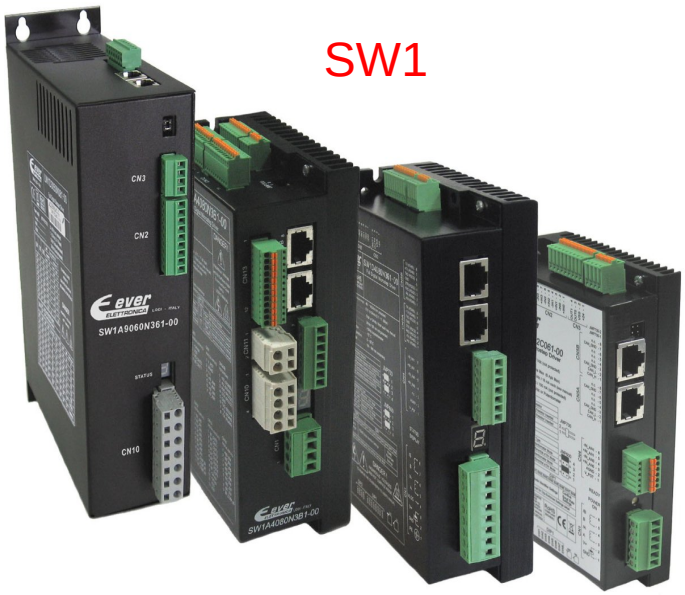
CANopen & EtherCAT DSP402 Protocol Specification for 'Slim Line & Titanio Series Drives '

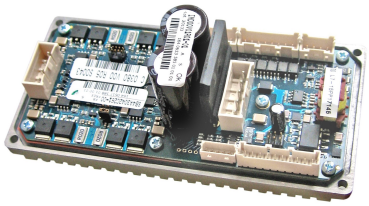
Release : 0.19 Build 00 (12-02-2020)

SW1

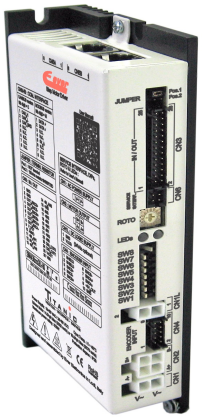


SM1A / SM4A

DMD105 / SB4A

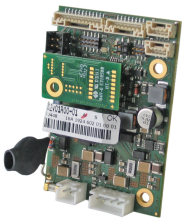


SW4

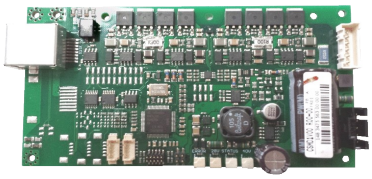


SW5

SB4D



CSMD1



IMPORTANT NOTICE

EVER Elettronica
VIA DEL Commercio , 2/4 Loc. S. Grato Z.I.
26900 – LODI - ITALY
Tel. ++39(0)371412318 Fax ++39(0)371412367
E-mail: infoever@everelettronica.it
Web: www.everelettronica.com

Release History:

Release	Date	Author	Description
0.1 Build 00	18-12-2012	Rota L.	First Issue.
0.2 Build 00	07/01/2013	Rota L.	- Added bit#7 (Disable_Digital_Outputs_FW_Handling) for 'Drive_Working_Settings' - Corrected 'Quick_stop_option_code' and 'Halt_option_code' description - Added information concerning 'Emergency codes' description - Removed 'Min_Profile_Velocity', 'Max_Profile_Velocity', 'Profile_Velocity', 'Profile_Acceleration', 'Profile_Deceleration' objects - Added Mapping #4 and Mapping #5 for Transmit PDO #2
0.3 Build 00	22/02/2013	Rota L.	- Added 'Profile Position Mode' and related objects - Added 'Homing Mode' and related objects - Added 'Profile Velocity Mode' and related objects - Removed conversion factor objects and fixed values for Position,Velocity,Acceleration dimension and notation index. - Added 'Min_Profile_Velocity', 'Profile_Velocity', 'Profile_Acceleration', 'Profile_Deceleration' objects - Added Mapping #5 Receive PDO #2 - Added Mapping #6 Transmit PDO #2
0.4 Build 00	27/03/2013	Rota L.	- Corrected 'Drive_Canopen_config' default value description. - Added 'Forward_Limit_Switch_Check','Backward_Limit_Switch_Check' bits description in the 'Drive_Working_Settings' object - Added Emergency codes : Forward_limit_switch_reached (0x64), Backward_limit_switch_reached (0x65) - Added 'Drive_Inputs_Setting' and 'Drive_Inputs_Level' objects - Added Forward_Limit_switch and Backward_Limit_switch in the D Appendix table - Added 'Fault_reaction_option_code' object - Added TPDO#2 mapping #7
0.5 Build 00	27/05/2013	Rota L.	- Added 'Max_Profile_Velocity' - Added option 6 for 605A object - Modified Profile_Position_Mode description
0.6 Build 00	03/03/2014	Taglietti A.	- Changed in EMCY Error_Code value - Renamed 6063.0 object - Removed EMCY Error 63H, Changed description 6065.0 , 6066.0 objects - Added csp, csv modalities - Added EtherCAT description
0.7 Build 00	04/07/2014	Rota L.	- Modified default value for 1400.2h,1401.2h,1800.2h,1801.2h objects
0.8 Build 00	10/10/2016	Rota L. Taglietti A.	- Modified 'Table 3' - Added TPDO#2 mapping #8 - Added RPDO#1, RPDO#2, TPDO#1, TPDO#2 Mapping - Added csp and csv modalities in the list of §3.1 - Added description for CSMD1,SB4, SM4, SW4,SW5 drives - Added Homing methods 1,2,17,18 for CSMD1,SB4, SM4, SW4,SW5 drives - Added 'Velocity mode' and related objects for CSMD1,SB4, SM4, SW4,SW5 drives
0.9 Build 00	22/12/2016	Rota L. Taglietti A.	- Added description of 'Drive_CANopen_Config_Extended' object (2200.1Ah) - Added Receive RPDO#3, RPDO#4, TPDO#3, TPDO#4 description - Added description of 'Digital Inputs' object (60FD.0h) - Added 'Step_angle' resolution with 65536 steps per revolution - Added description of 1402.1h,1402.2h,1403.1h,1403.2h,1802.1h, 1802.2h, 1802.5h, 1803.1h, 1803.2h, 1803.5h objects - changed §2.1, added objects 100Ch, 100Dh
0.10 Build 00	23/11/2017	Rota L.	- Added 'Touch Probe' functionality
0.11 Build 00	24/01/2018	Rota L. Taglietti A.	- Modified 'Factor Group' description . - Added objects related to 'Factor Group'. - Added examples for Velocity, position, acceleration, deceleration unit settings. - Modified description of objects related to Velocity, position, acceleration units. - Modified description of objects related to Position, Acceleration and Velocity Unit. - Added Feedback feature description. - Added objects related to Feedback feature (closed loop). - Added error 5Fh (Feedback error) to emergency codes - Modified description of objects 2012.1h,2012.2h,2010.0h,6067.0h - Added 60EF.0h object - Added description of bit4 (Feedback_Motor_check) and bit13 (Feedback_Motor_action) in 'Drive_Working_Settings' object - Added description of bit9 (Motor R,L detection) in 'Drive_Working_Settings_Extended' object - Added description of object 2005.4h, 2005.6h, 2005.7h, 2005.8h, 2005.9h, 6078.0h - Added description of 'Motor R,L detecting' - Update Ethercat chapter
0.12 Build 00	07/02/2018	Rota L.	- Added 59A2.0h object. - Updated 60F4 object
0.13 Build 00	09/07/2018	Rota L.	- Added 2084.0h object
0.14 Build 00	30/11/2018	Rota L.	- Modified §3.2
0.15 Build 00	13/12/2018	Rota L.	- Added 'Store supported' in NVRAM for objects 60C2.1h, 606A.0h, 60D0.1h, 60D0.2h, 2082.0h, 2083.0h, 60C3.2h, 2081.0h, 607C.0h, 6099.1h, 6099.2h, 609A.0h, 6098.0h, 2080.0h - Modified description of 60FD.0h object.

