_{Acc} = |V_2 - V_1| / (Acceleration(0x7n20:23) * 128)$
- $T_{DecStop} = |V_2| / (Deceleration(0x7n20:24) * 128)$

3 → 4

Accelerate from "V_2" to the target "V_3" (0x7n20:21):

- Acceleration (0x7n20:23): Acceleration [128* μ Step/sec2] from "V_2" to target "V_3"
- $T_{Acc}=|V_3-V_2|/(Acceleration(0x7n20:23) *128)$
- $T_{DecStop}=|V_3|/(Deceleration(0x7n20:24) *128)$

5 → 6

Decelerate from the current "V_3" to "V_4" (0x7n20:21):

- Deceleration (0x7n20:24): Deceleration [128* μ Step/sec2] from "V_3" to V_4.
- $T_{Dec}= (V_3- V_4)/(Acceleration (0x7n20:23)*128)$
- $T_{DecStop}=|V_4|/(Deceleration(0x7n20:24) *128)$

6.2.1.4 Acceleration time from Vcurrent to Vtarget

In this mode the parameters "Acceleration" (0x7n20:23) and "Deceleration" (0x7n20:24) sets the time [ms] to accelerate or decelerate to a new target velocity "Velocity" (0x7n20:21). The steepness of the acceleration/deceleration gradient depends on the "Acceleration" (0x7n20:23), "Deceleration" (0x7n20:24) and the velocity difference between the current velocity and new target velocity "Velocity" (0x7n20:21) as shown in Figure 32. This acceleration unit can only be used together with the acceleration type 0 "Acceleration-->Start phase & Deceleration-->Stop Phase" (Table 14).

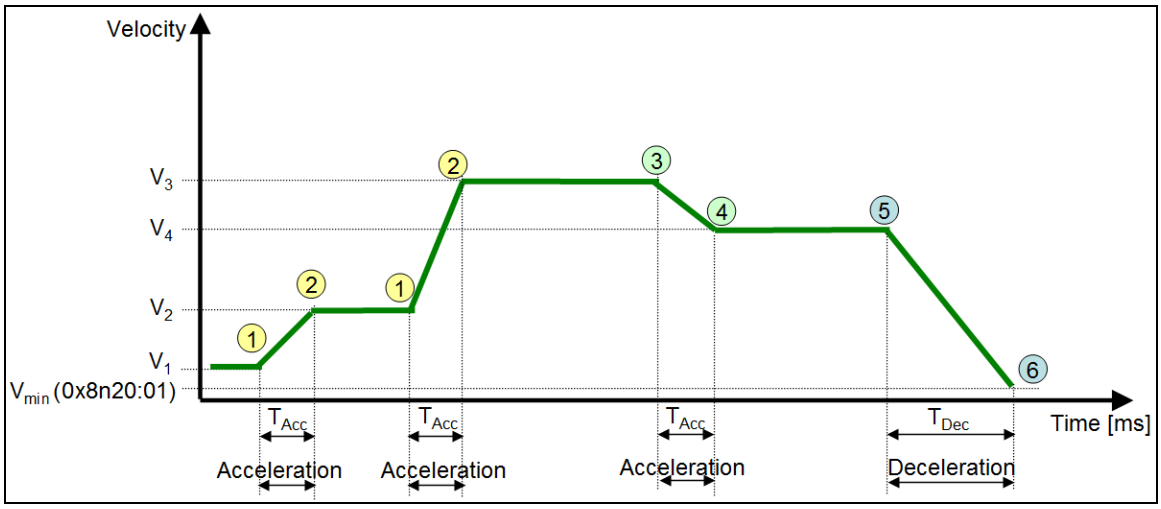


Figure 32: Acceleration unit - " Acceleration time [ms] from Vcurrent to Vtarget"

1 → 2

Accelerate from the current velocity $V_{Current}$ to the new target velocity V_{Target} (0x7n20)

- The "Acceleration" (0x7n20:23) parameter represents the time T_{acc} [ms] to accelerates the axis from $V_{Current}$ to target V_{Target}
- T_{Acc}:
  - $T_{Acc} = |V_{Target} - V_{Current}| / Acc$
  - If the "Acceleration" value remains constant then the acceleration gradient changes with the difference between current and target velocity.
- $T_{DecStop}$:
  - $T_{DecStop} = |V_{Target} - V_{min}| / Acc$



Decelerate from the current velocity $V_{Current}$ to the new target velocity V_{Target} (0x7n20)

- This mode only supports the acceleration type 0 (Table 13) therefore the acceleration time "Acceleration" (0x7n20:23) is being used for deceleration to the new target velocity
- "Acceleration" (0x7n20:23) = Deceleration time [ms] from $V_{Current}$ to V_{Target} (0x7n20)
- T_{Acc}:
  - $T_{Acc} = |V_4 - V_3| / Acc$
- $T_{DecStop}$:
  - $T_{DecStop} = |V_4 - V_{min}| / Acc$



Decelerate from the current velocity $V_{Current}$ to the "Velocity min" (0x8n20:01) V_{min}.

- The "Deceleration" (0x7n20:24) parameter represents the deceleration time T_{Dec} [ms] from $V_{Current}$ to V_{min} (0x8n20:01).
- T_{Dec}:
  - $T_{Dec} = (V_4 - V_{min}) / Dec$

6.2.2 Acceleration/Deceleration Type

The acceleration/deceleration type describes which parameters are being used for accelerating/decelerating the motor to the target velocity. Two types are defined (Table 13).

Name	Value
Acceleration-->Start phase & Deceleration-->Stop Phase	0
Acceleration-->Acceleration & Deceleration--> Deceleration	1

Table 13: DT0815EN01 object

Type 1 requires considerable firmware calculation and therefore to increase the ECAT-2094S response time it is strongly suggested to just use type 0.

The following section describes the properties of the acceleration/deceleration types.

6.2.2.1 Start-Stop Phase Type

The acceleration (0x7n20:23) setting is being used for the acceleration phase (1->2, 5->6) and the deceleration (0x7n20:24) value for the deceleration to stop phase (3->4, 7->8). When changing the direction of rotation the ECAT-2094S will first decelerate to stop using the deceleration (0x7n20:24) value and then accelerate to the new target velocity by using the acceleration (0x7n20:23) value.

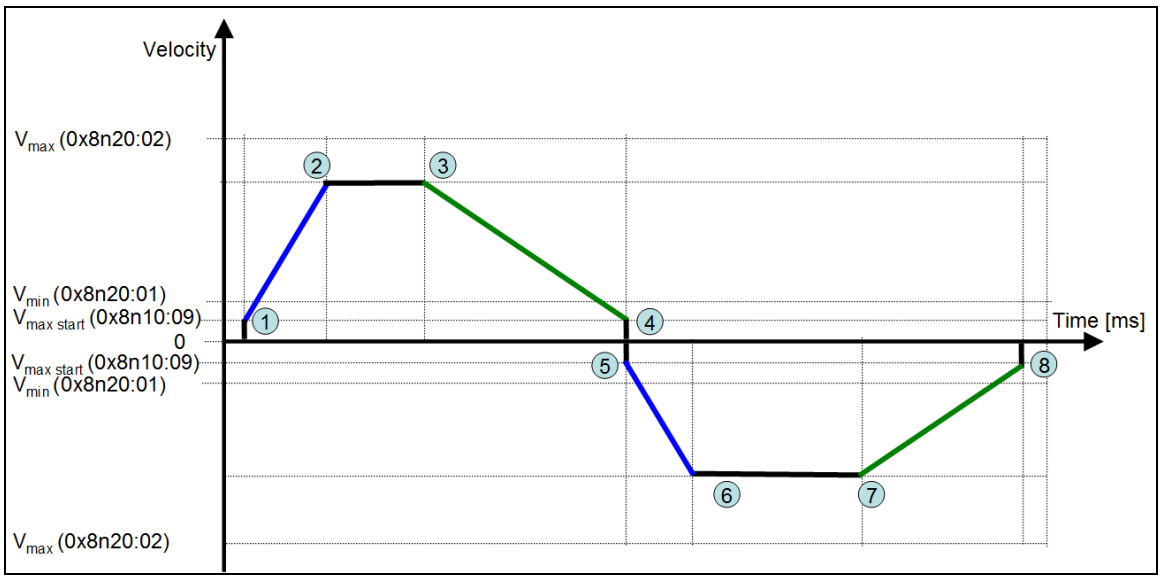


Figure 33: Acceleration - deceleration behavior for the start-stop phase

When changing the velocity on the fly the acceleration (0x7n20:23) value determines the gradient of accelerate or decelerate curve for reaching the new target velocity (Figure 34).

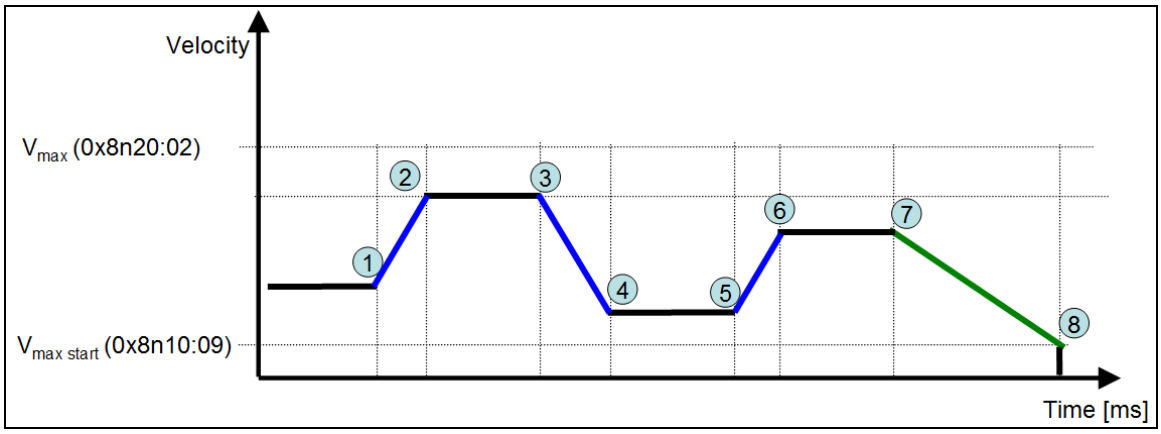


Figure 34: Change velocity on the fly for Start-Stop phase mode

The following description refers to Figure 34:

- The acceleration (0x7n20:23) value sets the accelerating (1->2, 5->6) and the decelerating (3->4) property to reach the new velocity.
- The deceleration (0x7n20:24) value determines the deceleration to stop section (7->8).

6.2.2.2 Standard Acceleration/Deceleration

Here the acceleration (0x7n20:23) setting are always applied when the a acceleration takes place. This is the case when the motor is accelerated from stand still or when the target velocity is greater than the current driving velocity. The deceleration (0x7n20:24) setting determines the gradient of the deceleration curve regardless whether the motor decelerates to stop or decelerates to a new velocity. The acceleration/deceleration for changing the velocity direction is shown in Figure 33.

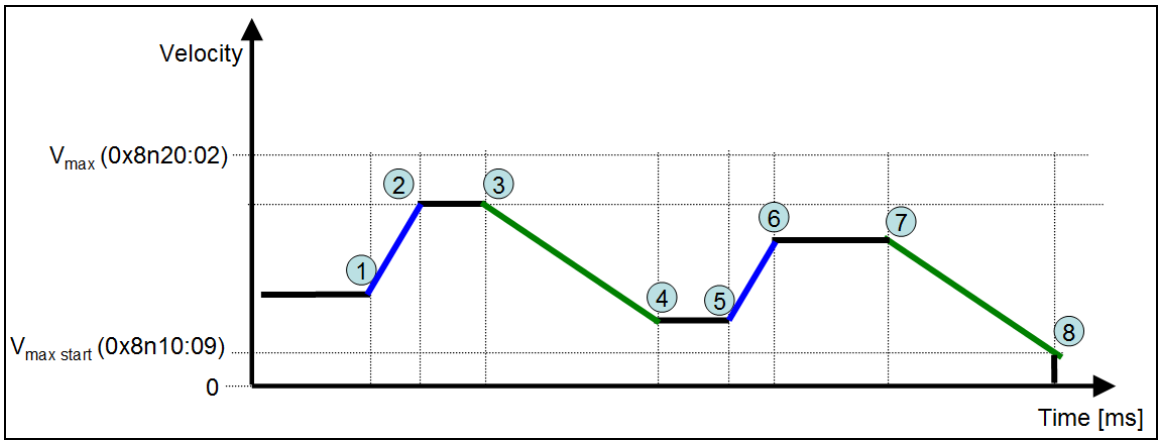


Figure 35: Changing velocity on the fly for "standard acceleration-deceleration"

Figure 35 shows the acceleration/deceleration curve for changing the velocity while the

motor is busy moving to the target position:

- The acceleration (0x7n20:23) value sets the acceleration gradient (1->2, 5->6).
- The deceleration (0x7n20:24) value sets the deceleration gradient (3->4, 7->8).

This acceleration type is not being supported by the acceleration unit "Vcurrent to Vtarget"

6.3 Positioning Interface Compact

In the following the procedure for executing a travel command in "Positioning interface compact" mode is being described.

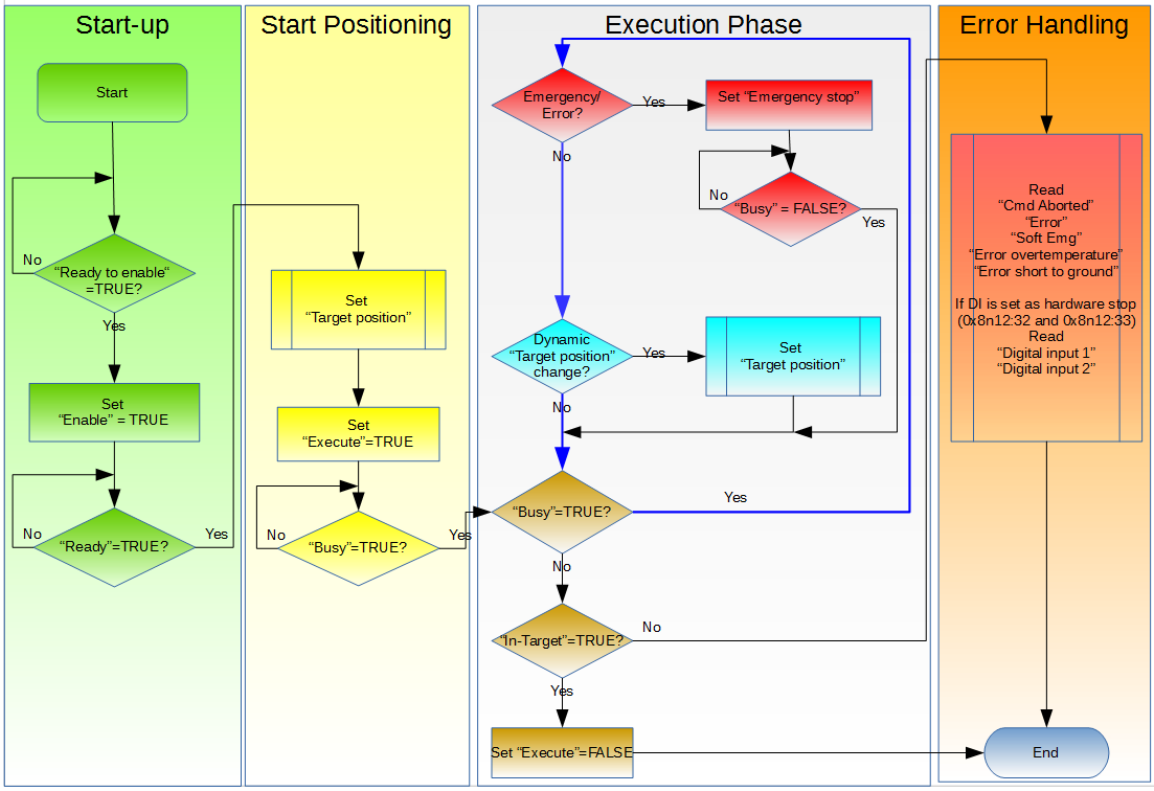


Figure 36: "Positioning interface compact" setting sequence

Operating procedure:

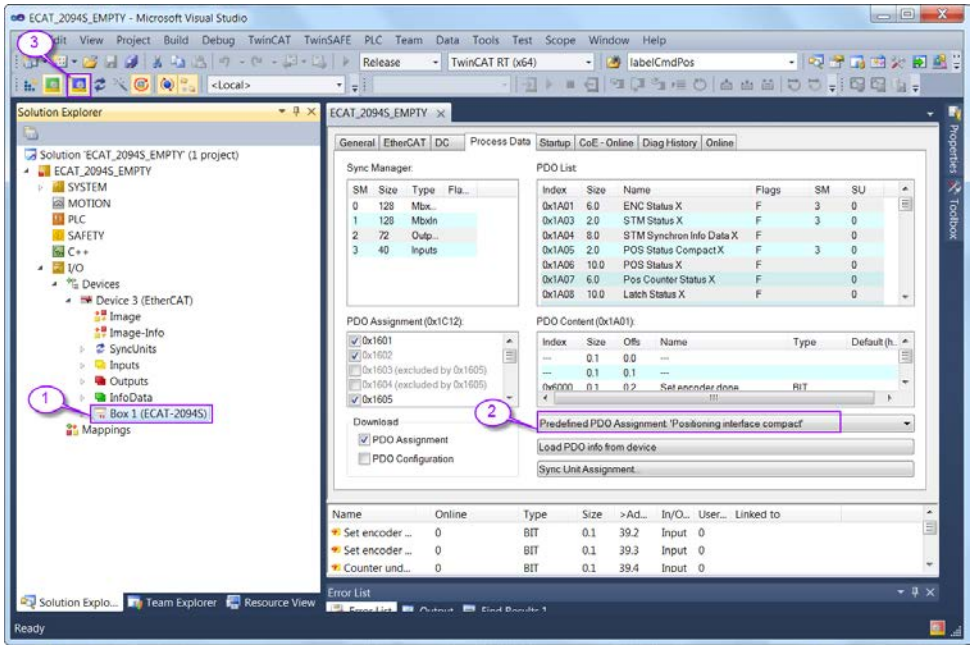
Step 1: PDO assignment

Select the function "Positioning interface compact" from the predefined PDO assignment selection box in the lower part of the "Process data" tab. This causes TwinCAT to automatically activated all necessary PDOs and deactivate the

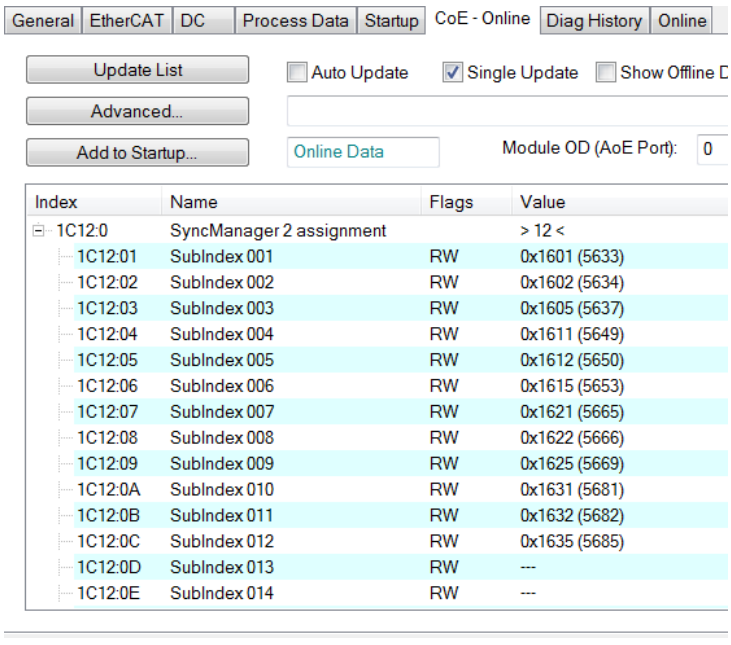
unnecessary ones.