Release	Date	Author	Description
0.16 Build 00	11/01/2019	Rota L.	- Added ‘Store supported’ in NVRAM for object 605A.0h
0.17 Build 00	12/09/2019	Rota L.	- Added ‘Store supported’ in NVRAM for objects 605E.0h, 2084.0h
0.18 Build 00	21/10/2019	Taglietti A.	- Added bit #2 Drive_Working_Settings object - Added bit #2 Drive_Inputs_Level object - Added bits #8÷11 Drive_Input_Settings object - Added bit #4 DS402_Working_Settings object
0.19 Build 00	12/02/2020	Rota L.	- Added Homing methods 23,24,25,26,27,28. - Added ‘BiSS Encoder’ functions and objects : 2A00.0h, 2A02.0h, 2A04.0h, 2A06.0h, 2A08.0h, 2A0A.0h. - Modified ‘Sensor_Selection_code’ object description : added selection for ‘BiSS Encoder’.

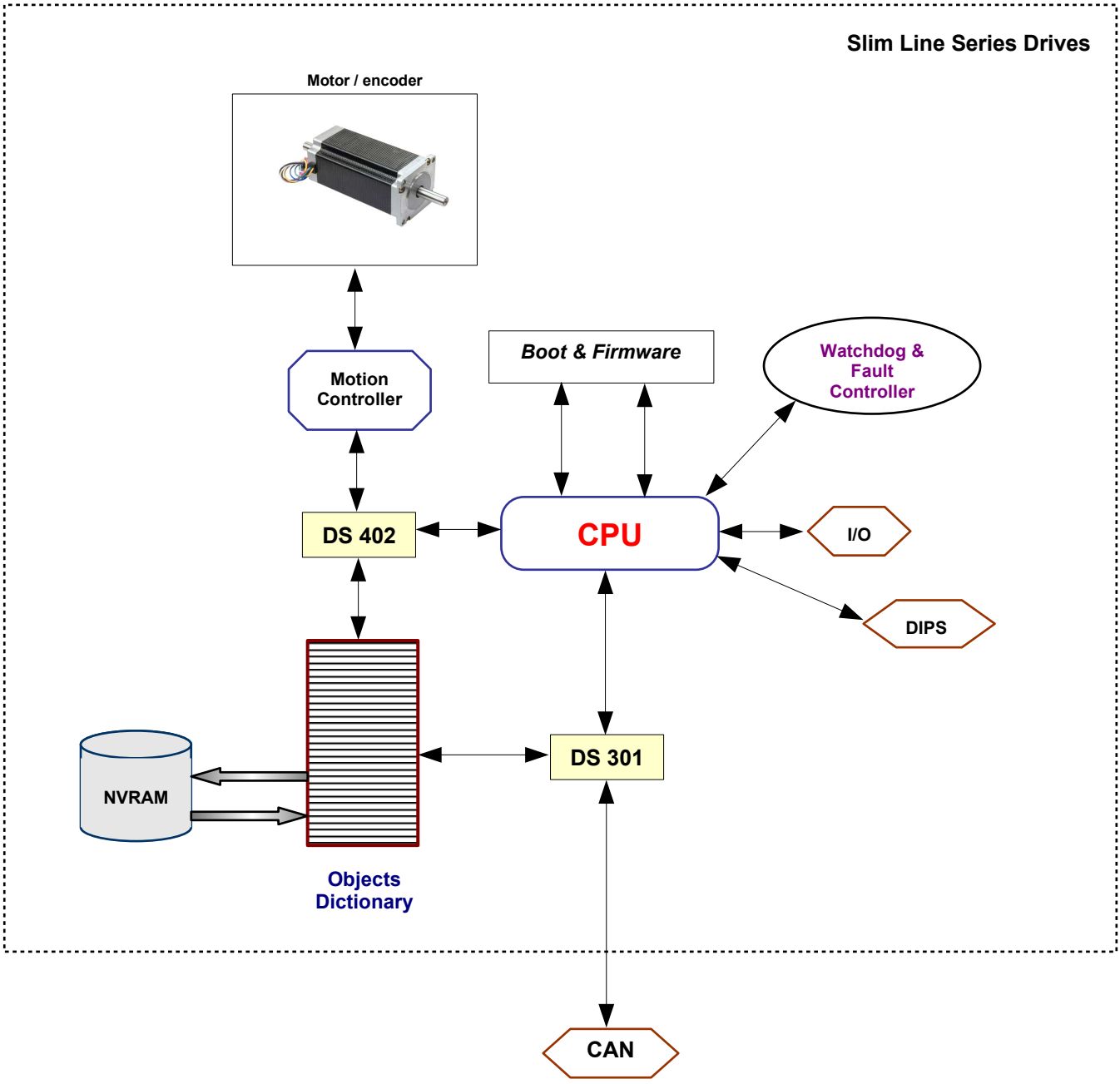
Related Publications

CiA DS 301 V4.02	**CANopen Application Layer and Communication Profile**
CiA DSP 402 V2.0	**CANopen Device Profile for Drives and Motion Control**
IEC-61800-7-201:2015	**Adjustable speed electrical power drive systems**

1.0 Introduction

The Slim family drives use a subset of the standard CANopen protocol to provide access to whole drive parameters. Several standard CANopen functions codes are supported as described in the CiA DS301 and CiA DSP402. The Slim family drives are CANopen slave devices and then they need a CANopen master system (PC, PLC, etc.) to be configured and managed by CAN bus.

1.1 Slim Drives Working Logic



2.0 CANopen Protocol

The CANopen protocol is one of the most common CAN protocols. Since 1995 the CANopen specification is handed over to CAN in Automation (CiA) international users and manufacturers group. The CANopen Device Specification version 4.01 has been accepted by the European standardization authorities as EN 50325-4.

The main concept of CANopen is based on use of an object dictionary. The object dictionary is essentially a grouping of objects accessible via the network in an ordered pre-defined fashion. To access to these objects two methods are used: SDO & PDO that are explained further in this manual.
The overall layout of the standard Object Dictionary is :

Index	Object
0000h - 0FFFh	Data definition / reserved
1000h - 1FFFh	Communication profile area (DS301)
2000h - 5FFFh	Manufacturer specific area
6000h - 9FFFh	Standardized device profile area (DSP402)
A000h - FFFFh	Other profile / reserved

Each object within the dictionary is addressed using a 16-bit index. In case of a simple variable (VAR) the index directly references the value. In case of records (RECORD) and arrays (ARRAY), the index addresses the whole data structure. To allow individual elements of structures of data to be accessed, a sub-index has been defined. For single object dictionary entries such as an unsigned8, boolean, integer32,etc the value for the sub-index is always zero. For complex object dictionary entries such as arrays or records with multiple data fields , the sub-index refers to fields within a data-structure pointed to by the main index.

2.1 CANopen Protocol Parameters

For SW1,DMD105,SM1A Drives (Slim Line Drives) :

CANopen Specifications	
CANopen Functionality	Slave
Device Id Number	1 to 127
Baud Rate Supported (Kbits)	1000,500,250,125
NMT	Slave
Server SDOs	1 (Standard)
Client SDOs	No
Receive PDOs	2
Transmit PDOs	2
PDO Mapping	Static
Emergency Telegram	Yes
Nodeguarding	No
Heartbeat	Yes
Sync supported	Yes

For CSMD1,SB4,SW4,SM4,SW5 Drives (Titanio Drives) with firmware V00r74 or superior :

CANopen Specifications	
CANopen Functionality	Slave
Device Id Number	1 to 127
Baud Rate Supported (Kbits)	1000,500,250,125
NMT	Slave
Server SDOs	1 (Standard)
Client SDOs	No
Receive PDOs	4
Transmit PDOs	4
PDO Mapping	Dynamic
Emergency Telegram	Yes
Nodeguarding	Yes
Heartbeat	Yes
Sync supported	Yes

2.2 Baud Rate & Node Id Selection on SW1 Drives

Drive with dip switches :

The drive's id number and baud rate are selected via dip switches as follows:

DIP2				DIP1							
U1	U0	ID6	ID5	ID4	ID3	ID2	ID1	ID0	BD2	BD1	BD0
1	2	3	4	1	2	3	4	5	6	7	8

Baud Rate Selection:

BD2	BD1	BD0	Baud Rate
OFF	OFF	OFF	1 M
OFF	OFF	ON	500 K
OFF	ON	OFF	250 K
OFF	ON	ON	125 K
ON	OFF	OFF	reserved
ON	OFF	ON	reserved
ON	ON	OFF	reserved
ON	ON	ON	reserved

Node Id Selection:

ID6	ID5	ID4	ID3	ID2	ID1	ID0	Node Id #
OFF	OFF	OFF	OFF	OFF	OFF	OFF	Not allowed
OFF	OFF	OFF	OFF	OFF	OFF	ON	1
OFF	OFF	OFF	OFF	OFF	ON	OFF	2
OFF	OFF	OFF	OFF	OFF	ON	ON	3
OFF	OFF	OFF	OFF	ON	OFF	OFF	4
OFF	OFF	OFF	OFF	ON	OFF	ON	5
OFF	OFF	OFF	OFF	ON	ON	OFF	6
OFF	OFF	OFF	OFF	ON	ON	ON	7
OFF	OFF	OFF	ON	OFF	OFF	OFF	8
OFF	OFF	OFF	ON	OFF	OFF	ON	9
OFF	OFF	OFF	ON	OFF	ON	OFF	10
OFF	OFF	OFF	ON	OFF	ON	ON	11
OFF	OFF	OFF	ON	ON	OFF	OFF	12
OFF	OFF	OFF	ON	ON	OFF	ON	13
OFF	OFF	OFF	ON	ON	ON	OFF	14
OFF	OFF	OFF	ON	ON	ON	ON	15
OFF	OFF	ON	OFF	OFF	OFF	OFF	16
OFF	OFF	ON	OFF	OFF	OFF	ON	17
OFF	OFF	ON	OFF	OFF	ON	OFF	18
OFF	OFF	ON	OFF	OFF	ON	ON	19
OFF	OFF	ON	OFF	ON	OFF	OFF	20
OFF	OFF	ON	OFF	ON	OFF	ON	21
OFF	OFF	ON	OFF	ON	ON	OFF	22
OFF	OFF	ON	OFF	ON	ON	ON	23
OFF	OFF	ON	ON	OFF	OFF	OFF	24
XX	XX	XX	XX	XX	XX	XX	....
ON	ON	ON	ON	ON	ON	ON	127

Note: The user dips U0 and U1 are available for the specific application. The EtherCAT drives do not support NodeId and baud rate identification by dips, since is automatically set by the master.