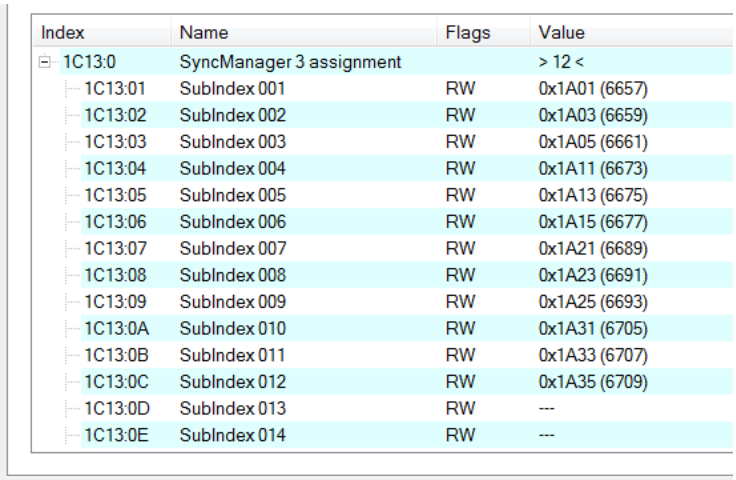
Procedure:

1. Select the "Process Data" tab of the ECAT-2094S
2. Select the "Predefined PDO Assignment: "Position interface compact" from the combo box
3. Send the PDO assignment to the slave by clicking the "Reload I/O device" button



The SyncManager 2 and 3 in the "CoE-Online" tab displays the new PDO assignment:





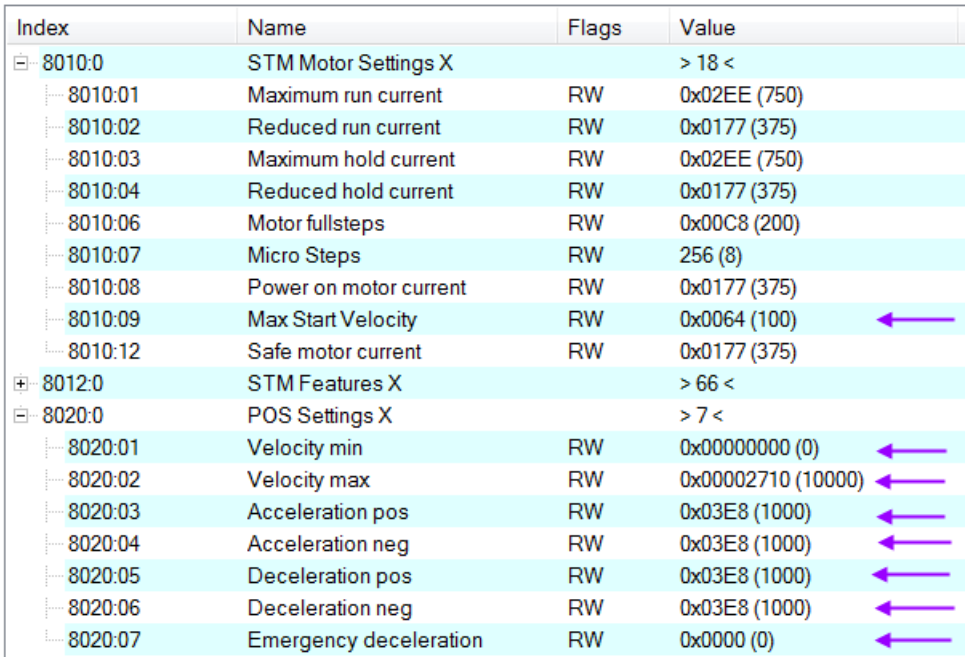
Index	Name	Flags	Value
1C13:0	SyncManager 3 assignment		> 12 <
1C13:01	SubIndex 001	RW	0x1A01 (6657)
1C13:02	SubIndex 002	RW	0x1A03 (6659)
1C13:03	SubIndex 003	RW	0x1A05 (6661)
1C13:04	SubIndex 004	RW	0x1A11 (6673)
1C13:05	SubIndex 005	RW	0x1A13 (6675)
1C13:06	SubIndex 006	RW	0x1A15 (6677)
1C13:07	SubIndex 007	RW	0x1A21 (6689)
1C13:08	SubIndex 008	RW	0x1A23 (6691)
1C13:09	SubIndex 009	RW	0x1A25 (6693)
1C13:0A	SubIndex 010	RW	0x1A31 (6705)
1C13:0B	SubIndex 011	RW	0x1A33 (6707)
1C13:0C	SubIndex 012	RW	0x1A35 (6709)
1C13:0D	SubIndex 013	RW	---
1C13:0E	SubIndex 014	RW	---

Step 2: Set the motor torque (see "Positioning interface", chapter 6.2 Step 2:)

Step 3: Set the number of micro-steps per full step (8n10:07). The motor runs smoother and with less vibration with higher micro-steps value setting.

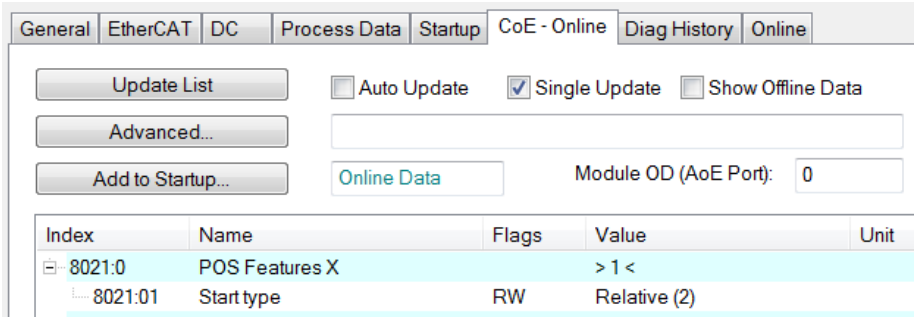
Step 4: Set the motion parameters for the system: max velocity, max acceleration, etc.

1. The velocity has to be set according the description of "Positioning interface" (chapter 6.2 Step 4:)
2. Set the "Acceleration unit" (0x8n20:08) and "Acc-Dec parameter definition" (0x8n20:09) which are described in chapter 6.2.1 and 6.2.2.
3. Set the acceleration and deceleration parameter values. Their units are determined by the "Acceleration unit" (0x8n20:08). Values for both the positive and negative direction are required.
   i. Acceleration pos (0x8n20:03): Acceleration in the positive direction of rotation.
   ii. Acceleration neg (0x8n20:04): Acceleration in the negative direction of rotation
   iii. Deceleration pos (0x8n20:05): Deceleration in the positive direction of rotation
   iv. Deceleration neg (0x8n20:06): Deceleration in the negative direction of rotation
4. "Emergency deceleration" describes how fast the motor stops after the emergency stop flag has been set to TRUE (0x7n20:02 - Emergency stop)

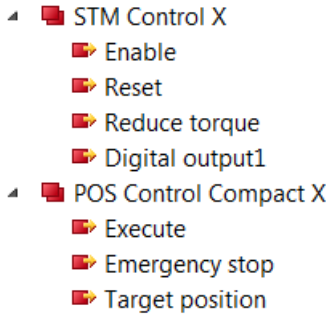


Index	Name	Flags	Value
8010:0	STM Motor Settings X		> 18 <
8010:01	Maximum run current	RW	0x02EE (750)
8010:02	Reduced run current	RW	0x0177 (375)
8010:03	Maximum hold current	RW	0x02EE (750)
8010:04	Reduced hold current	RW	0x0177 (375)
8010:06	Motor fullsteps	RW	0x00C8 (200)
8010:07	Micro Steps	RW	256 (8)
8010:08	Power on motor current	RW	0x0177 (375)
8010:09	Max Start Velocity	RW	0x0064 (100)
8010:12	Safe motor current	RW	0x0177 (375)
8012:0	STM Features X		> 66 <
8020:0	POS Settings X		> 7 <
8020:01	Velocity min	RW	0x00000000 (0)
8020:02	Velocity max	RW	0x00002710 (10000)
8020:03	Acceleration pos	RW	0x03E8 (1000)
8020:04	Acceleration neg	RW	0x03E8 (1000)
8020:05	Deceleration pos	RW	0x03E8 (1000)
8020:06	Deceleration neg	RW	0x03E8 (1000)
8020:07	Emergency deceleration	RW	0x0000 (0)

Step 5: Set the start type. The "Start type" (0x8n21:01) describes whether the target position is a relative or absolute position. In addition the user can determine whether the target position can be changed on the fly. Consult Table 9 for the correct parameter value.



Step 6: Motion execution procedure:



1. Activate the "Enable" (0x7n10:01) flag
2. Set the target position (unit: steps) (0x7n20:11). The target position distance is being defined by the "Start type" (0x8n21:01) configuration.
3. Start motion execution by setting the "Execute"-variable to true

(0x7n20:01).

4. If an emergency stop (0x7n20:02) has been activated during driving, then the "Emergency stop"-variable has to be set to false and "Execute" back to false before the next command can be executed
5. Error: If an error occurred during driving (overheating, EtherCAT communication failed, Master sets slave from OP to none OP mode, etc.) the error flag is activated (0x6n10:04 Error). In order to clear this flag the "Reset" variable has to be activated (0x7n10:02 - Reset) for one cycle time.

6.4 Position Control

Position control mode has to be selected if the application program needs to sent a new absolute target position in every communication cycle. The maximum velocity and the acceleration time have to be set at a high value in order for the driver to reach the new target position at the end of the cycle time. In this mode the application program basically calculates and control the velocity profile of the motor.

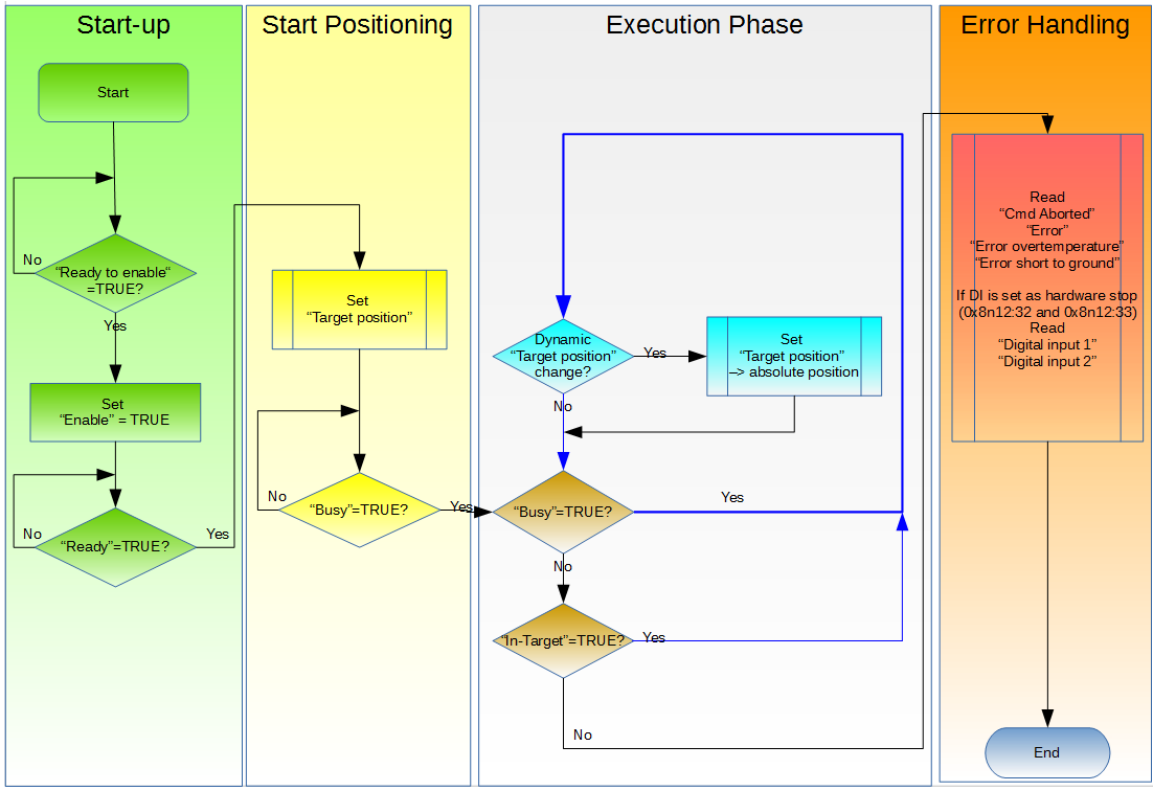


Figure 37: Variable execution sequence for the position control mode

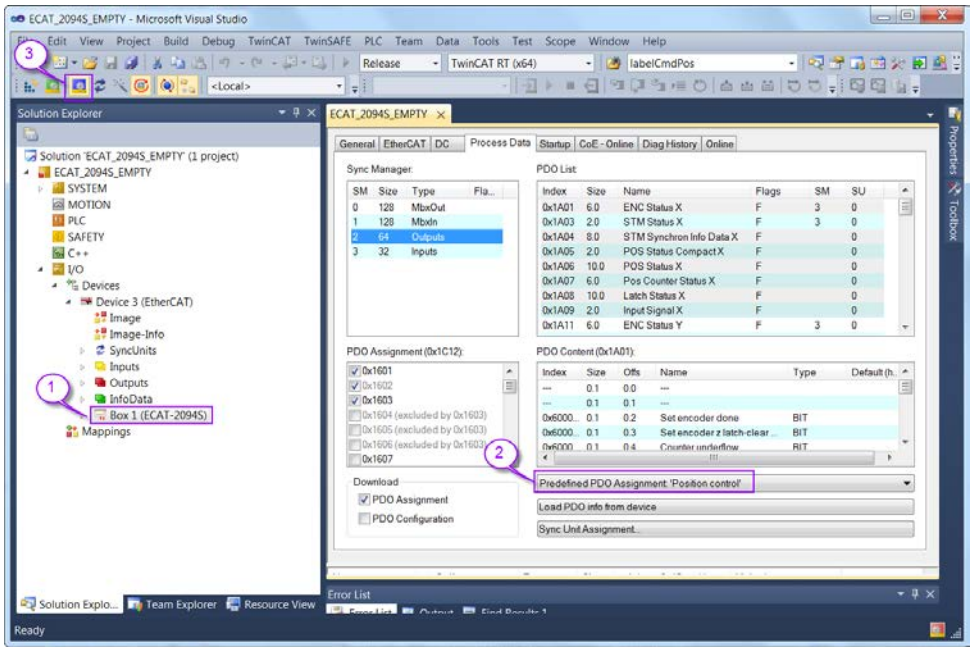
Operating procedure:

Step 1: PDO assignment

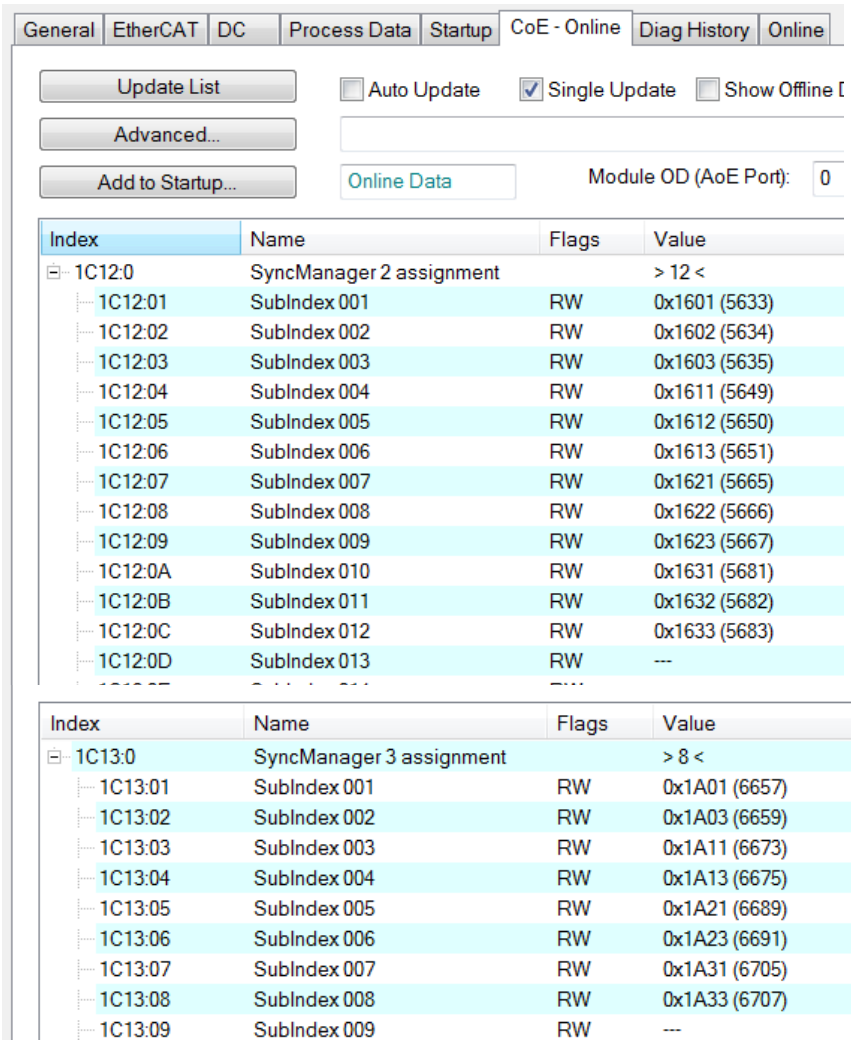
Select the function "Positioning control" predefined PDO assignment in the lower part of the Process data tab. As a result, all necessary PDOs are automatically activated and the unnecessary PDOs are deactivated.