Drive with rotoswitches :

The drive's id number and baud rate are selected via rotoswitches as follows:

Baud Rate Selection:

(SW4) ROTOSWITCH POSITION	Baud Rate
0	1 M
1	500 K
2	250 K
3	125 K
4	100 K
5	50 K
6	50 K
7	50 K
8	reserved
9	reserved
A	reserved
B	reserved
C	reserved
D	reserved
E	reserved
F	reserved

Node Id Selection:

ROTOSWITCH POSITION		Node Id #
(SW3) NodeId High	(SW2) NodeId Low	
0	0	Not allowed
0	1	1
0	2	2
0	3	3
0	4	4
0	5	5
0	6	6
0	7	7
0	8	8
0	9	9
0	A	10
0	B	11
0	C	12
0	D	13
0	E	14
0	F	15
1	0	16
1	1	17
XX	XX	....
7	D	125
7	E	126
7	F	127
8	0	Not allowed
8	1	Not allowed
8	2	Not allowed
XX	XX	....
F	E	Not allowed
F	F	Not allowed

User Setting Selection:

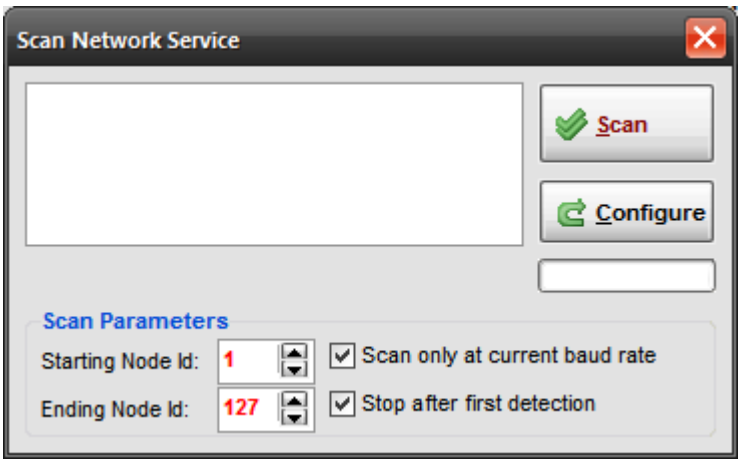
(SW1) ROTOSWITCH POSITION	TBD	TBD	U1(1)	U0(2)
0	0	0	0	0
1				1
2			1	0
3				1
4		1	0	0
5				1
6			1	0
7				1
8	1	0	0	0
9				1
A			1	0
B				1
C		1	0	0
D				1
E			1	0
F				1

Note : (1) This function is equivalent to dip-switch setting dip2.1 (0 = OFF ; 1 = ON)
(2) This function is equivalent to dip-switch setting dip2.2 (0 = OFF ; 1 = ON)

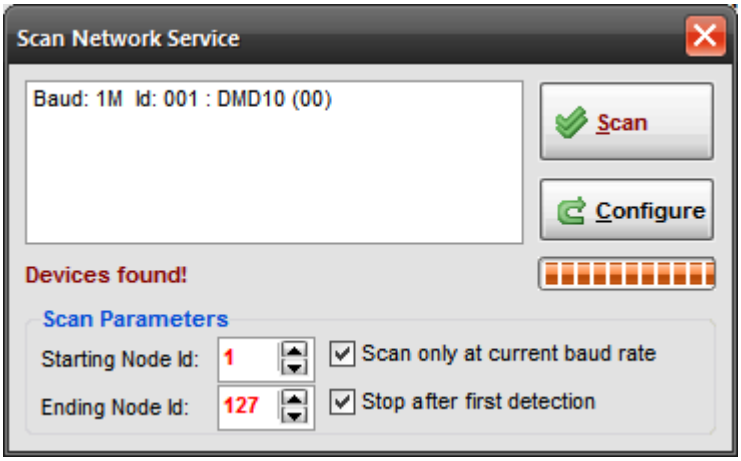
2.3 Baud Rate & Node Id Selection on SM1A/SM4A/SB4 Drives

The SM1A drives must be configured by means of Slim Line Drives CAN Monitor Software Tool. The drives factory settings is baud rate = 500 Kbit and NodeId = 1. Follows the steps to perform to change the default settings:

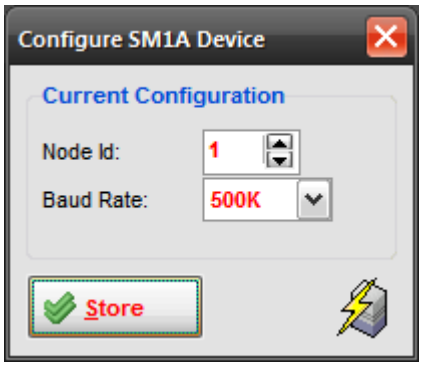
1. Open Slim Line Drives CAN Monitor Software Tool.
2. From the main window select the *'Tools/Scan & Config Service'* menu item. The following window will appear:



3. If not sure on drive's baud rate deselect the *"Scan only at current baud rate"* check box.
4. Press on ***"Scan"*** button.
5. If the drive has been detected within the "Starting Node Id" and "Ending Node Id" its system code will be added to the list box.



6. Select the device and press on ***"Configure"*** button. The following window will appear:



7. Change the Node Id and Baud Rate settings as desired and press on ***"Store"*** button.
8. The new settings will be used at the next drive's switch on.

Instead of using the Slim Line Drives CAN Monitor Software Tool, the Baud Rate & Node Id can be changed directly writing on the objects (4000.7 & 4000.8) according to the following instructions:

1. Write using SDO service 2 bytes in the object 4000.7 (Node Id) keeping the high byte equal to 0xAA and setting in the low byte the new Id. (example to set the Node Id = 5 write 0xAA05)
2. Write using SDO service 2 bytes in the object 4000.8 (Baud Rate) keeping the high byte equal to 0x55 and setting in the low byte the new baud rate according to the following table (example to set the baud rate = 500K write 0x5501):

Value	Baud Rate
0	1M
1	500K
2	250K
3	125K

3. After having written the new value it is necessary to wait about 1 second to permit to the system to store the new values in NVRAM.
4. The new BaudRate & NodeId will be effective at the next drive switch on.

2.4 CANopen SDO (Service Data Object)

Service Data Objects are used to establish a peer to peer connection between two CANopen devices. This kind of connection is based on a Client/Server based mechanism.
The SDO server is the device that is serving the object dictionary to which the access is required.
The SDO client is the device that wants to access the object dictionary of a specific device.
The SDO service is based on two CAN messages with different identifiers. One message is used by the SDO client and the second message is used by the SDO server.
There are two different methods for SDO download/upload:

➔ **Expedited SDO transfer:**
- For objects long no more than 4 bytes.

➔ **Segmented SDO transfer:**
- For objects longer than 4 bytes.

Request (Client → Server)

Cob_ID	B0	B1	B2	B3	B4	B5	B6	B7
600H+NodeId	ccs/cntrl	Object Index		SubIndex	Data (Optional)			

Response (Client ← Server)

Cob_ID	B0	B1	B2	B3	B4	B5	B6	B7
580H+NodeId	scs/cntrl	Object Index		SubIndex	Data (Optional)			

Examples:

SDO - Expedited protocol download (write an object 4 bytes long):

Request (Client → Server)

Cob_ID	B0	B1	B2	B3	B4	B5	B6	B7
600H+NodeId	22H	Index		SubIndex	Value			

Response (Client ← Server)

Cob_ID	B0	B1	B2	B3	B4	B5	B6	B7
580H+NodeId	60H	Reserved						

SDO - Expedited protocol upload (read an object 4 bytes long):

Request (Client → Server)

Cob_ID	B0	B1	B2	B3	B4	B5	B6	B7
600H+NodeId	40H	Index		SubIndex	Reserved			

Response (Client ← Server)

Cob_ID	B0	B1	B2	B3	B4	B5	B6	B7
580H+NodeId	43H	Index		SubIndex	Value			

2.5 CANopen PDO (Process Data Object)

Process Data Objects (PDO) are used to transmit any process data for the process control.
The PDOs are transmitted in broadcast and without any confirmation back to the transmitting device.

There are two types of PDOs :

- ➔ **Receive PDO** (RPDO, from master to slave)
- ➔ **Transmit PDO** (TPDO, from slave to master)

All PDOs can be asynchronous or synchronous (See objects 1400.2h,1401.2h,1800.2h,1801.2h) :
The Receive PDOs are handled as soon as possible after their receipt if set as asynchronous.
For Transmit PDOs has been implemented subindex 5 (event timer) of 18xxH objects as described in the standard CiA DS 301 V4.01 that permits to specify also a transmission frequency for asynchronous PDOs.

The PDO Mapping is static for Slim Line Drives and dynamic for Titanio Line Drives (firmware V00r74 or superior).

2.5.1 Receive PDO #1 (RPDO #1)

For the Receive PDO #1 is possible to choose (by means of 2200.5H object) one of the following static mapping (only SW1 Drives with firmware V02r18 or superior, and only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r70 or superior):

Mapping #0 (object 2200.5 , bit15÷bit12 = 0)

Cob_ID	B0	B1
200H+NodeId	6040.0H	
	Control Word	

Mapping #1 (object 2200.5 , bit15÷bit12 = 1)

Cob_ID	B0	B1	B2	B3	B4	B5	B6	B7
200H+NodeId	6040.0H		6060.0H	6200.1H	607A.0H			
	Control Word		Modes of Operation	B0_Digital Outputs	Target_Position (pp)			

Mapping #2 (object 2200.5 , bit15÷bit12 = 2)

Cob_ID	B0	B1	B2
200H+NodeId	6040.0H		6060.0H
	Control Word		Modes of Operation

2.5.2 Receive PDO #2 (RPDO #2)

For the Receive PDO #2 is possible to choose (by means of 2200.5H object) one of the following static mapping:

Mapping #0 (object 2200.5 , bit7÷bit4 = 0)

Cob_ID	B0	B1	B2
300H+NodeId	6040.0H		6060.0H
	Control Word		Modes of Operation

Mapping #1 (object 2200.5 , bit7÷bit4 = 1)

Cob_ID	B0	B1	B2	B3	B4	B5
300H+NodeId	6040.0H		607A.0H			
	Control Word		Target position (pp)			

Mapping #3 (object 2200.5 , bit7÷bit4 = 3)

Cob_ID	B0	B1	B2	B3
300H+NodeId	6040.0H		6042.0H	
	Control Word		vl_Target Velocity (vl)	

Mapping #4 (object 2200.5 , bit7÷bit4 = 4)

Cob_ID	B0	B1	B2	B3	B4	B5
300H+NodeId	6040.0H		60C1.1H			
	Control Word		Interpolation Data x1 (ip)			

Mapping #5 (object 2200.5 , bit7÷bit4 = 5)

Cob_ID	B0	B1	B2	B3	B4	B5
300H+NodeId	6040.0H		60FF.0H			
	Control Word		Target Velocity (pv)			

Mapping #6 (object 2200.5 , bit7÷bit4 = 6)

Cob_ID	B0	B1	B2	B3	B4	B5	B6
300H+NodeId	6040.0H		60FF.0H				6200.1
	Control Word		Target Velocity (pv)				B0_Digital Outputs

Mapping #9 (object 2200.5 , bit7÷bit4 = 9)

Cob_ID	B0	B1	B2	B3	B4	B5	B6	B7
300H+NodeId	60C1.1H				60FF.0H			
	Interpolation Data x1 (ip)				Target Velocity (pv)			

Mapping #10 (object 2200.5 , bit7÷bit4 = 10)

Cob_ID	B0	B1	B2	B3	B4	B5
300H+NodeId	607A.0H				6042.0H	
	Target position (pp)				vl_Target Velocity (vl)	

Note:

1) The Mapping # is available only for the CANopen version of the Drive. For EtherCAT refers to §5.2.
2) The Mapping #3 is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior

2.5.3 Receive PDO #3 (RPDO #3)

For the Receive PDO #3 is possible to choose (by means of 2200.1AH object) one of the following static mapping (only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r67 or superior):

Mapping #0 (object 2200.1A , bit15÷bit12 = 0)

Cob_ID	B0	B1	B2	B3	B4	B5	B6	B7
400H+NodeId	607A.0H				6081.0H			
	Target position (pp)				Profile Velocity (pp)			

Mapping #1 (object 2200.1A , bit15÷bit12 = 1)

Cob_ID	B0	B1	B2	B3
400H+NodeId	60C1.1H			
	Interpolation Data x1 (ip)			

2.5.4 Receive PDO #4 (RPDO #4)

Only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r67 or superior :

Mapping #0 (object 2200.1A , bit7÷bit4 = 0)

Cob_ID	B0	B1	B2	B3
500H+NodeId	60FF.0H			
	Target Velocity (pv)			

2.5.5 Transmit PDO #1 (TPDO #1)

For the Transmit PDO #1 is possible to choose (by means of 2200.5H object) one of the following static mapping (only SW1 Drives with firmware V02r18 or superior, and only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r70 or superior):

Mapping #0 (object 2200.5 , bit11÷bit8 = 0)

Cob_ID	B0	B1
180H+NodeId	6041.0H	
	Status Word	

Mapping #1 (object 2200.5 , bit11÷bit8 = 1)

Cob_ID	B0	B1	B2	B3	B4	B5	B6	B7
180H+NodeId	6041.0H		6061.0H	6000.1H	6064.0H			
	Status Word		Modes of Op Display	B0_Digital Inputs	Position_Actual_Value			

Mapping #1 (object 2200.5 , bit11÷bit8 = 2)

Cob_ID	B0	B1	B2	B3	B4	B5	B6	B7
180H+NodeId	6041.0H		6061.0H	6000.1H	60FD.0H			
	Status Word		Modes of Op Display	B0_Digital Inputs	Digital_Inputs			