Procedure:

1. Select the "Process Data" tab of the ECAT-2094S
2. Select the "Predefined PDO Assignment: "Position control" from the combo box
3. Send the PDO assignment to the slave by clicking the "Reload I/O device" button



The SyncManager 2 and 3 in the "CoE-Online" tab displays the new PDO assignment:



Step 2: Set the motor torque (see "Positioning interface", chapter 6.2 Step 2:)

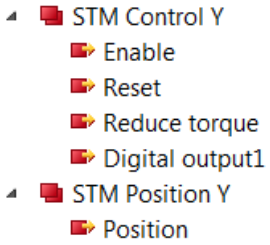
Step 3: Set the number of micro-steps per full step (8n10:07). The motor runs smoother and with less vibration with higher micro-steps value setting.

Step 4: Set the motion parameters for the system: max velocity, max acceleration, etc.

1. The velocity has to be set as described for the "Positioning interface" (chapter 6.2 Step 4:)
2. Set the "Acceleration unit" (0x8n20:08) and "Acc-Dec parameter definition" (0x8n20:09) which are described in chapter 6.2.1 and 6.2.2.
3. In addition the acceleration and deceleration values (unit: 0x8n20:08) have to be set. The values for both the positive and negative directions are required.
   i. Acceleration pos (0x8n20:03)
   ii. Acceleration neg (0x8n20:04)
   iii. Deceleration pos (0x8n20:05)
   iv. Deceleration neg (0x8n20:06)

Index	Name	Flags	Value
8010:0	STM Motor Settings X		> 18 <
8010:01	Maximum run current	RW	0x02EE (750)
8010:02	Reduced run current	RW	0x0177 (375)
8010:03	Maximum hold current	RW	0x02EE (750)
8010:04	Reduced hold current	RW	0x0177 (375)
8010:06	Motor fullsteps	RW	0x00C8 (200)
8010:07	Micro Steps	RW	256 (8)
8010:08	Power on motor current	RW	0x0177 (375)
8010:09	Max Start Velocity	RW	0x0064 (100)
8010:12	Safe motor current	RW	0x0177 (375)
8012:0	STM Features X		> 66 <
8020:0	POS Settings X		> 7 <
8020:01	Velocity min	RW	0x00000000 (0)
8020:02	Velocity max	RW	0x00002710 (10000)
8020:03	Acceleration pos	RW	0x03E8 (1000)
8020:04	Acceleration neg	RW	0x03E8 (1000)
8020:05	Deceleration pos	RW	0x03E8 (1000)
8020:06	Deceleration neg	RW	0x03E8 (1000)
8020:07	Emergency deceleration	RW	0x0000 (0)

Step 5: Motion execution procedure:



1. Activate the "Enable" (0x7n10:01) parameter
2. Set the absolute target position (unit: steps). The driver will output steps as soon as the actual and target position are not identical. Set the acceleration time and velocity (Step 4:) to a high value if the application if the motor needs to reach the target position at the end of each cycle.
3. Error: If an error occurred during driving (overheating, EtherCAT communication failed, Master sets slave from OP to none OP mode, etc.) the error flag is activated (0x6n10:04 Error). In order to clear this flag the "Reset" variable has to be activated (0x7n10:02 - Reset) for one cycle time.

7 Velocity Control Setting

In velocity control mode the motor accelerates to the target velocity and keeps running at this velocity until the user changes the velocity. When the user changes the velocity setting the controller will automatically accelerate/decelerate to the new value. In case of a rotation direction change the driver first slows the motor down to standstill before accelerating in the opposite direction. The motor will stop if the speed is set to zero. The acceleration and deceleration values have to be set via the configuration objects (0x8n20).

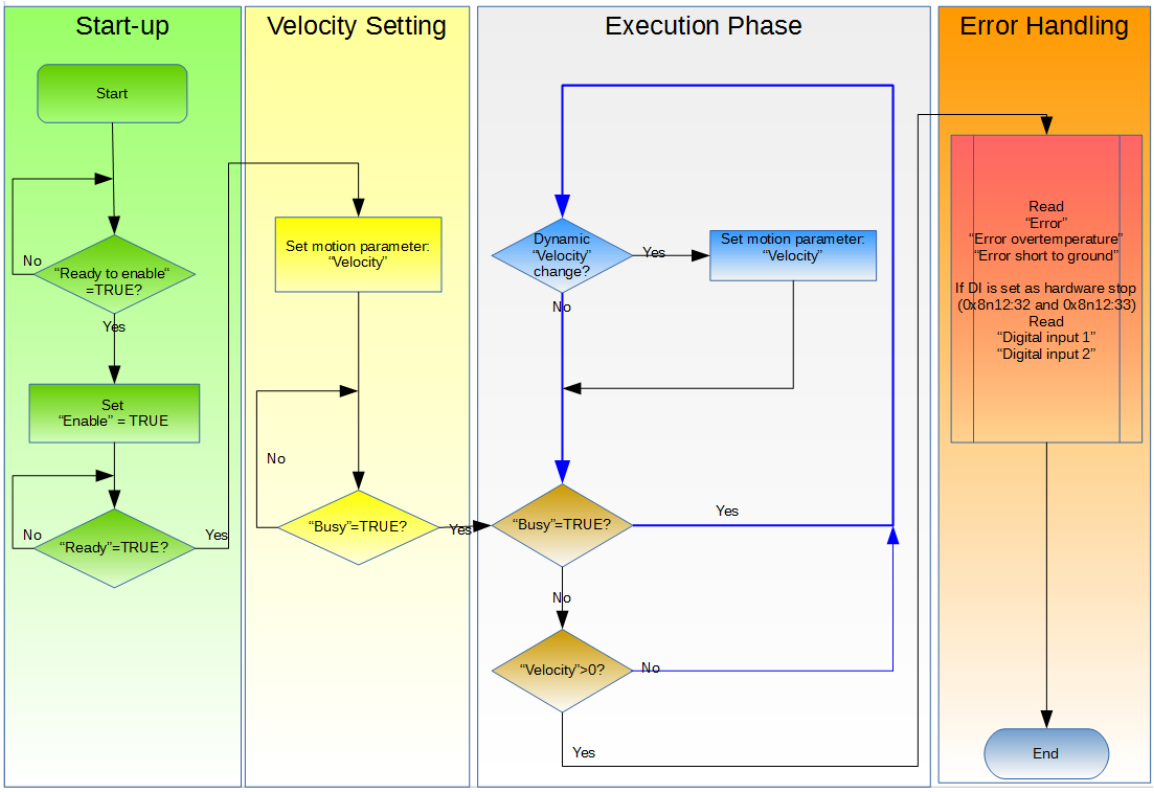


Figure 38: Velocity control settings

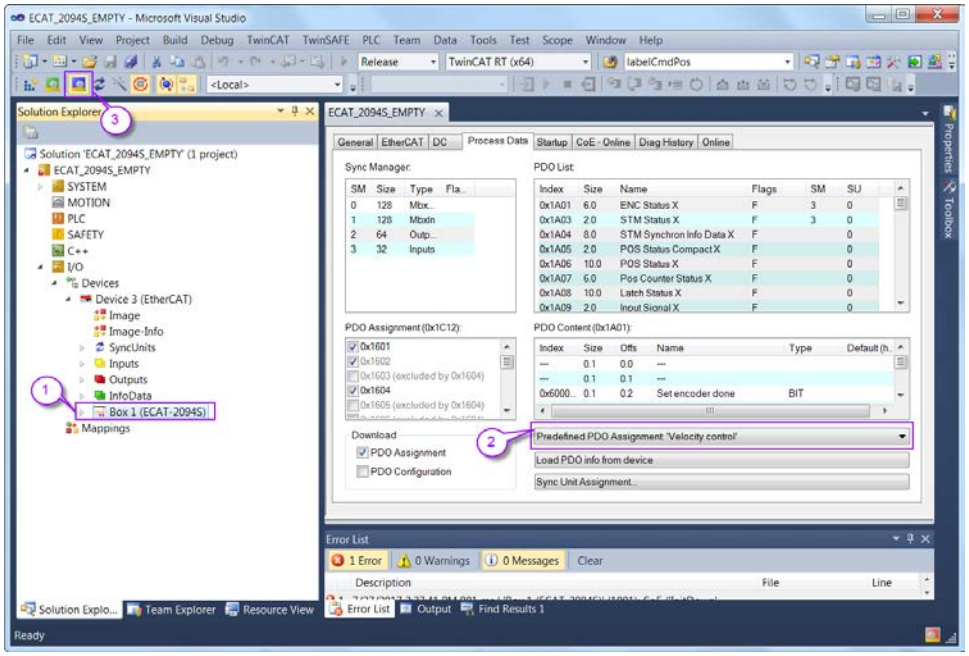
Procedure for the velocity control operation:

Step 1: PDO assignment

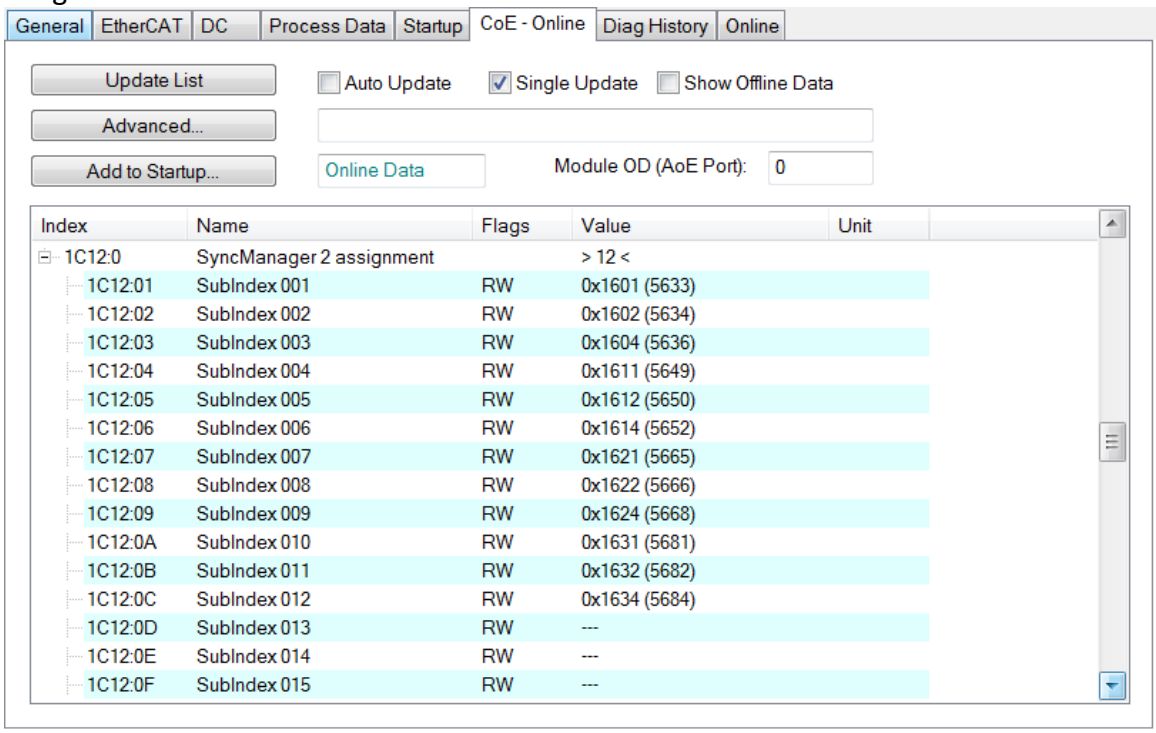
Select the function "Velocity control" predefined PDO assignment selection list in the lower part of the "Process data" tab:

1. Select the "Process Data" tab of the ECAT-2094S
2. Select the "Predefined PDO Assignment: " Velocity control " from the combo box
3. Send the PDO assignment to the slave by clicking the "Reload I/O device"

button



The SyncManager 2 and 3 in the "CoE-Online" tab displays the new PDO assignment:



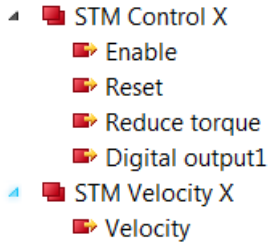
Index	Name	Flags	Value	Unit
1C13:0	SyncManager 3 assignment		> 8 <	
1C13:01	SubIndex 001	RW	0x1A01 (6657)	
1C13:02	SubIndex 002	RW	0x1A03 (6659)	
1C13:03	SubIndex 003	RW	0x1A11 (6673)	
1C13:04	SubIndex 004	RW	0x1A13 (6675)	
1C13:05	SubIndex 005	RW	0x1A21 (6689)	
1C13:06	SubIndex 006	RW	0x1A23 (6691)	
1C13:07	SubIndex 007	RW	0x1A31 (6705)	
1C13:08	SubIndex 008	RW	0x1A33 (6707)	
1C13:09	SubIndex 009	RW	---	
1C13:0A	SubIndex 010	RW	---	
1C13:0B	SubIndex 011	RW	---	
1C13:0C	SubIndex 012	RW	---	
1C13:0D	SubIndex 013	RW	---	
1C13:0E	SubIndex 014	RW	---	
1C13:0F	SubIndex 015	RW	---	

Step 2: Set the motor torque (see "Positioning interface", chapter 6.2 Step 2:)

Step 3: Set the number of micro-steps per full step (8n10:07). The motor runs smoother and with less vibration with higher micro-steps value setting.

Step 4: Set the motion parameters. Follow the steps described for the position control (chapter 6.2 Step 4:)

Step 5: Motion execution procedure:



1. Activate the "Enable" (0x7n10:01) parameter
2. Set the velocity (unit: step/second). The driver will immediately accelerate the motor to the set speed and continuously run at this speed until a new speed has been received. The motor will stop if the speed is set to zero or the "Enable" (0x7n10:01) flag has been put to FALSE or an error occurred.

8 CoE Interface

8.1 General Description

The CoE interface (CANopen over EtherCAT) is used for parameter management of EtherCAT devices. The CoE interface displays all the objects and parameters which are required for operating and diagnosing the ECAT-2094S device. Some parameters are fixed and can not be modified, they for example indicate the operating status of the device or the device properties. Motion related parameter need to be set before the actual motion control starts. These parameter setting are determined by the controlled stepper motor type and the setup of the motion application system.

CoE parameters has to be accessed via the CAN over EtherCAT protocol. The EtherCAT master accesses the local CoE lists of the slaves via CAN over EtherCAT. The user does not need to understand the CoE protocol when using the TwinCAT System Manager for CoE parameter configuration.

The CoE parameter describe a wide range of features such as manufacturer ID, device name, process data settings, calibration values for the stepper motor such as the current output, microsteps per full step, maximum velocity, etc..

The relevant ranges of the CoE list are:

- 0x1000: Stores fixed information of the device, including name, manufacturer, serial number etc.. In addition stores information about the current and available process data configurations.
  - 0x1600: RxPDO mapping
  - 0x1A00: TxPDO mapping
- 0x8000: Stores all the configuration data which are required for the stepper motor control.
- 0x6000: Input PDOs ("input" from the perspective of the EtherCAT master)
- 0x7000: Output PDOs ("output" from the perspective of the EtherCAT master)

The Figure 39 shows some of the CoE objects available for the ECAT-2094S device, ranging from 0x1000 to 0xF008. The parameters of the objects can be accessed by expanding the tree in the "CoE-Online" tab.