2.5.6 Transmit PDO #2 (TPDO #2)

For the Receive PDO #2 is possible to choose (by means of 2200.5H object) one of the following static mapping :

Mapping #0 (object 2200.5 , bit3÷bit0 = 0)

Cob_ID	B0	B1	B2
280H+NodeId	6041.0H		6061.0H
	Status Word		Modes of Op Display

Mapping #1 (object 2200.5 , bit3÷bit0 = 1)

Cob_ID	B0	B1	B2	B3	B4	B5
280H+NodeId	6041.0H		6064.0H			
	Status Word		Position Actual Value			

Mapping #2 (object 2200.5 , bit3÷bit0 = 2)

Cob_ID	B0	B1	B2	B3	B4	B5
280H+NodeId	6041.0H		606C.0H			
	Status Word		Velocity Actual Value			

Mapping #3 (object 2200.5 , bit3÷bit0 = 3)

Cob_ID	B0	B1	B2	B3
280H+NodeId	6041.0H		6044.0H	
	Status Word		vl Control Effort	

Mapping #4 (object 2200.5 , bit3÷bit0 = 4)

Cob_ID	B0	B1	B2	B3	B4	B5
280H+NodeId	6041.0H		2007.0H			
	Status Word		Encoder Actual Value[0]			

Mapping #5 (object 2200.5 , bit3÷bit0 = 5)

Cob_ID	B0	B1	B2	B3	B4	B5
280H+NodeId	6041.0H		2008.0H			
	Status Word		Encoder_Actual_Value[1]			

Mapping #6 (object 2200.5 , bit3÷bit0 = 6)

Cob_ID	B0	B1	B2	B3	B4	B5
280H+NodeId	6041.0H		60F4.0H			
	Status Word		Following_error_actual_value			

Mapping #7 (object 2200.5 , bit3÷bit0 = 7)

Cob_ID	B0	B1	B2	B3
280H+NodeId	6064.0H			
	Position Actual Value			

Mapping #8 (object 2200.5 , bit3÷bit0 = 8)

Cob_ID	B0	B1	B2	B3	B4	B5	B6
280H+NodeId	2007.0H				6000.1H	6404.1H	
	Encoder_Actual_Value[0]				B0_Digital_Inputs	Analog_In[0]	

Mapping #9 (object 2200.5 , bit3÷bit0 = 9)

Cob_ID	B0	B1	B2	B3	B4	B5	B6	B7
280H+NodeId	606C.0H				60F4.0H			
	Velocity_Actual_Value				Following_error_actual_value			

Mapping #10 (object 2200.5 , bit3÷bit0 = 10)

Cob_ID	B0	B1	B2	B3	B5	B6
280H+NodeId	6064.0H				6404.1H	
	Position Actual Value				Analog_In[0]	

Note:

1) The Mapping # is available only for the CANopen version of the Drive. For EtherCAT refers to §5.2.
2) The Mapping #3 is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior
3) The Mapping #10 is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r70 or superior

2.5.7 Transmit PDO #3 (TPDO #3)

Only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r67 or superior :

Mapping #0 (object 2200.1A , bit11÷bit8 = 0)

Cob_ID	B0	B1	B2	B3	B4	B5	B6	B7
380H+NodeId	606C.0H				6064.0H			
	Velocity_Actual_Value				Position Actual Value			

2.5.8 Transmit PDO #4 (TPDO #4)

Only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r67 or superior :

Mapping #0 (object 2200.1A , bit3÷bit0 = 0)

Cob_ID	B0	B1
480H+NodeId	603F.0H	
	Error Code	

2.6 CANopen SYNC (Synchronization Message)

The Cob-Id is fixed to 80h (CiA DS301 default value) and cannot be change by means of object 1005.0H (Cob-Id SYNC). The SYNC message is useful when it is necessary to retrieve PDOs from the drive only when requested by the master (SYNC producer) or to makes PDOs sent by the master processed at the same time by the drives.

Cob_ID
80H

2.7 CANopen Heartbeat

The Slim Line family drives implement the heartbeat protocol as defined in CiA DS 301 V4.01. This permits to the Master to check the drive working condition. It is possible to change the frequency of heartbeat transmission with the object 1017.0H (Producer Heartbeat Time). At switch-on the drive send the Boot-up message that is an heartbeat message with Status = 0;

Cob_ID	B0
700H+NodeId	Status

The Slim Line family drives support only Pre-Operational (127) and Operational (5) states.

2.7.1 CANopen Nodeguarding

CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r74 or superior implements also the nodeguarding protocol as defined in CiA DS 301 V4.01. Nodeguarding cannot be enabled together with Heartbeat (1017.0H). To enable the nodeguarding, the 1017.0 must be set to 0 and 100C and 100D must be set different than 0. The bit #7 of the status act a toggle and change at each remote request by the master.

Cob_ID	B0
700H+NodeId	Status

2.8 CANopen Emergency Telegram (EMCY)

The Slim Line drives send an Emergency Telegram every time a fault (software or hardware) is detected.
The Slim Line drives send also an Emergency Telegram at switch on without any data bytes (only Cob_Id).

Cob_ID	B0	B1	B2	B3	B4	B5	B6	B7
80H+NodeId	Error Code		Error Register	Manufacturer Specific error register				

Error Code : standard CiA error code (object 603Fh.0h)
Error Register : standard CiA error register (object 1001h.0h)
Manufacturer Specific Error Register : (object 1002h.0h)

Note: The EtherCAT version of drives supports either EMCY or Diagnostics.

2.8.1 Emergency Codes

Error Code	Error Register	Manufacturer Specific Error Register	Fault Type	Description
FF01h	81h	01h	Non-Fatal	Division by zero
FF08h	81h	08h	Non-Fatal	Value Out of range
FF15h	81h	15h	Non-Fatal	Thermal protection
FF16h	81h	16h	Non-Fatal	Voltage Protection
FF17h	81h	17h	Fatal	Current Protection
FF1Ah	81h	1Ah	Fatal	Watch dog
FF28h	81h	28h	Fatal	Missing Calibration
FF33h	81h	33h	Fatal	Open Phase : verify motor connections
FF4Ah	81h	4Ah	Fatal	Internal Software
FF59h	81h	59h	Fatal	EEprom Fail
FF5Bh	81h	5Bh	Non-Fatal	Motor current regulation out of range
FF5Fh	81h	5Fh	Non-Fatal	Feedback Error (closed loop)
FF62h	81h	62h	Fatal	EEprom write overrun
FF63h	81h	63h	-	Reserved
FF64h	81h	64h	Non-Fatal	Sync not received (ip/csp mode)
FF65h	81h	65h	Non-Fatal	Forward Limit switch reached
FF66h	81h	66h	Non-Fatal	Backward Limit switch reached
8130h	11h	67h	Non-Fatal	Guarding Fail
FF68	81h	68h	Non-Fatal	Missing Torque Enable
FF69	81h	69h	Non-Fatal	I2T Protection