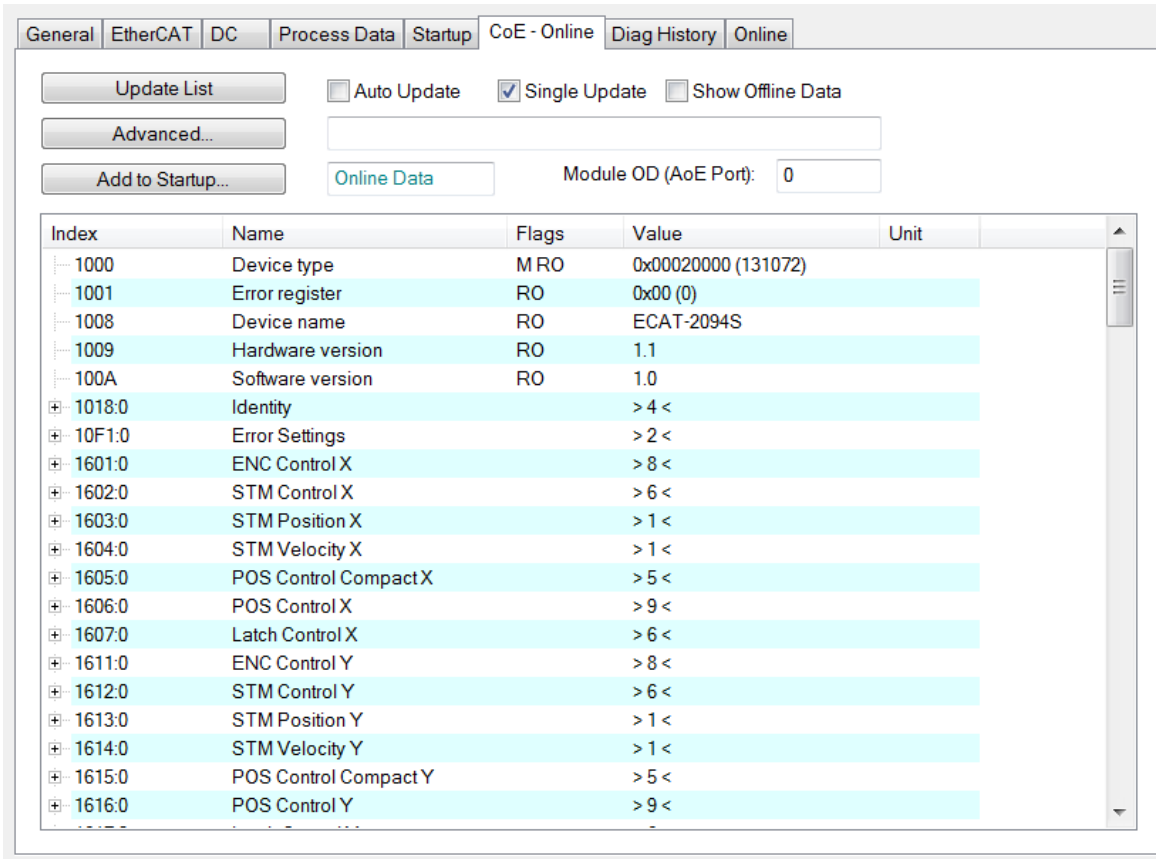


Figure 39: "CoE - Online " tab

8.2 Save Configuration Data to Memory

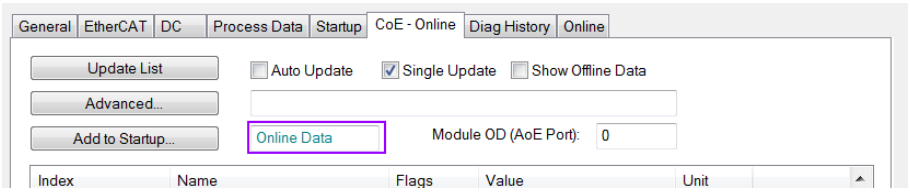
In this section the procedure of saving motion configuration parameters to the device non-volatile memory is being discussed.

The CoE object range 0x8000 to 0x8321 contains all the motion related parameters which are configurable and storable. TwinCAT allows the user to set the configuration parameters via the System Manager (Figure 39) or from a TwinCAT PLC via ADS (TcEtherCAT.lib library).

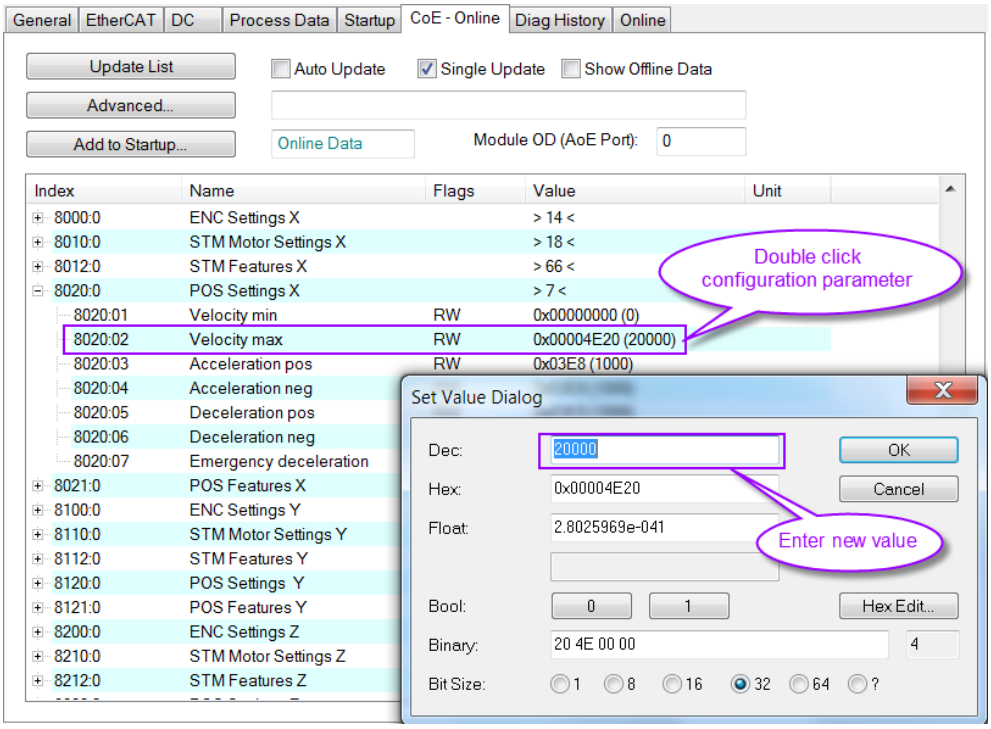
If slave CoE parameters are modified online, the ECAT-2094S device does not automatically store the data to a non-volatile memory. The data are lost if the device is switched off. The 0xF008 object provides functions to store the modified configuration data to the non-volatile memory of the device and the setting will be immediately available after a restart.

Procedure for storing configuration data to the local ECAT-2094S memory:

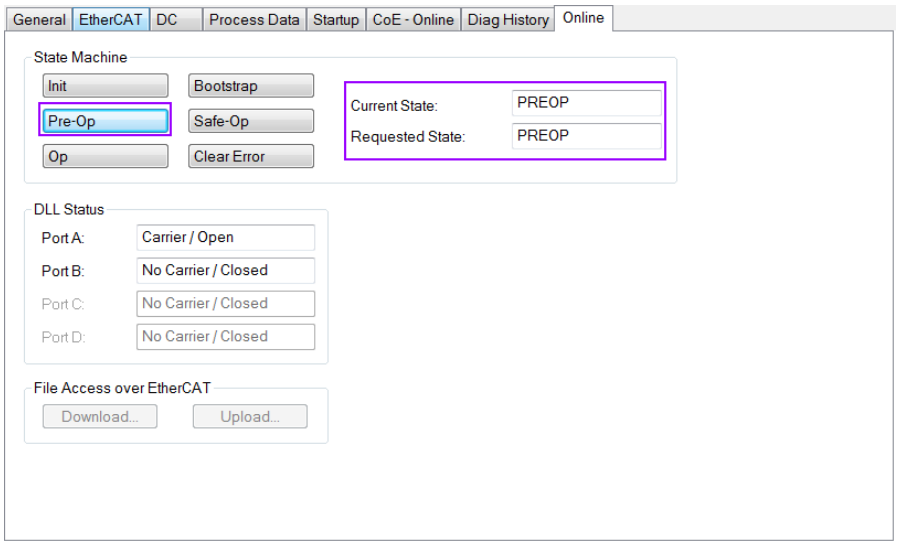
Step 1: Make sure the TwinCAT System Manager is connected to the ECAT-2094S and the "CoE-Online" tab is showing that the slave is online.



Step 2: Set all the necessary configuration objects (0x8n00, 0x8n10, 0x8n12, 0x8n21) for the X, Y, Z and U motor (n=0 to 3). Setting is being done by double clicking the configuration parameter and entering a new value in the popup window. In the following picture the maximum velocity of motor X is being set to 20000 steps/second.

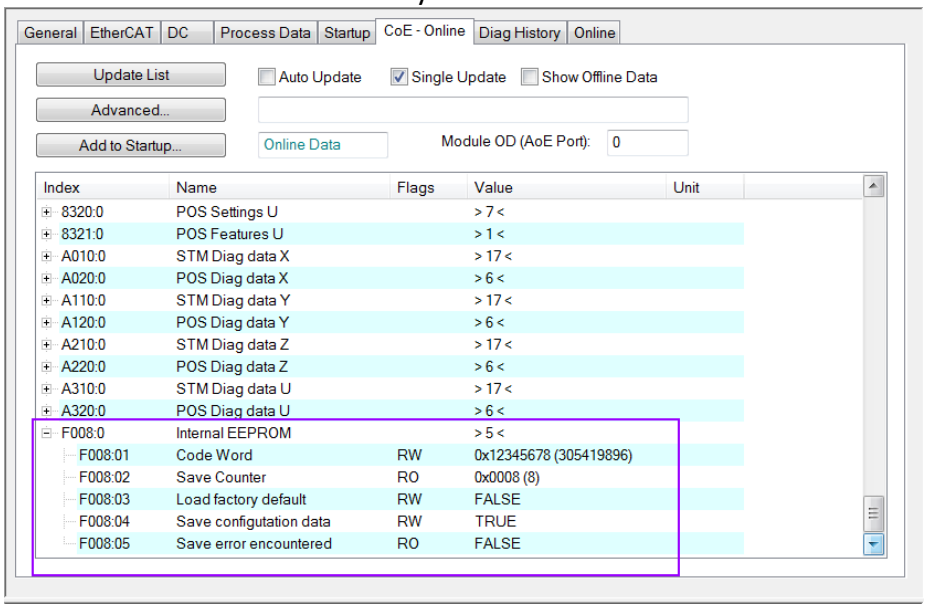


Step 3: After all the configurations have been done set the slave into Pre-Op mode. Data can only be stored to the local device if it is in Pre-Op mode. On the "Online" tab click the "Pre-OP" button to put the slave into Pre-OP mode.

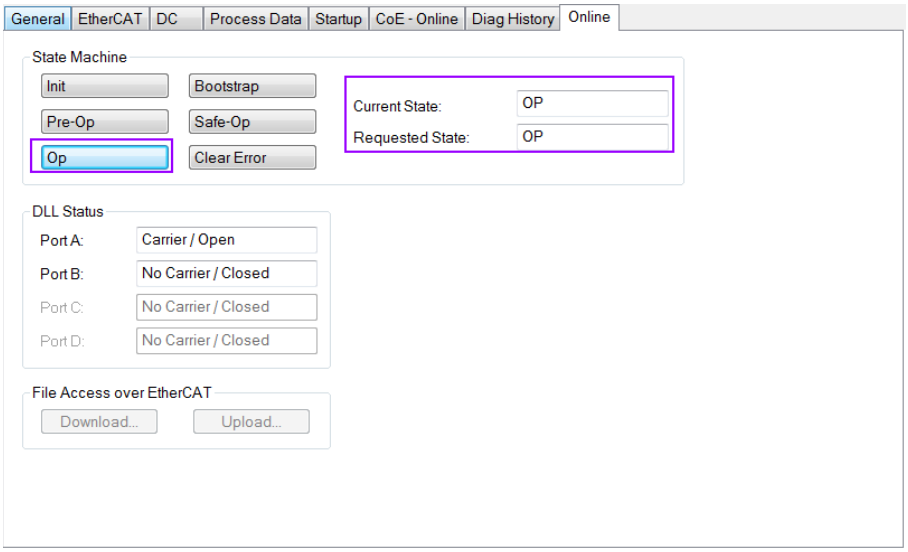


Step 4: The parameters of the 0xF008 object handles the save procedure.

1. Scroll to the end of the "CoE-Online" list and expand the tree view of the 0xF008 object
2. Enter the value 0x12345678 for the "Code Word"
3. Set the "Save configuration data" from FALSE to TRUE in order to save the configuration data to the internal EEPROM. The parameter "Save error encountered" (F008:05) indicates whether an error occurred during save process.
4. In order for the user configuration data to take effect after device restart set the "Load factory default" to FALSE. It is always possible to return to the factory default setting by setting this value back to TRUE.
5. The "Save Counter" (F008:02) shows how often configuration data has been stored to the local memory in the lifetime of the device.



Step 5: Set the ECAT-2094S back into OP mode.



8.3 Driver Tuning

The ECAT-2094S uses the motion control chip TMC5130A from Trinamic to do stepper motor control. The following TMC5130A modes are being supported by the ECAT-2094S:

- stealthChop™ No-noise, high-precision chopper algorithm for inaudible motion and inaudible standstill of the motor.
- spreadCycle™ High-precision chopper algorithm available as an alternative to the traditional constant off-time algorithm.
- dcStep™ Load dependent speed control. The motor moves as fast as possible and never loses a step.
- coolStep™ Load-adaptive current control which reduces energy consumption by as much as 75%.

By default the ECAT-2094S uses the spreadCycle™ mode. This mode is simple to use and very precise as the chopper algorithm of the TMC5130A automatically determines the optimum length for the fast-decay phase. The spreadCycle will provide superior microstepping quality even with default settings.

If additional driver tuning is required for the spreadCycle mode or for the other modes listed above then

- consult the "TMC5130_datasheet.pdf" manual which can be downloaded from the Trinamic website. The "Quick Configuration Guide" chapter provides steps for tuning the driver. The Vendor Specific Register Definitions (chapter 11) shows through which Service Data Object (SDO) of the ECAT-2094S the motion chip register can be directly accessed.
- contact ICP DAS CO., LTD. (http://www.icpdas.com) to get additional information and support.

9 Object Description and Parameterization

9.1 Standard Objects

Index 1000 Device type

Index (hex)	Name	Description	Data type	Flags	Default
1000:0	Device type	Device type of the EtherCAT slave	UINT32	RO	0x00020000

Index 1008 Device name

Index (hex)	Name	Description	Data type	Flags	Default
1008:0	Device name	Device name of the EtherCAT slave	STRING	RO	ECAT-2094S

Index 1009 Hardware version

Index (hex)	Name	Description	Data type	Flags	Default
1009:0	Hardware version	Hardware version of the EtherCAT slave	STRING	RO	2.0 (or greater)

Index 100A Software version

Index (hex)	Name	Description	Data type	Flags	Default
100A:0	Software version	Software version of the EtherCAT slave	STRING	RO	2.0.0 (or greater)

Index 1018 Identity

Index (hex)	Name	Description	Data type	Flags	Default
1018:0	Identity		UINT8	RO	0x04
1018:01	Vendor ID	Vendor ID of the EtherCAT slave	UINT32	RO	0x00494350
1018:02	Product code	Product code of the EtherCAT slave	UINT32	RO	0x00209453
1018:03	Revision	Revision number of the EtherCAT slave	UINT32	RO	0x00010000
1018:04	Serial number	Serial number of the EtherCAT slave (not supported)	UINT32	RO	0x00000000

Index 10F1 Error settings

Index (hex)	Name	Description	Data type	Flags	Default
10F1:0	Error settings		UINT8	RO	0x02
10F1:01	Local error reaction	Not implemented	UINT32	RW	0x00000001
10F1:02	Sync error counter limit	For DC mode only: The Sync Error Counter is incremented with every missing Sync Management Event by three and decremented by one if an event is	UINT16	RW	0x0004

Index (hex)	Name	Description	Data type	Flags	Default
		received. If the Sync Error Counter exceeds this limit the system changes into the SAFEOP state with the 'Synchronization Lost' error. The Sync Error Counter is reset when the error was acknowledged.			

9.2 RxPDO Mapping Objects

The symbol **n** represents the axis number: 0 to 3.

Index 16n1 ENC Control (RxPDO-Map)

Index (hex)	Name	Description	Data type	Flags	Default
16n1:0	ENC Control	Encoder control	UINT8	RO	0x08
16n1:01	Status__Gap1	empty (1 Bit)	UINT32	RO	0x00000001
16n1:02	Control__Set encoder	Set encoder (1 Bit)	UINT32	RO	0x7n000201
16n1:03	Control__Set position counter	Set position counter (1 Bit)	UINT32	RO	0x7n000301
16n1:04	Control__Set encoder z latch-clear mode	Activate the encoder index latch-clear mode (1 Bit)	UINT32	RO	0x7n000401
16n1:05	Control__Gap2	BYTE padding (4 Bit)	UINT32	RO	0x00000004
16n1:06	Control_Encoder z latch-clear mode	Encoder index clear mode (8 Bit)	UINT32	RO	0x7n000608
16n1:07	Control__Set encoder value	Set encoder value (32-bit)	UINT32	RO	0x7n001120
16n1:08	Control__Set position counter value	Set position counter value (32-bit)	UINT32	RO	0x7n001220

Index 16n2 STM Control (RxPDO-Map)

Index (hex)	Name	Description	Data type	Flags	Default
16n2:0	STM Control	Stepper motor control	UINT8	RO	0x06
16n2:01	Control__Enable	Enable	UINT32	RO	0x7n100101

Index (hex)	Name	Description	Data type	Flags	Default
		(1 Bit)			
16n2:02	Control__Reset	Reset (1 Bit)	UINT32	RO	0x7n100201
16n2:03	Control__Reduce torque	Reduce torque (1 Bit)	UINT32	RO	0x7n100301
16n2:04	Control__Gap1	BYTE padding (5 Bit)	UINT32	RO	0x00000005
16n2:05	Control__Digital output1	Digital output1 (1 Bit)	UINT32	RO	0x7n100C01
16n2:06	Control__Gap2	BYTE padding (7 Bit)	UINT32	RO	0x00000007

Index 16n3 STM Position (RxPDO-Map)

Index (hex)	Name	Description	Data type	Flags	Default
16n3:0	STM Position	Stepper motor position control	UINT8	RO	0x01
16n3:01	Control__Position	Position (32 Bit)	UINT32	RO	0x7n101120

Index 16n4 STM Velocity (RxPDO-Map)

Index (hex)	Name	Description	Data type	Flags	Default
16n4:0	STM Velocity	Stepper motor velocity control	UINT8	RO	0x01
16n4:01	Control__Velocity	Velocity (32 Bit)	UINT32	RO	0x7n102120

Index 16n5 POS Control Compact (RxPDO-Map)

Index (hex)	Name	Description	Data type	Flags	Default
16n5:0	POS Control Compact	Stepper motor compact control	UINT8	RO	0x05
16n5:01	Control__Execute	Execute (1 Bit)	UINT32	RO	0x7n200101
16n5:02	Control__Emergency stop	Emergency stop (1 Bit)	UINT32	RO	0x7n200201
16n5:03	Control__Gap1	BYTE padding (6 Bit)	UINT32	RO	0x00000006
16n5:04	Control__Gap2	WORD padding (8 Bit)	UINT32	RO	0x00000008
16n5:05	Control__Target position	Target position (32 Bit)	UINT32	RO	0x7n201120

Index 16n6 POS Control (RxPDO-Map)

Index (hex)	Name	Description	Data type	Flags	Default
16n6:0	POS Control	Position control	UINT8	RO	0x09
16n6:01	Control__Execute	Execute (1 Bit)	UINT32	RO	0x7n200101
16n6:02	Control__Emergency stop	Emergency stop (1 Bit)	UINT32	RO	0x7n200201
16n6:03	Control__Gap1	BYTE padding (6 Bit)	UINT32	RO	0x00000006
16n6:04	Control__Gap2	WORD padding (8 Bit)	UINT32	RO	0x00000008
16n6:05	Control__Target position	Target position (32 Bit)	UINT32	RO	0x7n201120
16n6:06	Control__Velocity	Max Velocity (32 Bit)	UINT32	RO	0x7n202120
16n6:07	Control__Start type	Start type (16 Bit)	UINT32	RO	0x7n202210
16n6:08	Control__Acceleration	Acceleration (16 Bit)	UINT32	RO	0x7n202310
16n6:09	Control__Deceleration	Deceleration (16 Bit)	UINT32	RO	0x7n202410

Index 16n7 Latch Control (RxPDO-Map)

Index (hex)	Name	Description	Data type	Flags	Default
16n7:0	Latch Control	Latch setting	UINT8	RO	0x06
16n7:01	Control__Enable latch active edge DI 1	Enable Latch extern on positive edge DI 1 (1 Bit)	UINT32	RO	0x7n300101
16n7:02	Control__Enable latch active edge DI 2	Enable Latch extern on positive edge DI 2 (1 Bit)	UINT32	RO	0x7n300201
16n7:03	Control__Enable Latch inactive edge DI 1	Enable Latch extern on negative edge DI 1 (1 Bit)	UINT32	RO	0x7n300301
16n7:04	Control__Enable Latch inactive edge DI 2	Enable Latch extern on negative edge DI 2 (1 Bit)	UINT32	RO	0x7n300401
16n7:05	Control__Gap1	BYTE padding (4 Bit)	UINT32	RO	0x00000004
16n7:06	Control__Gap2	WORD padding (8 Bit)	UINT32	RO	0x00000008

9.3 TxPDO Mapping Objects

The symbol **n** represents the axis number: 0 to 3.

Index 1An1 ENC Status (TxPDO-Map)

Index (hex)	Name	Description	Data type	Flags	Default
1An1:0	ENC Status	Encoder status	UINT8	RO	0x0D
1An1:01	Status__Gap1	empty (1 Bit)	UINT32	RO	0x00000001
1An1:02	Status__Gap2	empty (1 Bit)	UINT32	RO	0x00000001
1An1:03	Status__Set encoder done	Set position counter done (1 Bit)	UINT32	RO	0x6n000301
1An1:04	Status__Set encoder z latch-clear mode done	Set z latch clear mode done(1 Bit)	UINT32	RO	0x6n000401
1An1:05	Status__Counter underflow	Counter underflow (1 Bit)	UINT32	RO	0x6n000501
1An1:06	Status__Counter overflow	Counter overflow (1 Bit)	UINT32	RO	0x6n000601
1An1:07	Status__Index	Encoder index event (1 Bit)	UINT32	RO	0x6n000701
1An1:08	Status__Gap3	BYTE padding (1 Bit)	UINT32	RO	0x00000001
1An1:09	Status__Gap4	empty (5 Bit)	UINT32	RO	0x00000005
1An1:0A	Status__Sync error	Sync error (1 Bit)	UINT32	RO	0x6n000E01
1An1:0B	Status__Gap5	empty (1 Bit)	UINT32	RO	0x00000001
1An1:0C	Status__TxPDO Toggle	TxPDO Toggle (1 Bit)	UINT32	RO	0x6n001001
1An1:0D	Status__Encoder value	Encoder value (32-Bit)	UINT32	RO	0x6n001120

Index 1An3 STM Status (TxPDO-Map)

Index (hex)	Name	Description	Data type	Flags	Default
1An3:0	STM Status	Stepper motor status	UINT8	RO	0x10
1An3:01	Status__Ready to enable	Ready to enable (1 Bit)	UINT32	RO	0x6n100101
1An3:02	Status__Ready	Ready (1 Bit)	UINT32	RO	0x6n100201
1An3:03	Status__Warning	Warning (1 Bit)	UINT32	RO	0x6n100301
1An3:04	Status__Overtemperatur	Pre-Warning overtemperature	UINT32	RO	0x6n100401

Index (hex)	Name	Description	Data type	Flags	Default
		(1 Bit)			
1An3:05	Status__Warning open load	Open load detected on phase A or B	UINT32	RO	0x60100501
1An3:05	Status__Error	Error (1 Bit)	UINT32	RO	0x6n100601
1An3:06	Status__Error overtemperature	Error overtemperature (1 Bit)	UINT32	RO	0x6n100701
1An3:07	Status__Error short to ground	Error short to ground (1 Bit)	UINT32	RO	0x6n100801
1An3:09	Status__Moving positive	Moving positive (1 Bit)	UINT32	RO	0x6n100901
1An3:0A	Status__Moving negative	Moving negative (1 Bit)	UINT32	RO	0x6n100A01
1An3:0B	Status__Torque reduced	Torque reduced (1 Bit)	UINT32	RO	0x6n100B01
1An3:0C	Status__Digital input 1	Digital input 1 (1 Bit)	UINT32	RO	0x6n100C01
1An3:0D	Status__Digital input 2	Digital input 2 (1 Bit)	UINT32	RO	0x6n100D01
1An3:0E	Status__Sync error	Sync error (1 Bit)	UINT32	RO	0x6n100E01
1An3:0F	Status__Motor standstill	Motor is at standstill (1 Bit)	UINT32	RO	0x6n100F01
1An3:10	Status__TxPDO Toggle	TxPDO Toggle (1 Bit)	UINT32	RO	0x6n101001

Index 1An4 STM Synchron Info Data (TxPDO-Map)

Index (hex)	Name	Description	Data type	Flags	Default
1An4:0	STM Synchron Info Data		UINT8	RO	0x02
1An4:01	Status__Info data 1	Info data 1 (32 Bit)	UINT32	RO	0x6n101120
1An4:02	Status__Info data 2	Info data 2 (32 Bit)	UINT32	RO	0x6n101220

Index 1An5 POS Status Compact (TxPDO-Map)

Index (hex)	Name	Description	Data type	Flags	Default
1An5:0	POS Status Compact		UINT8	RO	0x0B
1An5:01	Status__Busy	Busy (1 Bit)	UINT32	RO	0x6n200101
1An5:02	Status__In-Target	In-Target (1 Bit)	UINT32	RO	0x6n200201
1An5:03	Status__Warning	Warning (1 Bit)	UINT32	RO	0x6n200301
1An5:04	Status__Error	Error (1 Bit)	UINT32	RO	0x6n200401
1An5:05	Status__Target overrun	Target position overrun	UINT32	RO	0x6n200501

Index (hex)	Name	Description	Data type	Flags	Default
		(1 Bit)			
1An5:06	Status__Accelerate	Accelerate (1 Bit)	UINT32	RO	0x6n200601
1An5:07	Status__Decelerate	Decelerate (1 Bit)	UINT32	RO	0x6n200701
1An5:08	Status__SoftEmg	Software Emergency (1 Bit)	UINT32	RO	0x6n200801
1An5:09	Status__CmdRejected	Command rejected (1 Bit)	UINT32	RO	0x6n200901
1An5:0A	Status__CmdAborted	Command Aborted (1 Bit)	UINT32	RO	0x6n200A01
1An5:0B	Status__Gap2	BYTE padding (6 Bit)	UINT32	RO	0x00000006

Index 1An6 POS Status (TxPDO-Map)

Index (hex)	Name	Description	Data type	Flags	Default
1An6:0	POS Status		UINT8	RO	0x0D
1An6:01	Status__Busy	Busy (1 Bit)	UINT32	RO	0x6n200101
1An6:02	Status__In-Target	In-Target (1 Bit)	UINT32	RO	0x6n200201
1An6:03	Status__Warning	Warning (1 Bit)	UINT32	RO	0x6n200301
1An6:04	Status__Error	Error (1 Bit)	UINT32	RO	0x6n200401
1An6:05	Status__Target overrun	Target position overrun (1 Bit)	UINT32	RO	0x6n200501
1An6:06	Status__Accelerate	Accelerate (1 Bit)	UINT32	RO	0x6n200601
1An6:07	Status__Decelerate	Decelerate (1 Bit)	UINT32	RO	0x6n200701
1An6:08	Status__SoftEmg	Software Emergency (1 Bit)	UINT32	RO	0x6n200801
1An6:09	Status__CmdRejected	Command rejected (1 Bit)	UINT32	RO	0x6n200901
1An6:0A	Status__CmdAborted	Command Aborted (1 Bit)	UINT32	RO	0x6n200A01
1An6:0B	Status__Gap2	BYTE padding (6 Bit)	UINT32	RO	0x00000006
1An6:0C	Status__Actual motor position	Actual position (32 Bit)	UINT32	RO	0x6n201120
1An6:0D	Status__Actual motor velocity	Actual velocity (32 Bit)	UINT32	RO	0x6n202120

Index 1An7 Pos Counter Status (TxPDO-Map)

Index (hex)	Name	Description	Data type	Flags	Default
1An7:0	Pos Counter Status	Position counter status	UINT8	RO	0x02
1An7:01	Status__Set position counter done	Set position counter done (1 Bit)	UINT32	RO	0x6n202301
1An7:02	Status__Sync error	Sync error (1 Bit)	UINT32	RO	0x6n202401
1An7:03	Status__TxPDO Toggle	TxPDO Toggle (1 Bit)	UINT32	RO	0x6n202501
1An7:04	Status__Gap1	BYTE padding (5 Bit)	UINT32	RO	0x00000005
1An7:05	Status__Gap2	WORD padding (8 Bit)	UINT32	RO	0x00000008
1An7:06	Status__Position counter value	Position counter value (32-Bit)	UINT32	RO	0x6n201120

Index 1An8 Latch Status (TxPDO-Map)

Index (hex)	Name	Description	Data type	Flags	Default
1An8:0	Latch Status		UINT8	RO	0x09
1An8:01	Status__Latch extern valid DI 1	Latch DI 1 extern valid (1 Bit)	UINT32	RO	0x6n300101
1An8:02	Status__Latch extern valid DI 2	Latch DI 2 extern valid (1 Bit)	UINT32	RO	0x6n300201
1An8:03	Status__Status of extern latch DI 1	Status of the ext. latch input DI 1 (1 Bit)	UINT32	RO	0x6n300301
1An8:04	Status__Status of extern latch DI 2	Status of the ext. latch input DI 2 (1 Bit)	UINT32	RO	0x6n300401
1An8:05	Status__Gap1	BYTE padding (4 Bit)	UINT32	RO	0x00000004
1An8:06	Status__Gap2	empty (7 Bit)	UINT32	RO	0x00000007
1An8:07	Status__TxPDO Toggle	TxPDO Toggle (1 Bit)	UINT32	RO	0x6n301001
1An8:08	Status__Encoder latched value	Latched encoder value (32-Bit)	UINT32	RO	0x6n301220
1An8:09	Status__Position counter latched value	Latched position counter value (32-Bit)	UINT32	RO	0x6n301320

Index 1An9 Input Signal (TxPDO-Map)

Index (hex)	Name	Description	Data type	Flags	Default
1An9:0	Input Signal		UINT8	RO	0x08
1An9:01	Status__Left reference input	Left reference input	UINT32	RO	0x6n101301

Index (hex)	Name	Description	Data type	Flags	Default
		(1 Bit)			
1An9:02	Status__Right reference input	Right reference input (1 Bit)	UINT32	RO	0x6n101401
1An9:03	Status__Encoder A channel input	Encoder A channel input (1 Bit)	UINT32	RO	0x6n101501
1An9:04	Status__Encoder B channel input	Encoder B channel input (1 Bit)	UINT32	RO	0x6n101601
1An9:05	Status__Encoder Z channel input	Encoder Z channel input (1 Bit)	UINT32	RO	0x6n101701
1An9:06	Status__Driver enable	Driver enabled signal (1 Bit)	UINT32	RO	0x6n101801
1An9:07	Status__Gap1	BYTE padding (2 Bit)	UINT32	RO	0x00000002
1An9:08	Status__Gap2	WORD padding (8 Bit)	UINT32	RO	0x00000008

9.4 Sync Manager Objects

Index 1C00 Sync manager type

Index (hex)	Name	Description	Data type	Flags	Default
1C00:0	Sync manager type	Using the sync managers	UINT8	RO	0x04
1C00:01	SubIndex 001	Sync-Manager Type Channel 1: Mailbox Write	UINT8	RO	0x01
1C00:02	SubIndex 002	Sync-Manager Type Channel 2: Mailbox Read	UINT8	RO	0x02
1C00:03	SubIndex 003	Sync-Manager Type Channel 3: Process Data Write (Outputs)	UINT8	RO	0x03
1C00:04	SubIndex 004	Sync-Manager Type Channel 4: Process Data Read (Inputs)	UINT8	RO	0x04

Index 1C12 RxPDO assign

Index (hex)	Name	Description	Data type	Flags	Default*
1C12:0	RxPDO assign	SyncManager 2 assignment: PDO Assign Outputs	UINT8	RO	0x1C
1C12:01	SubIndex 001	default assignment: Velocity control	UINT16	RW	0x1601
1C12:02	SubIndex 002	default assignment: Velocity	UINT16	RW	0x1602

Index (hex)	Name	Description	Data type	Flags	Default*
		control			
1C12:03	SubIndex 003	default assignment: Velocity control	UINT16	RW	0x1604
1C12:04	SubIndex 004	default assignment: Velocity control	UINT16	RW	0x1611
1C12:05	SubIndex 005	default assignment: Velocity control	UINT16	RW	0x1612
1C12:06	SubIndex 006	default assignment: Velocity control	UINT16	RW	0x1614
1C12:07	SubIndex 007	default assignment: Velocity control	UINT16	RW	0x1621
1C12:08	SubIndex 008	default assignment: Velocity control	UINT16	RW	0x1622
1C12:09	SubIndex 009	default assignment: Velocity control	UINT16	RW	0x1624
1C12:0A	SubIndex 010	default assignment: Velocity control	UINT16	RW	0x1631
1C12:0B	SubIndex 011	default assignment: Velocity control	UINT16	RW	0x1632
1C12:0C	SubIndex 012	default assignment: Velocity control	UINT16	RW	0x1634
1C12:0D	SubIndex 013	reserve space for additional RxPDO assignment	UINT16	RW	0x0000
1C12:0E	SubIndex 014	reserve space for additional RxPDO assignment	UINT16	RW	0x0000
1C12:0F	SubIndex 015	reserve space for additional RxPDO assignment	UINT16	RW	0x0000
1C12:10	SubIndex 016	reserve space for additional RxPDO assignment	UINT16	RW	0x0000
1C12:11	SubIndex 017	reserve space for additional RxPDO assignment	UINT16	RW	0x0000
1C12:12	SubIndex 018	reserve space for additional RxPDO assignment	UINT16	RW	0x0000
1C12:13	SubIndex 019	reserve space for additional RxPDO assignment	UINT16	RW	0x0000
1C12:14	SubIndex 020	reserve space for additional RxPDO assignment	UINT16	RW	0x0000
1C12:15	SubIndex 021	reserve space for additional RxPDO assignment	UINT16	RW	0x0000
1C12:16	SubIndex 022	reserve space for additional RxPDO assignment	UINT16	RW	0x0000
1C12:17	SubIndex 023	reserve space for additional RxPDO assignment	UINT16	RW	0x0000
1C12:18	SubIndex 024	reserve space for additional RxPDO assignment	UINT16	RW	0x0000
1C12:19	SubIndex 025	reserve space for additional RxPDO assignment	UINT16	RW	0x0000
1C12:1A	SubIndex 026	reserve space for additional RxPDO assignment	UINT16	RW	0x0000

Index (hex)	Name	Description	Data type	Flags	Default*
1C12:1B	SubIndex 027	reserve space for additional RxPDO assignment	UINT16	RW	0x0000
1C12:1C	SubIndex 028	reserve space for additional RxPDO assignment	UINT16	RW	0x0000