Note :

- If Fault Type is 'Fatal' is necessary to reset the drive by mean of 'Reset Node command' or switch off the drive to try to reset the fault condition.
- If Fault is 'Watch dog' , only by mean of switch off the drive is possible to reset the fault condition.
- Contact 'Ever company' if is not possible to reset a Fatal fault
- Non Fatal fault is reset by mean of 'Fault' reset transition #15 (See state machine)
- Except when specified, the reaction of non-fatal faults is indicated in the Fault Reaction option code (605E.0 object)

2.9 CANopen Boot Up / NMT Protocols

At switch-on the Slim Line drives are in Pre-Operational state, this means that PDOs are disabled. At switch-on the drive send the Boot-up message that is a heartbeat message with Status = 0.

The boot-up process could take 6-7 seconds.

Cob_ID	B0
700H+NodeId	0

The Master have to send the NMT frame with command Start Node.

Start Node Command

Cob_ID	B0	B1
00H	1	NodeId

Enter Pre-Operational State Command

Cob_ID	B0	B1
00H	128	NodeId

If NodeId = 0 all devices connected to CAN network will execute the command.

It is supported also the NMT - Reset Node Protocol to reset the drive:

Reset Node Command

Cob_ID	B0	B1
00H	129	NodeId

3.0 CANopen profile DSP402

The purpose of the profile is to give drives an understandable and unique behavior on the CAN bus network.
The two principal advantages of the profile approach for device specification are in the areas of system integration and device standardization. A device profile defines a 'standard' device. This standard device represents really basic functionality, every device within this device class must support. This mandatory functionality is necessary to ensure that at least simple non-manufacturer-specific operation of a device is possible. The concept of device standardization is extended by the notion of *optional* functionality defined within the standardized device profile. Such optional functionality does not have to be implemented by all manufactures. However, if a manufacturer implements such optional functionality , he must do so in a fixed manner. Providing optional functionality is a very powerful mechanism to ensure all manufacturers implementing particular functionality in a defined fashion. The device profiles provide a mechanism by which manufacturers wishing to implement manufacturer specific functionality can do so as well. This is necessary since it would be impossible to anticipate all possible device functionality and define this in the optional category of each device class. By defining mandatory device features, basic network operation is guaranteed. By defining optional device features, a degree of defined flexibility can be built in. By leaving hooks for manufacturer specific functionality, manufacturers will not be constrained to an out-of-date standard.

3.1 Device Control

The device control function block controls all functions of the drive. It is divided into :

- **State machine**
- **Operation mode**

The **State machine** describes the device status and the possible control sequence of the drive. A single state represents a special internal or external behavior. The state of the drive also determines which commands are accepted. States may be changed by the controlword (object 6040.0h) and/or according to internal events. The current state is shown in the statusword (object 6041.0h).

The **Operation mode** defines the behavior of the drive. The drive functions depend from the select mode of operation (object 6060.0h). The specific drive function is executed only when the drive status is **Operation Enabled**. The following modes of operation are defined in the 'Slim Line Series Drives' :

- **Interpolated Position Mode (ip)**
- **Profile Position Mode (pp)**
- **Homing Mode (hm)**
- **Profile Velocity Mode (pv)**
- **Cyclic Synchronous Position Mode (csp)**
- **Cyclic Synchronous Velocity mode (csv)**
- **Velocity Mode (vl)**