*Sub index 001 to 028 contains the index of the associated RxPDO mapping object

Index 1C13 TxPDO assign

Index (hex)	Name	Description	Data type	Flags	Default*
1C13:0	TxPDO assign	SyncManager 3 assignment: PDO Assign Inputs	UINT8	RO	0x20
1C13:01	SubIndex 001	default assignment: Velocity control	UINT16	RW	0x1A01
1C13:02	SubIndex 002	default assignment: Velocity control	UINT16	RW	0x1A03
1C13:03	SubIndex 003	default assignment: Velocity control	UINT16	RW	0x1A11
1C13:04	SubIndex 004	default assignment: Velocity control	UINT16	RW	0x1A13
1C13:05	SubIndex 005	default assignment: Velocity control	UINT16	RW	0x1A21
1C13:06	SubIndex 006	default assignment: Velocity control	UINT16	RW	0x1A23
1C13:07	SubIndex 007	default assignment: Velocity control	UINT16	RW	0x1A31
1C13:08	SubIndex 008	default assignment: Velocity control	UINT16	RW	0x1A33
1C13:09	SubIndex 009	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:0A	SubIndex 010	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:0B	SubIndex 011	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:0C	SubIndex 012	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:0D	SubIndex 013	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:0E	SubIndex 014	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:0F	SubIndex 015	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:10	SubIndex 016	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:11	SubIndex 017	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:12	SubIndex 018	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:13	SubIndex 019	reserve space for additional TxPDO assignment	UINT16	RW	0x0000

Index (hex)	Name	Description	Data type	Flags	Default*
1C13:14	SubIndex 020	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:15	SubIndex 021	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:16	SubIndex 022	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:17	SubIndex 023	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:18	SubIndex 024	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:19	SubIndex 025	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:1A	SubIndex 026	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:1B	SubIndex 027	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:1C	SubIndex 028	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:1D	SubIndex 029	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:1E	SubIndex 030	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:1F	SubIndex 031	reserve space for additional TxPDO assignment	UINT16	RW	0x0000
1C13:20	SubIndex 032	reserve space for additional TxPDO assignment	UINT16	RW	0x0000

*Sub index 001 to 032 contains the index of the associated TxPDO mapping object

Index 1C32 Sync Manager (SM) output parameter

Index (hex)	Name	Description	Data type	Flags	Default
1C32:0	SM output parameter	Synchronization parameters for the outputs	UINT8	RO	0x20
1C32:01	Synchronization Type	Current synchronization mode: • 0: Free Run • 1: Synchronous without SM 2 event • 2: DC-Mode - Synchronous with SYNC0 Event • 3: DC-Mode - Synchronous with SYNC1 event	UINT8	RO	0x0001
1C32:02	Cycle Time	Cycle time (in ns): • Free Run: Cycle time of the local timer • Synchronous with SM 2 event: Master cycle time • DC mode: SYNC0/SYNC1	UINT8	RO	0x00000000

Index (hex)	Name	Description	Data type	Flags	Default
		Cycle Time			
1C32:04	Synchronization Types supported	Supported synchronization modes: • Bit 0 = 1: free run is supported • Bit 1 = 1: Synchron with SM 2 event is supported • Bit 2-3 = 01: DC mode is supported • Bit 4-5 = 10: Output shift with SYNC1 event (only DC mode)	UINT8	RO	0x8007
1C32:05	Minimum Cycle Time	Minimum cycle time (in ns)		RO	0x00000000
1C32:06	Calc and Copy Time	Minimum time between SYNC0 and SYNC1 event (in ns, DC mode only)		RO	0x00000000
1C32:08	Get Cycle Time	• 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started Set parameter to 1 in order to update the Cycle Time (1C32:02, 1C33:02) parameter with the maximum measured value		RW	0x0000
1C32:09	Delay Time	Time between SYNC1 event and output of the outputs (in ns, DC mode only)		RO	0x00000000
1C32:0A	Sync0 Cycle Time			RW	0x00000000
1C32:0B	SM-Event Missed	Number of missed SM events in OPERATIONAL (DC mode only)		RO	0x0000
1C32:0C	Cycle Time Too Small	Cycle was not completed in time or the next cycle began too early		RO	0x0000
1C32:20	Sync Error	The synchronization was not correct in the last cycle (outputs were output too late; DC mode only)		RO	FALSE

Index 1C33 Sync Manager (SM) input parameter

Index (hex)	Name	Description	Data type	Flags	Default
1C33:0	SM input parameter	Synchronization parameters for the inputs	UINT8	RO	0x20
1C33:01	Synchronization Type	Current synchronization mode: • 0: Free Run • 1: Synchron with SM 3 Event (no outputs	UINT8	RO	0x0022

Index (hex)	Name	Description	Data type	Flags	Default
		available) • 2: DC - Synchron with SYNC0 Event • 3: DC - Synchron with SYNC1 Event • 34: Synchron with SM 2 Event (outputs available)			
1C33:02	Cycle Time	Cycle time (in ns): • Free Run: Cycle time of the local timer • Synchronous with SM 2 event: Master cycle time • DC mode: SYNC0/SYNC1 Cycle Time	UINT8	RO	0x00000000
1C33:04	Synchronization Types supported	Supported synchronization modes: • Bit 0 = 1: free run is supported • Bit 1 = 1: Synchron with SM 2 event is supported • Bit 2-3 = 01: DC mode is supported • Bit 4-5 = 10: Output shift with SYNC1 event (only DC mode)	UINT8	RO	0x8007
1C33:05	Minimum Cycle Time	Minimum cycle time (in ns)		RO	0x00000000
1C33:06	Calc and Copy Time	Time between reading of the inputs and availability of the inputs for the master (in ns, only DC mode)		RO	0x00000000
1C33:08	Get Cycle Time	• 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started Set parameter to 1 in order to update the Cycle Time (1C32:02, 1C33:02) parameter with the maximum measured value		RW	0x0000
1C33:09	Delay Time	Time between SYNC1 event and reading of the inputs (in ns, only DC mode)		RO	0x00000000
1C33:0A	Sync0 Cycle Time			RW	0x00000000
1C33:0B	SM-Event Missed	Number of missed SM events in OPERATIONAL (DC mode only)		RO	0x0000
1C33:0C	Cycle Time Too Small	Cycle was not completed in time or the next cycle began too early		RO	0x0000
1C33:20	Sync Error	The synchronization was not		RO	FALSE

Index (hex)	Name	Description	Data type	Flags	Default
		correct in the last cycle (outputs were output too late; DC mode only)			

9.5 Input Data

The symbol **n** represents the axis number: 0 to 3.

Index 6n00 ENC Inputs

Index (hex)	Name	Description	Data type	Flags	Default
6n00:0	ENC Status	Encoder status inputs	UINT8	RO	0x11
6n00:03	Set encoder done	The encoder value has been set	BOOLEAN	RO	FALSE
6n00:04	Set encoder z latch-clear mode done	Indicates whether the encoder index latch-clear mode was set successfully	BOOLEAN	RO	FALSE
6n00:05	Counter underflow	Counter underflow	BOOLEAN	RO	FALSE
6n00:06	Counter overflow	Counter overflow	BOOLEAN	RO	FALSE
6n00:07	Encoder index event	Encoder index event detected	BOOLEAN	RO	FALSE
6n00:0E	Sync error	The Sync error bit is only required for DC mode. It indicates whether a synchronization error has occurred during the previous cycle	BOOLEAN	RO	FALSE
6n00:10	TxPDO Toggle	The TxPDO toggle is toggled by the slave when the data of the associated TxPDO is updated	BOOLEAN	RO	FALSE
6n00:11	Actual encoder value	The counter value	INT32	RO	0x00000000

Index 6n10 STM Inputs

Index (hex)	Name	Description	Data type	Flags	Default
6n10:0	STM Inputs	Stepper motor inputs	UINT8	RO	0x18
6n10:01	Ready to enable	Driver stage is ready for enabling	BOOLEAN	RO	FALSE
6n10:02	Ready	Driver stage is ready for operation	BOOLEAN	RO	FALSE

Index (hex)	Name	Description	Data type	Flags	Default
6n10:03	Warning	A warning has occurred	BOOLEAN	RO	FALSE
6n10:04	Warning over temperature	Over-temperature pre-warning	BOOLEAN	RO	FALSE
6n10:05	Warning open load	Open load detected on phase A or phase B • In motor stand still, open load cannot be measured, as the coils might eventually have zero current	BOOLEAN	RO	FALSE
6n10:06	Error	An error has occurred	BOOLEAN	RO	FALSE
6n10:07	Error over temperature	Over-temperature error	BOOLEAN	RO	FALSE
6n10:08	Error short to ground	Short to ground phase A or phase B	BOOLEAN	RO	FALSE
6n10:09	Moving positive	Motor turns in positive direction	BOOLEAN	RO	FALSE
6n10:0A	Moving negative	Motor turns in negative direction	BOOLEAN	RO	FALSE
6n10:0B	Torque reduced	Reduced torque is active	BOOLEAN	RO	FALSE
6n10:0C	Digital input 1	Digital input 1	BOOLEAN	RO	FALSE
6n10:0D	Digital input 2	Digital input 2	BOOLEAN	RO	FALSE
6n10:0E	Sync error	The Sync error bit is only required for DC mode. It indicates whether a synchronization error has occurred during the previous cycle.	BOOLEAN	RO	FALSE
6n10:0F	Motor standstill	Indicates the whether motor is in standstill (TRUE - standstill)	BOOLEAN	RO	FALSE
6n10:10	TxPDO Toggle	The TxPDO toggle is toggled by the slave when the data of the associated TxPDO is updated.	BOOLEAN	RO	FALSE
6n10:11	Info data 1	Synchronous information (selection via sub index 8n12:11)	INT32	RO	0x00000000
6n10:12	Info data 2	Synchronous information (selection via sub index 8n12:19)	INT32	RO	0x00000000
6n10:13	Left reference input	Signal of the left reference input	BOOLEAN	RO	FALSE
6n10:14	Right reference input	Signal of the right reference input	BOOLEAN	RO	FALSE

Index (hex)	Name	Description	Data type	Flags	Default
6n10:15	Encoder A channel input	Signal of the encoder A channel	BOOLEAN	RO	FALSE
6n10:16	Encoder B channel input	Signal of the encoder B channel	BOOLEAN	RO	FALSE
6n10:17	Encoder Z channel input	Signal of the encoder Z channel	BOOLEAN	RO	FALSE
6n10:18	Driver disabled	Indicates whether the driver has been enabled	BOOLEAN	RO	FALSE

Index 6n20 POS Inputs

Index (hex)	Name	Description	Data type	Flags	Default
6n20:0	POS Inputs		UINT8	RO	0x25
6n20:01	Busy	A travel command is active	BOOLEAN	RO	FALSE
6n20:02	In-Target	Motor has arrived at target	BOOLEAN	RO	FALSE
6n20:03	Warning	A warning has occurred	BOOLEAN	RO	FALSE
6n20:04	Error	An error has occurred	BOOLEAN	RO	FALSE
6n20:05	Target overrun	Change the target position on the fly may lead to an overshoot of the position and change in direction of rotation is necessary	BOOLEAN	RO	FALSE
6n20:06	Accelerate	Motor is in the acceleration phase	BOOLEAN	RO	FALSE
6n20:07	Decelerate	Motor is in the deceleration phase	BOOLEAN	RO	FALSE
6n20:08	Soft Emg	Emergency stop has been triggered by software	BOOLEAN	RO	FALSE
6n20:09	Cmd rejected	Motion command has been reject	BOOLEAN	RO	FALSE
6n20:0A	Cmd aborted	Motion command has been aborted	BOOLEAN	RO	FALSE
6n20:11	Actual motor position	Current target position of the travel command generator	INT32	RO	0x00000000
6n20:21	Actual motor velocity	Current velocity of the travel command generator	INT32	RO	0x00000000
6n20:22	Actual drive time_xx	Travel command time information (see subindex 8n21:11) (Not supported)	UINT32	RO	0x00000000

Index (hex)	Name	Description	Data type	Flags	Default
6n20:23	Set position counter done	The position counter has been set	BOOLEAN	RO	FALSE
6n20:24	Sync error	The Sync error bit is only required for DC mode. It indicates whether a synchronization error has occurred during the previous cycle	BOOLEAN	RO	FALSE
6n20:25	TxPDO Toggle	The TxPDO toggle is toggled by the slave when the data of the associated TxPDO is updated.	BOOLEAN	RO	FALSE

Index 6n30 Latch Inputs

Index (hex)	Name	Description	Data type	Flags	Default
6n30:0	Latch Inputs		UINT8	RO	0x13
6n30:01	Latch extern valid DI 1	The counter value was stored via the external latch DI 1	BOOLEAN	RO	FALSE
6n30:02	Latch extern valid DI 2	The counter value was stored via the external latch DI 2	BOOLEAN	RO	FALSE
6n30:03	Status of extern latch DI 1	Status of the ext. latch input DI 1	BOOLEAN	RO	FALSE
6n30:04	Status of extern latch DI 2	Status of the ext. latch input DI 2	BOOLEAN	RO	FALSE
6n30:10	TxPDO Toggle	The TxPDO toggle is toggled by the slave when the data of the associated TxPDO is updated	BOOLEAN	RO	FALSE
6n30:12	Latched encoder value	Latched encoder value	INT32	RO	0x00000000
6n30:13	Latched position counter value	Latched position counter value	INT32	RO	0x00000000

9.6 Output Data

The symbol **n** represents the axis number: 0 to 3.

Index 7n00 ENC Outputs

Index (hex)	Name	Description	Data type	Flags	Default
7n00:0	ENC Outputs		UINT8	RO	0x12
7n00:02	Set encoder	Assigned value to encoder counter • By setting the bit from FALSE to TRUE the encoder value (7n00:11) will be set	BOOLEAN	RO	FALSE
7n00:03	Set position counter	Assigned value to position counter • By setting the bit from FALSE to TRUE the position counter value (7n00:12) will be set	BOOLEAN	RO	FALSE
7n00:04	Set encoder z latch-clear mode	Set the latch and clear mode of the encoder counter in case of an index (z) event • By setting the bit from FALSE to TRUE the Encoder z latch-clear mode (7n00:06) will be set	BOOLEAN	RO	FALSE
7n00:06	Encoder z latch-clear mode	Latch-clear mode of the encoder for an index (z) event • Valid range: 0x00 ~ 0x04 • 0: Disable index latch • 1: Latch index once. The encoder value will be latched by the index signal only once after 7n00:06 has been set to 1 • 2: Latch continuous: After setting 7n00:06 to 2 the encoder value will be latched each time an index latch occurs. • 3: Latch and clear once: After setting 7n00:06 to 3 the encoder value will be latched and the encoder counter will be cleared at the first index signal	BIT8	RO	0x00

Index (hex)	Name	Description	Data type	Flags	Default
		encountered • 4: Latch and clear continuous: After setting 7n00:06 to 4 the encoder value will be latched and the encoder counter will be cleared after each index signal event			
7n00:11	Set encoder value	Encoder value to be set via "Set encoder"	INT32	RO	0x00000000
7n00:12	Set position counter value	Position counter value to be set via "Set position counter"	INT32	RO	0x00000000

Index 7n10 STM Outputs

Index (hex)	Name	Description	Data type	Flags	Default
7n10:0	STM Outputs	Stepper motor outputs	UINT8	RO	0x21
7n10:01	Enable	Activates the output stage • Enable = TRUE Output "Maximal current" (0x8n10:01) • Enable = FALSE Output to "Power on motor current" (0x8n10:08)	BOOLEAN	RO	FALSE
7n10:02	Reset	All errors that may have occurred are reset by setting this bit (rising edge) • By setting the bit from FALSE to TRUE errors which occurred during motion execution (e.g. over-temperature) will be cleared	BOOLEAN	RO	FALSE
7n10:03	Reduce torque	Activation of reduced torque (coil current) (sub index 8n10:02)	BOOLEAN	RO	FALSE
7n10:0c	Digital output1	Digital output1	BOOLEAN	RO	FALSE
7n10:11	Position	Set position; Absolute target	INT32	RO	0x00000000

Index (hex)	Name	Description	Data type	Flags	Default
		position for the "Position control" mode (see chapter 6.4)			
7n10:21	Velocity	Set velocity The target velocity for the "Velocity control" mode (see chapter 7)	INT32	RO	0x00000000

Index 7n20 POS Outputs

Index (hex)	Name	Description	Data type	Flags	Default
7n20:0	POS Outputs		UINT8	RO	0x24
7n20:01	Execute	Start travel command (rising edge), or prematurely abort travel command (falling edge)	BOOLEAN	RO	FALSE
7n20:02	Emergency stop	Prematurely abort travel command with an emergency ramp (rising edge)	BOOLEAN	RO	FALSE
7n20:11	Target position	Specification of the target position (unit: steps). • Depending on the "Start type" (0x7n20:22) the position can either be relative, absolute or additive	INT32	RO	0x00007FFF
7n20:21	Velocity	Specification of the maximum set velocity (unit: steps/second)	INT32	RO	0x00000000
7n20:22	Start type	Specification of the start types (see Table 9: Start type definition)	UINT16	RO	0x0000
7n20:23	Acceleration	Acceleration time (unit: see 0x8n20:08) • see chapter 6.2.1 and 6.2.2. • Valid range: 0x0000~0xFFFF	UINT16	RO	0x0000
7n20:24	Deceleration	Deceleration time (unit: see 0x8n20:08) • see chapter 6.2.1 and 6.2.2. • Valid range:	UINT16	RO	0x0000

Index (hex)	Name	Description	Data type	Flags	Default
		0x0000~0xFFFF			

Index 7n30 Latch Outputs

Index (hex)	Name	Description	Data type	Flags	Default
7n30:0	Latch Outputs		UINT8	RO	0x04
7n30:01	Enable latch active edge DI 1	DI 1 trigger level: rising edge	BOOLEAN	RO	FALSE
7n30:02	Enable latch active edge DI 2	DI 2 trigger level: rising edge	BOOLEAN	RO	FALSE
7n30:03	Enable latch inactive edge DI 1	DI 1 trigger level: falling edge	BOOLEAN	RO	FALSE
7n30:04	Enable latch inactive edge DI 2	DI 2 trigger level: falling edge	BOOLEAN	RO	FALSE

9.7 Configuration Data

The symbol **n** represents the axis number: 0 to 3.

Index 8n00 ENC Settings

Index (hex)	Name	Description	Data type	Flags	Default
8n00:0	ENC Settings	Encoder settings	UINT8	RO	0x0E
8n00:0E	Reversion of rotation	Activates reversion of rotation of the encoder	BOOLEAN	RW	FALSE

Index 8n10 STM Motor Settings

Index (hex)	Name	Description	Data type	Flags	Default
8n10:0	STM Motor Settings	Stepper motor settings	UINT8	RO	0x12
8n10:01	Maximum run current	Peak motor coil current for driving (unit: mA), default: 750, max: 1500	UINT16	RW	0x02EE (750)
8n10:02	Reduced run current	Reduced peak motor coil current for driving(reduced torque, unit: mA), default: 375, max: 1500	UINT16	RW	0x0177 (375)

Index (hex)	Name	Description	Data type	Flags	Default
		Will be activated when "Reduced torque" (0x7n10:03) has been set to true			
8n10:03	Maximum hold current	Motor standstill current (unit: mA), default: 750, max: 1500	UINT16	RW	0x02EE (750)
8n10:04	Reduced hold current	Reduced Motor standstill current (unit: mA), default: 375, max: 1500. Will be activated when "Reduced torque" (0x7n10:03) has been set to true	UINT16	RW	0x0177 (375)
8n10:06	Motor fullsteps	Motor full steps per revolution (not supported)	UINT16	RW	0x0000
8n10:07	Micro Steps	Number of microsteps per full step. • Supported values: 256, 128, 64, 32, 16, 8, 4, 2, 1	DT0801EN16	RW	0x0008 ("256")
8n10:08	Power on motor current	Motor coil current output directly after power on (unit: 1 mA)	UINT16	RW	0x0177 (375)
8n10:09	Max Start Velocity	Maximum possible start velocity of the motor	UINT16	RW	0x0064 (100)
8n10:12	Safe motor current	Set the safe motor coil current (will be applied if state changes from OP to a different state)(unit: 1 mA)	UINT16	RW	0x0177 (375)

Index 8n12 STM Features

Index (hex)	Name	Description	Data type	Flags	Default
8n12:0	STM Features	Stepper motor features	UINT8	RO	0x42
8n12:01	Operation mode	Operating mode, • 0: automatic • 2: velocity control • 3: position control	DT0802EN04	RW	0x00 ("Automatic")
8n12:09	Invert motor polarity	Activates reversal of the motor rotation	BOOLEAN	RW	FALSE

Index (hex)	Name	Description	Data type	Flags	Default
		direction.			
8n12:11	Select info data 1	Select "Info data 1": • 3: Motor coil current A • 4: Motor coil current B • 7: Motor velocity • 8: Encoder position • 9: Position counter • 10: Error code (see motion stop error table)	DT0803EN08	RW	0x03 ("Motor coil current A")
8n12:19	Select info data 2	Select "Info data 2": • 3: Motor coil current A • 4: Motor coil current B • 7: Motor velocity • 8: Encoder position • 9: Position counter • 10: Error code (see motion stop error table)	DT0803EN08	RW	0x04 ("Motor coil current B")
8n12:2A	Power on DO 1	Set the power on DO 1 (will be applied directly after switching the device on)	BOOLEAN	RW	FALSE
8n12:2B	Safety DO 1	Set the safe DO 1 (will be applied if state changes from OP to different state)	BOOLEAN	RW	FALSE
8n12:30	Invert digital input 1	Inversion of digital input 1	BOOLEAN	RW	FALSE
8n12:31	Invert digital input 2	Inversion of digital input 2	BOOLEAN	RW	FALSE
8n12:32	Function for input 1	Select the digital input 1 type: • 0: Normal input • 1: Hardware stop enable • 2: Emergency stop enable • 3: All axes emergency stop enable	DT080AEN04	RW	0x00 ("Normal input")
8n12:36	Function for input 2	Select the digital input 2 type: • 0: Normal input • 1: Hardware stop	DT080AEN04	RW	0x00 ("Normal input")

Index (hex)	Name	Description	Data type	Flags	Default
		enable • 2: Emergency stop enable • 3: All axes emergency stop enable			
8n12:37	Limit switch stop mode	Hardware limit stop mode • 0: Limit switch hard stop - motion output stops immediately once the limit switch has been activated • 1: Limit switch soft stop - motor decelerates to stop. The deceleration setting has to be done via 8n20:0D	DT080BEN01	RW	0x00 ("Limit switch hard stop")
8n12:40	Encoder index latch trigger	Latch trigger setting for the encoder index (z) pulse: • 0: Level trigger • 1: Rising edge trigger • 2: Falling edge trigger • 3: Rising and falling edge trigger	DT0811EN03	RW	0x00 ("Level trigger")
8n12:42	Encoder index polarity	Active polarity of the encoder index (z): • 0: Low active • 1: High active	DT0813EN01	RW	0x01 ("High active")
8n12:43	Generate hardware limit stop error	Set whether a error stop will be generated if motion has been stopped by hardware limit switch. The error first have to be cleared before the next motion command can be issued. • 0: Generate an	BOOLEAN		TRUE

Index (hex)	Name	Description	Data type	Flags	Default
		error stop - Error reset is necessary (0x7010:02) • 1: no error stop - No error reset is necessary to issue the next command. This mode is being used if limit stop is being used for home search.			

Index 8n20 POS Settings

Index (hex)	Name	Description	Data type	Flags	Default
8n20:0	POS Settings	Position settings	UINT8	RO	0x07
8n20:01	Velocity min	Minimum set velocity	UINT32	RW	0x00000000
8n20:02	Velocity max	Maximum set velocity • Maximum velocity supported by the system	UINT32	RW	0x00002710 (10000)
8n20:03	Acceleration pos	Acceleration time in positive direction of rotation (unit: see 8n20:08)	UINT16	RW	0x03E8 (1000)
8n20:04	Acceleration neg	Acceleration time in negative direction of rotation (unit: see 8n20:08)	UINT16	RW	0x03E8 (1000)
8n20:05	Deceleration pos	Deceleration time in positive direction of rotation (unit: see 8n20:08)	UINT16	RW	0x03E8 (1000)
8n20:06	Deceleration neg	Deceleration time in negative direction of rotation (unit: see 8n20:08)	UINT16	RW	0x03E8 (1000)
8n20:07	Emergency deceleration	Emergency deceleration time (both directions of rotation, unit: see 8n20:08)	UINT16	RW	0x0000
8n20:08	Acceleration unit	Set the acceleration unit: • 0: Acceleration time	DT0814EN02	RW	0 ("Acceleration time from

Index (hex)	Name	Description	Data type	Flags	Default
		from Vmin to Vmax [ms] • 1: Acceleration time from Vmin to Vtarget [ms] • 2: Acceleration [128*uStep/sec^2] • 3: Acceleration time from Vcurrent to Vtarget [ms]			Vmin to Vmax [ms]")
8n20:09	Acc-Dec parameter definition	Set the acceleration-deceleration profile: • 1: Acceleration-->Start phase & Deceleration-->Stop Phase • 2: Acceleration-->Acceleration & Deceleration--> Deceleration (not support by "Acceleration unit" = 3)	DT0815EN01	RW	0 ("Acceleration-->Start phase & Deceleration-->Stop Phase")
8n20:0D	Hardlimit deceleration	Hardware limit deceleration (both directions of rotation, unit: see 8020:08) • Valid range: 0~65535	UINT16	RW	0x0000

Index 8n21 POS Features

Index (hex)	Name	Description	Data type	Flags	Default
8n21:0	POS Features		UINT8	RO	0x01
8n21:01	Start type	Standard start type: • 0: Idle, • 1: Absolute, • 2: Relative, • 3:Endless plus • 4:Endless minus • 6: Additive, • 1001: absolute change, • 1002: relative change, • 1006: additive change	DT080FEN16	RW	0x0002 ("Relative")

9.8 Driver Tuning Functions

Index 8n30 Vendor Specific

Index (hex)	Name	Description	Data type	Flags	Default
8n30:0	Vendor specific	Stepper motor specific tuning configuration	UINT8	RO	0x0D
8n30:01	GCONF	General configuration	UINT32	RW	0x00000000
8n30:02	CHOPCONF	Chopper configuration	UINT32	RW	0x100101D5
8n30:03	COOLCONF	Smart energy control "coolStep" and "stallGuard"	UINT32	RW	0x00000000
8n30:04	PWMCONF	Voltage PWM mode stealthChop	UINT32	RW	0x00050480
8n30:05	TPOWER_DOWN	Sets the delay time after standstill of the motor to motor current power down. The time range is about 0 to 4 seconds.	UINT32	RW	0x00000007
8n30:06	TSTEP	Actual measured time between two 1/256 micro steps derived from the step input frequency in units of 1/F_CLK. (F_CLK = 2^24)	UINT32	RO	--
8n30:07	TPWMTHRS	This is the upper velocity of the for "stealthChop" voltage PMW mode. TSTEP>=TPWMTHRS - "steathChop" PWM mode is enabled, if configured - "dsStep" is disabled	UINT32	RW	0x00000000
8n30:08	TCOOLTHRS	This is the lower threshhold velocity for switching on smart energy "coolStep and "stallGuard" feature.	UINT32	RW	0x00000000
8n30:09	THIGH	This velocity setting allows velocity dependent switching into different chopper mode and fullstepping to maximize torque.	UINT32	RW	0x00000000
8n30:0A	VDCMIN	Automatic commutation dcStep enabled above velocity VDCMIN (unsigned)	UINT32	RW	0x00000000
8n30:0B	DCCTRL	dcStep (DC) automatic		WO	--

Index (hex)	Name	Description	Data type	Flags	Default
		commutation configuration register			
8n30:0C	DRV_ STATUS	"stallGuard" value and driver error flags (see separate table)		RO	--
8n30:0D	PWM_SCALE	Actual PWM amplitude scaler (255=max. Voltage) In voltage mode PWM, this value allows to detect a motor stall.		RO	--

The tables in chapter 11 provides further information about the trinamic TMC5130A register represented by the "Vendor Specific" Service Data Object.

9.9 Information and Diagnostic Data

The symbol **n** represents the axis number: 0 to 3.

Index An10 STM Diag data

Index (hex)	Name	Description	Data type	Flags	Default
An10:0	STM Diag data	Stepper motor diagnostic data	UINT8	RO	0x11
An10:02	Over temperature	Driver IC temperature has reached more than 80 °C • ATTENTION: This error message must be acknowledged by the user (see index 0x7n10:02)	BOOLEAN	RO	FALSE
An10:03	Torque overload	Not supported	BOOLEAN	RO	FALSE
An10:04	Under voltage	Indicates an undervoltage on the charge pump. The driver is disabled in this case • ATTENTION: This error message must be acknowledged by the user (see index 0x7n10:02)	BOOLEAN	RO	FALSE
An10:05	Over voltage	Not supported	BOOLEAN	RO	FALSE

Index (hex)	Name	Description	Data type	Flags	Default
An10:06	Short circuit A	Short to GND detected on phase A. The driver becomes disabled.	BOOLEAN	RO	FALSE
An10:07	Short circuit B	Short to GND detected on phase B. The driver becomes disabled.	BOOLEAN	RO	FALSE
An10:08	No control power	Not supported	BOOLEAN	RO	FALSE
An10:09	Misc error	Driver has been shut down due to - overtemperature - short circuit detection - undervoltage "uv_cp" ATTENTION: This error message must be acknowledged by the user (see index 0x7n10:02)	BOOLEAN	RO	FALSE
An10:0a	Configuration	Not supported	BOOLEAN	RO	FALSE
An10:11	Actual operation mode	Not supported	DT0809EN04	RO	0x00 ("undefined")
An10:1e	Error code	Shows the error which resulted in a motion execution stopped (see motion stop error table)	UINT32	RO	0x00000000

Index An20 POS Diag data

Index (hex)	Name	Description	Data type	Flags	Default
An20:0	POS Diag data	Stepper motor diagnostic data for position control	UINT8	RO	0x06
An20:01	Command rejected	Dynamic change of the target position was not accepted	BOOLEAN	RO	FALSE
An20:02	Command aborted	Command aborted due to internal error or emergency stop	BOOLEAN	RO	FALSE
An20:03	Target overrun	Change the target position on the fly may lead to an overshoot of the position and therefore change in direction of rotation may be necessary	BOOLEAN	RO	FALSE
An20:04	Target timeout	Not supported	BOOLEAN	RO	FALSE

Index (hex)	Name	Description	Data type	Flags	Default
An20:05	Position lag	Not supported	BOOLEAN	RO	FALSE
An20:06	Emergency stop	Emergency stop	BOOLEAN	RO	FALSE

9.10 Configuration Parameters Storage

Index F008 Internal EEPROM

Index (hex)	Name	Description	Data type	Flags	Default
F008:0	Internal EEPROM	Storing CoE parameters to the internal EEPROM.	UINT8	RO	0x05
F008:01	Code Word	Password for saving CoE configuration data to the EEPROM Password: 0x12345678	UINT32	RW	0x00000000
F008:02	Save Counter	Total number of save sequence	UINT16	RO	0x0000
F008:03	Load factory default	Load factory default configuration immediately after power on. • By setting this parameter to FALSE the user set configuration data (0x8000 to 0x8321) will be loaded after power on	BOOLEAN	RW	TRUE
F008:04	Save configuration data	Save all configuration setting to local non-volatile memory. • Set to TRUE in order to save the configuration data (0x8000 to 0x8321) to the memory of the ECAT-2094S	BOOLEAN	RW	FALSE
F008:05	Save error encountered	Indicates whether data has been successfully written to memory	BOOLEAN	RO	FALSE
F008:05	Initialization error	Internal EEPROM access failed	BOOLEAN	RO	FALSE

9.11 Station alias Configuration

Index F009 Station Alias

Index (hex)	Name	Description	Data type	Flags	Default
F009:0	Station Alias		UINT8	RO	0x02
F009:01	Rotary switch value	rotary switch value • set this value to the corresponding ESC register if the "ESC register load setting" is 0	UINT16	RO	0x00
F009:02	ESC register load setting	Get alias value from rotary switch or SII, and set this value to the corresponding ESC register: • 0: rotary switch • 1: SII	BOOLEAN	RW	0x0

10 Error Table

Motion Stop Error	
Error	Description
0x0001	Software emergency stop
0x0002	Over-temperature
0x0004	Under voltage
0x0008	Short circuit A
0x0010	Short circuit B
0x0020	Left hardware switch
0x0040	Right hardware switch
0x0080	EtherCAT status changed from OP to PreOP or SafeOP during motion execution
0x0100	The motion chip has been reset. This error mainly occurs when there is no motor voltage (VM) supply

Table 14: Motion stop error table

11 Vendor Specific Register Definitions

GCONF - General Configuration Register			
Bit	Name	Description	Default
0	I_scale_analog	0: Normal operation, use internal reference voltage 1: Use voltage supplied to AIN as current reference	0
1	internal_Rsense	0: Normal operation 1: Internal sense resistors. Use current supplied into AIN as reference for internal sense resistor	0
2	en_pwm_mode	1: stealthChop voltage PWM mode enabled (depending on velocity thresholds). Switch from off to on state while in stand still, only.	0
3	enc_commutation	1: Enable commutation by full step encoder (DCIN_CFG5 = ENC_A, DCEN_CFG4 = ENC_B)	0
4	shaft	1: Inverse motor direction	0
5	diag0_error	(only with SD_MODE=1) 1: Enable DIAG0 active on driver errors: Over temperature (ot), short to GND (s2g), undervoltage chargepump (uv_cp) DIAG0 always shows the reset-status, i.e. is active low during reset condition.	0
6	diag0_otpw	(only with SD_MODE=1) 1: Enable DIAG0 active on driver over temperature prewarning (otpw)	0
7	diag0_stall	(with SD_MODE=1) 1: Enable DIAG0 active on motor stall (set TCOOLTHRS before using this feature) diag0_step (with SD_MODE=0) 0: DIAG0 outputs interrupt signal 1: Enable DIAG0 as STEP output (dual edge triggered steps) for external STEP/DIR driver	0
8	diag1_stall	(with SD_MODE=1) 1: Enable DIAG1 active on motor stall (set TCOOLTHRS before using this feature) diag1_dir (with SD_MODE=0) 0: DIAG1 outputs position compare signal 1: Enable DIAG1 as DIR output for external STEP/DIR driver	0

GCONF - General Configuration Register			
Bit	Name	Description	Default
9	diag1_index	(only with SD_MODE=1) 1: Enable DIAG1 active on index position (microstep look up table position 0)	0
10	diag1_onstate	(only with SD_MODE=1) 1: Enable DIAG1 active when chopper is on (for the coil which is in the second half of the fullstep)	0
11	diag1_steps_skipped	(only with SD_MODE=1) 1: Enable output toggle when steps are skipped in dcStep mode (increment of LOST_STEPS). Do not enable in conjunction with other DIAG1 options.	0
12	diag0_int_pushpull	0: SWN_DIAG0 is open collector output (active low) 1: Enable SWN_DIAG0 push pull output (active high)	0
13	diag1_poscomp_pushpull	0: SWP_DIAG1 is open collector output (active low) 1: Enable SWP_DIAG1 push pull output (active high)	0
14	small_hysteresis	0: Hysteresis for step frequency comparison is 1/16 1: Hysteresis for step frequency comparison is 1/32	0
15	stop_enable	0: Normal operation 1: Emergency stop: ENCA_DCIN stops the sequencer when tied high (no steps become executed by the sequencer, motor goes to standstill state).	0
16	direct_mode	0: Normal operation 1: Motor coil currents and polarity directly programmed via serial interface: Register XTARGET (0x2D) specifies signed coil A current (bits 8..0) and coil B current (bits 24..16). In this mode, the current is scaled by IHOLD setting. Velocity based current regulation of voltage PWM is not available in this mode. The automatic voltage PWM current regulation will work only for low stepper motor velocities.	0
17	test_mode	0: Normal operation 1: Enable analog test output on pin ENCN_DCO. IHOLD[1..0] selects the function of ENCN_DCO: 0…2: T120, DAC, VDDH Attention: Not for user, set to 0 for normal operation!	0

Table 15: GCONF - General Configuration Register

ATTENTION:

Functions marked in grey are not allowed to be changed. These functions are fixed by hardware and should not be changed.

CHOPCONF- Chopper Configuration				
Bit	Name	Function	Description	Default
31	-	reserved	-	0
30	diss2g	Short to GND protection disable	0: Short to GND protection is on 1: Short to GND protection is disabled	0

CHOPCONF- Chopper Configuration				
Bit	Name	Function	Description	Default
29	dedge	enable double edge step pulses	1: Enable step impulse at each step edge to reduce step frequency requirement.	0
28	intpol	interpolation to 256 microsteps	1: The actual microstep resolution (MRES) becomes extrapolated to 256 microsteps for smoothest motor operation (useful for Step/Dir operation, only)	1
27	mres3	MRES micro step resolution	▪ %0000: Native 256 microstep setting. Normally use this setting with the internal motion controller. ▪ %0001 … %1000: 128, 64, 32, 16, 8, 4, 2, FULLSTEP Reduced microstep resolution esp. for Step/Dir operation. The resolution gives the number of microstep entries per sine quarter wave. The driver automatically uses microstep positions which result in a symmetrical wave, when choosing a lower microstep resolution. step width=2^MRES [microsteps]	0
26	mres2			0
25	mres1			0
24	mres0			0
23	sync3	SYNC PWM synchronization clock	This register allows synchronization of the chopper for both phases of a two phase motor in order to avoid the occurrence of a beat, especially at low motor velocities. It is automatically switched off above VHIGH. ▪ %0000: Chopper sync function chopSync off ▪ %0001 … %1111: Synchronization with fSYNC = fCLK/(sync*64) Hint: Set TOFF to a low value, so that the chopper cycle is ended, before the next sync clock pulse occurs. Set for the double desired chopper frequency for chm=0, for the desired base chopper frequency for chm=1.	0
22	sync2			0
21	sync1			0
20	sync0			0
19	vhighchm	high velocity chopper mode	This bit enables switching to chm=1 and fd=0, when VHIGH is exceeded. This way, a higher velocity can be achieved. Can be combined with vhighfs=1. If set, the TOFF setting automatically becomes doubled during high velocity operation in order to avoid doubling of the chopper frequency.	0
18	vhighfs	high velocity fullstep selection	This bit enables switching to fullstep, when VHIGH is exceeded. Switching takes place only at 45° position. The fullstep target current uses the current value from	0

CHOPCONF- Chopper Configuration				
Bit	Name	Function	Description	Default
			the microstep table at the 45° position.	
17	vsense	sense resistor voltage based current scaling	0: Low sensitivity, high sense resistor voltage 1: High sensitivity, low sense resistor voltage	0
16	tbl1	TBL blank time select	%00 … %11: Set comparator blank time to 16, 24, 36 or 54 clocks Hint: %01 or %10 is recommended for most applications	1
15	tbl0			0
14	chm	chopper mode	▪ 0: Standard mode (spreadCycle) ▪ 1: Constant off time with fast decay time. Fast decay time is also terminated when the negative nominal current is reached. Fast decay is after on time.	0
13	rndtf	random TOFF time	▪ 0: Chopper off time is fixed as set by TOFF ▪ 1: Random mode, TOFF is random modulated by dN_{CLK}= -12 … +3 clocks.	0
12	disfdcc	fast decay mode	chm=1: disfdcc=1 disables current comparator usage for termi-nation of the fast decay cycle	0
11	fd3	TFD [3]	chm=1: MSB of fast decay time setting TFD	0
10	hend3	HEND hysteresis low value OFFSET sine wave offset	▪ chm=0 %0000 … %1111: Hysteresis is -3, -2, -1, 0, 1, …, 12 (1/512 of this setting adds to current setting) This is the hysteresis value which becomes used for the hysteresis chopper. ▪ chm=1 %0000 … %1111: Offset is -3, -2, -1, 0, 1, …, 12 This is the sine wave offset and 1/512 of the value becomes added to the absolute value of each sine wave entry.	0
9	hend2			0
8	hend1			1
7	hend0			1
6	hstrt2	- HSTRT hysteresis start value added to HEND - TFD [2..0] fast decay time setting	▪ chm=0 %000 … %111: Add 1, 2, …, 8 to hysteresis low value HEND (1/512 of this setting adds to current setting) Attention: Effective HEND+HSTRT ≤ 16. Hint: Hysteresis decrement is done each 16 clocks ▪ chm=1	1
5	hstrt1			0
4	hstrt0			1

CHOPCONF- Chopper Configuration				
Bit	Name	Function	Description	Default
			Fast decay time setting (MSB: fd3): %0000 … %1111: Fast decay time setting TFD with N_{CLK}= 32*HSTRT (%0000: slow decay only)	
3	toff3	TOFF off time and driver enable	Off time setting controls duration of slow decay phase N_{CLK}= 12 + 32*TOFF %0000: Driver disable, all bridges off %0001: 1 – use only with TBL ≥ 2 %0010 … %1111: 2 … 15 General enable for the motor driver, the actual value does not influence stealthChop ▪ 0: Driver off ▪ 1 … 15: Driver enabled	0
2	toff2			1
1	toff1			0
0	toff0			1

Table 16: CHOPCONF- Chopper Configuration

COOLCONF- Smart Energy Control				
Bit	Name	Function	Description	Default
...	-	reserved	-	0
24	sfilt	stallGuard2 filter enable	▪ 0: Standard mode, high time resolution for stallGuard2 ▪ 1: Filtered mode, stallGuard2 signal updated for each four fullsteps (resp. six fullsteps for 3 phase motor) only to compensate for motor pole tolerances	0
23	-	reserved		0
22	sgt6	stallGuard2 threshold value	This signed value controls stallGuard2 level for stall output and sets the optimum measurement range for readout. A lower value gives a higher sensitivity. Zero is the starting value working with most motors. -64 to +63: A higher value makes stallGuard2 less sensitive and requires more torque to indicate a stall.	0
21	sgt5			0
20	sgt4			0
19	sgt3			0
18	sgt2			0
17	sgt1			0
16	sgt0			0
15	seimin	minimum current for smart current control	▪ 0: 1/2 of current setting (IRUN) ▪ 1: 1/4 of current setting (IRUN)	0
14	sedn1	current down step speed	▪ %00: For each 32 stallGuard2 values decrease by one ▪ %01: For each 8 stallGuard2 values decrease by one ▪ %10: For each 2 stallGuard2 values decrease by one ▪ %11: For each stallGuard2 value decrease by one	0
13	sedn0			0
12	-	reserved		0
11	semax3	stallGuard2 hysteresis	If the stallGuard2 result is equal to or	0

COOLCONF- Smart Energy Control				
Bit	Name	Function	Description	Default
10	semax2	value for smart current control	above (SEMIN+SEMAX+1)*32, the motor current becomes decreased to save energy. %0000 … %1111: 0 … 15	0
9	semax1			0
8	semax0			0
7	-	reserved		0
6	seup1	current up step width	Current increment steps per measured stallGuard2 value %00 … %11: 1, 2, 4, 8	0
5	seup0			0
4	-	reserved		0
3	semin3	minimum stallGuard2 value for smart current control and smart current enable	If the stallGuard2 result falls below SEMIN*32, the motor current becomes increased to reduce motor load angle. ▪ %0000: smart current control coolStep off ▪ %0001 … %1111: 1 … 15	0
2	semin2			0
1	semin1			0
0	semin0			0

Table 17: COOLCONF- Smart Energy Control

PWMCONF- Voltage PWM Mode stealthChop				
Bit	Name	Function	Description	Default
...	-	reserved	-	0
21	freewheel1	Allows different standstill modes	Stand still option when motor current setting is zero (I_HOLD=0). ▪ %00: Normal operation ▪ %01: Freewheeling ▪ %10: Coil shorted using LS drivers ▪ %11: Coil shorted using HS driverssetting is zero (I_HOLD=0). ▪ %00: Normal operation ▪ %01: Freewheeling ▪ %10: Coil shorted using LS drivers ▪ %11: Coil shorted using HS drivers	0
20	freewheel0			0
19	pwm_ symmetric	Force symmetric PWM	▪ 0: The PWM value may change within each PWM cycle (standard mode) ▪ 1: A symmetric PWM cycle is enforced	1
18	pwm_ autoscale	PWM automatic amplitude scaling	▪ 0: User defined PWM amplitude. The current settings have no influence. ▪ 1: Enable automatic current control Attention: When using a user defined sine wave table, the amplitude of this sine wave table should not be less than 244. Best results are obtained with 247 to 252 as peak values.	0
17	pwm_freq1	PWM frequency selection	▪ %00: fPWM=1/1024 fCLK ▪ %01: fPWM=1/683 fCLK ▪ %10: fPWM=1/512 fCLK ▪ %11: fPWM=1/410 fCLK	1
16	pwm_freq0			0

PWMCONF- Voltage PWM Mode stealthChop				
Bit	Name	Function	Description	Default
15	PWM_ GRAD	User defined amplitude (gradient) or regulation loop gradient	▪ pwm_ autoscale=0 Velocity dependent gradient for PWM amplitude: PWM_GRAD * 256 / TSTEP is added to PWM_AMPL ▪ pwm_ autoscale=1 User defined maximum PWM amplitude change per half wave (1 to 15)	0
14				0
13				0
12				0
11				1
10				0
9				0
8				1
7	PWM_ AMPL	User defined amplitude (offset)	▪ pwm_ autoscale=0 User defined PWM amplitude offset (0-255) The resulting amplitude (limited to 0…255) is: PWM_AMPL + PWM_GRAD * 256 / TSTEP ▪ pwm_ autoscale=1 User defined maximum PWM amplitude when switching back from current chopper mode to voltage PWM mode (switch over velocity defined by TPWMTHRS). Do not set too low values, as the regulation cannot measure the current when the actual PWM value goes below a setting specific value. Settings above 0x40 recommended.	0
6				0
5				0
4				0
3				0
2				0
1				0
0				0

Table 18: PWMCONF- Voltage PWM Mode stealthChop

Velocity Dependent Driver Feature Control Register				
R/W	Bit Qty	Name	Description	Default
W	8	TPOWER_DOWN	TPOWERDOWN sets the delay time after stand still (stst) of the motor to motor current power down. Time range is about 0 to 4 seconds. 0…((2^8)-1) * 2^18 t_{CLK}	0x00000007
R	20	TSTEP	Actual measured time between two 1/256 microsteps derived from the step input frequency in units of 1/fCLK. Measured value is (2^20)-1 in case of overflow or stand still. All TSTEP related thresholds use a hysteresis of 1/16 of the compare value to compensate for jitter in the clock or the step frequency. The flag small_hysteresis modifies the hysteresis to a smaller value of 1/32. (Txxx*15/16)-1 or (Txxx*31/32)-1 is used as a second compare value for each comparison value.	--

Velocity Dependent Driver Feature Control Register				
R/W	Bit Qty	Name	Description	Default
			This means, that the lower switching velocity equals the calculated setting, but the upper switching velocity is higher as defined by the hysteresis setting. When working with the motion controller, the measured TSTEP for a given velocity V is in the range (224 / V) ≤ TSTEP ≤ 224 / V - 1. In dcStep mode TSTEP will not show the mean velocity of the motor, but the velocities for each microstep, which may not be stable and thus does not represent the real motor velocity in case it runs slower than the target velocity.	
W	20	TPWMTHRS	This is the upper velocity for stealthChop voltage PWM mode. TSTEP ≥ TPWMTHRS - stealthChop PWM mode is enabled, if configured - dcStep is disabled	0
W	20	TCOOLTHRS	This is the lower threshold velocity for switching on smart energy coolStep and stallGuard feature. (unsigned) Set this parameter to disable coolStep at low speeds, where it cannot work reliably. The stop on stall function (enable with sg_stop when using internal motion controller) and the stall output signal become enabled when exceeding this velocity. In non-dcStep mode, it becomes disabled again once the velocity falls below this threshold. ▪ TCOOLTHRS ≥ TSTEP ≥ THIGH: - coolStep is enabled, if configured - stealthChop voltage PWM mode is disabled ▪ TCOOLTHRS ≥ TSTEP - Stop on stall and stall output signal is enabled, if configured	0x00000000
W	20	THIGH	This velocity setting allows velocity dependent switching into a different chopper mode and fullstepping to maximize torque. (unsigned) The stall detection feature becomes switched off for 2-3 electrical periods whenever passing THIGH threshold to compensate for the effect of switching modes. TSTEP ≤ THIGH: ▪ coolStep is disabled (motor runs with normal current scale) ▪ stealthChop voltage PWM mode is disabled ▪ If vhighchm is set, the chopper switches to chm=1 with TFD=0 (constant off time with slow decay, only). ▪ chopSync2 is switched off (SYNC=0)	0x00000000

Velocity Dependent Driver Feature Control Register				
R/W	Bit Qty	Name	Description	Default
			▪ - If vhighfs is set, the motor operates in fullstep mode and the stall detection becomes switched over to dcStep stall detection.	

Table 19: Velocity Dependent Driver Feature Control Register

Ramp Generator driver Feature Control Register				
R/W	Bit Qty	Name	Description	Default
W	23	VDCMIN	Automatic commutation dcStep becomes enabled above velocity VDCMIN (unsigned) (only when using internal ramp generator, not for STEP/DIR interface – in STEP/DIR mode, dcStep becomes enabled by the external signal DCEN) In this mode, the actual position is determined by the sensor-less motor commutation and becomes fed back to XACTUAL. In case the motor becomes heavily loaded, VDCMIN also is used as the minimum step velocity. Activate stop on stall (sg_stop) to detect step loss. ▪ 0: Disable, dcStep off ▪ |VACT| ≥ VDCMIN ≥ 256: - Triggers the same actions as exceeding THIGH setting. - Switches on automatic commutation dcStep Hint: Also set DCCTRL parameters in order to operate dcStep. (Only bits 22… 8 are used for value and for comparison)	0x00000000

Table 20: Ramp Generator driver Feature Control Register

Motor Drive Register				
R/W	Bit Qty	Name	Description	Default
W	24	DCCTRL	dcStep (DC) automatic commutation configuration register (enable via pin DCEN or via VDCMIN): ▪ bit 9… 0: DC_TIME: Upper PWM on time limit for commutation (DC_TIME * 1/fCLK). Set slightly above effective blank time TBL. ▪ bit 23… 16: DC_SG: Max. PWM on time for step loss detection using dcStep stallGuard2 in dcStep mode. (DC_SG * 16/fCLK) Set slightly higher than DC_TIME/16 0=disable Attention: Using a higher microstep resolution or interpolated operation, dcStep delivers a better	0x00000000

Motor Drive Register				
R/W	Bit Qty	Name	Description	Default
			stallGuard signal. DC_SG is also available above VHIGH if vhighfs is activated. For best result also set vhighchm.	
R	8	PWM_SCALE	Actual PWM amplitude scaler (255=max. Voltage) In voltage mode PWM, this value allows to detect a motor stall.	--

Table 21: Motor Drive Register

DRV_ STATUS - Drive Error Flags and stallGuard2 Value				
Bit	Name	Function	Description	Default
31	stst	standstill indicator	This flag indicates motor stand still in each operation mode. This occurs 2^20 clocks after the last step pulse.	--
30	olb	open load indicator phase B	1: Open load detected on phase A or B. Hint: This is just an informative flag. The driver takes no action upon it. False detection may occur in fast motion and standstill. Check during slow motion, only.	--
29	ola	open load indicator phase A		--
28	s2gb	short to ground indicator phase B	1: Short to GND detected on phase A or B. The driver becomes disabled. The flags stay active, until the driver is disabled by software (TOFF=0) or by the ENN input.	--
27	s2ga	short to ground indicator phase A		--
26	otpw	overtemperature pre-warning flag	1: Overtemperature pre-warning threshold is exceeded. The overtemperature pre-warning flag is common for both bridges.	--
25	ot	overtemperature flag	1: Overtemperature limit has been reached. Drivers become disabled until otpw is also cleared due to cooling down of the IC. The overtemperature flag is common for both bridges.	--
24	stallGuard	stallGuard2 status	1: Motor stall detected (SG_RESULT=0) or dcStep stall in dcStep mode.	--
23	-	reserved	Ignore these bits	--
22				
21				
20	CS ACTUAL	actual motor current / smart energy current	Actual current control scaling, for monitoring smart energy current scaling controlled via settings in register COOLCONF, or for monitoring the function of the automatic current scaling.	--
19				
18				
17				
16				
15	fsactive	full step active indicator	1: Indicates that the driver has switched to fullstep as defined by chopper mode settings and velocity thresholds.	--
14	-	reserved	Ignore these bits	--
13				

DRV_ STATUS - Drive Error Flags and stallGuard2 Value				
Bit	Name	Function	Description	Default
12				
11				
10				
9	SG_ RESULT	stallGuard2 result respectively PWM on time for coil A in stand still for motor temperature detection	Mechanical load measurement: The stallGuard2 result gives a means to measure mechanical motor load. A higher value means lower mechanical load. A value of 0 signals highest load. With optimum SGT setting, this is an indicator for a motor stall. The stall detection compares SG_RESULT to 0 in order to detect a stall. SG_RESULT is used as a base for coolStep operation, by comparing it to a programmable upper and a lower limit. It is not applicable in stealthChop mode. SG_RESULT is ALSO applicable when dcStep is active. stallGuard2 works best with microstep operation. Temperature measurement: In standstill, no stallGuard2 result can be obtained. SG_RESULT shows the chopper on-time for motor coil A instead. If the motor is moved to a determined microstep position at a certain current setting, a comparison of the chopper on-time can help to get a rough estimation of motor temperature. As the motor heats up, its coil resistance rises and the chopper on-time increases.	--
8				--
7				--
6				--
5				--
4				--
3				--
2				--
1				--
0				--

Table 22: DRV_ STATUS - Drive Error Flags and stallGuard2 Value

$f_{CLK} = 2^{24}$ [Hz]