3.1.1 State machine

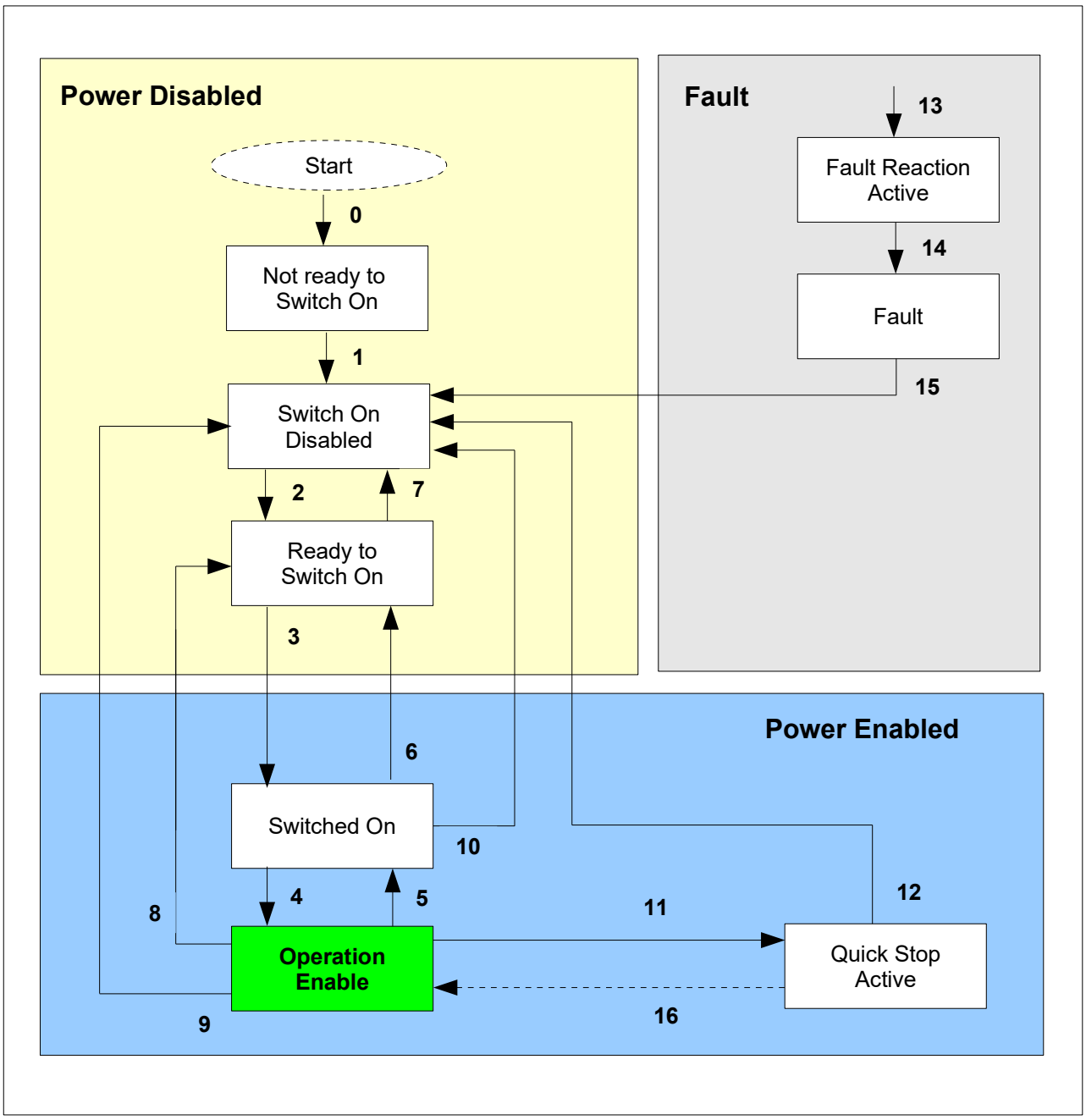


Figure 1 : Device Control State Machine

3.1.2 Drive states

State	StatusWord															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Not Ready to Switch On	X	X	X	X	X	X	X	X	X	0	X	X	0	0	0	0
Switch On Disabled	X	X	X	X	X	X	X	X	X	1	X	X	0	0	0	0
Ready to Switch On	X	X	X	X	X	X	X	X	X	0	1	X	0	0	0	1
Switched On	X	X	X	X	X	X	X	X	X	0	1	X	0	0	1	1
Operation Enable	X	X	X	X	X	X	X	X	X	0	1	X	0	1	1	1
Quick Stop Active	X	X	X	X	X	X	X	X	X	0	0	X	0	1	1	1
Fault Reaction Active	X	X	X	X	X	X	X	X	X	0	X	X	1	1	1	1
Fault	X	X	X	X	X	X	X	X	X	0	X	X	1	0	0	0

(Table 1 : Device state bits)

● Not Ready to Switch On :

– The drive is being initialized
– The drive function is disabled (no energy is supplied to the motor)
– The drive is not ready to accept commands

● Switch On Disabled :

– Drive initialization is complete
– The drive parameters have been set up
– Drive parameters may be changed
– The drive function is disabled (no energy is supplied to the motor)

● Ready To Switch On :

– The drive parameters may be changed
– The drive function is disabled (no energy is supplied to the motor)

● Switched On :

– The power amplifier is ready
– The drive parameters may be changed
– The drive function is disabled (no energy is supplied to the motor)

● Operation Enable :

– No faults have been detected
– The drive function is enabled and power is applied to the motor
– The drive parameters may be changed

● Quick Stop Active :

– The drive parameters may be changed
– The quick stop function is being executed
– The drive function is enabled and power is applied to the motor

● Fault Reaction Active(*) :

– The drive parameters may be changed
– A fault has occurred in the drive
– The drive function is enabled and power is applied to the motor

● Fault (*) :

– The drive parameters may be changed
– A fault has occurred in the drive
– The drive function is disabled (no energy is supplied to the motor)

(*) Drive faults may be fatal or non-fatal. When a fatal fault occurs, the drive is no longer able to control the motor, so a switch off of the drive is necessary.

3.1.3 State Transitions

State transitions are caused by internal events in the drive or by commands from the host with the 'Controlword'.

Transition	EVENT		ACTION
	Command from the host	*Internal*	
0		Reset.	The drive self-initializes.
1		The drive has initialized successfully.	Activate communication.
2	'Shutdown'		None.
3	'Switch On'		The power section is switched on.
4	'Enable Operation'		The drive function is enabled (power is applied to the motor).
5	'Disable Operation'		The drive operation will be disabled (no energy will be supplied to the motor). See object 605Ch.
6	'Shutdown'		The drive function is disabled (power is not applied to the motor).
7	'Quick Stop' or 'Disable Voltage'		None.
8	'Shutdown'		The power section is switched off immediately. Power is not applied to the motor. The motor is free to rotate if unbraked. See object 605Bh.
9	'Disable Voltage'		The power section is switched off immediately. Power is not applied to the motor. The motor is free to rotate if unbraked.
10	'Disable Voltage' or 'Quick Stop'		The power section is switched off immediately. Power is not applied to the motor. The motor is free to rotate if unbraked.
11	'Quick Stop'		The quick stop function is executed.
12	'Disable Voltage'	'Quick Stop' action is completed .	The power section is switched off immediately. Power is not applied to the motor. The motor is free to rotate if unbraked.
13		A fault has occurred in the drive.	The drive executes appropriate fault reaction (see 605E.0 h object).
14		The fault reaction is completed.	The drive function is disabled (power is not applied to the motor).
15	'Fault Reset'		The fault condition is reset if no fault currently exists in the drive. After leaving Fault state, the Fault Reset bit of the Controlword must be cleared by the host.
16	'Enable Operation'		The drive function is enabled (power is applied to the motor).

(Table 2 : State transitions)

Command	Transition	ControlWord															
		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Shutdown	[2][6][8]	X	X	X	X	X	X	X	X	X	X	X	X	X	**1**	**1**	**0**
Switch On	[3]	X	X	X	X	X	X	X	X	X	X	X	X	X	**1**	**1**	**1**
Disable Voltage	[7][9][10][12]	X	X	X	X	X	X	X	X	X	X	X	X	X	X	**0**	X
Quick Stop	[7][10][11]	X	X	X	X	X	X	X	X	X	X	X	X	X	**0**	**1**	X
Disable Operation	[5]	X	X	X	X	X	X	X	X	X	X	X	X	**0**	**1**	**1**	**1**
Enable Operation	[4][16]	X	X	X	X	X	X	X	X	X	X	X	X	**1**	**1**	**1**	**1**
Fault Reset	[15]	X	X	X	X	X	X	X	X	↑	X	X	X	X	X	X	X

(Table 3 : Commands in the controlword)

Note :

- bits marked X are irrelevant

3.2 Position, Velocity, Acceleration Units

The type of physical units for Position, Velocity and Acceleration depend from EVER Drive type, software version installed and if the 'Factor Group' objects are supported or not :

- The 'SW1,SM1A,DMD105 Drives' do not support the 'Factor Group' objects.
- The 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79' or lower do not support the 'Factor Group' objects.
- The 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior' support the 'Factor Group' objects.

The 'Factor Group' objects define a relationship between 'user-defined units' and 'device internal units'.
If the 'Factor Group' objects are not supported then it is necessary refer directly to 'device internal units'.

The 'device internal units' are :

'Position units'	→	Increments	[Inc]
'Velocity units'	→	Increments/sec	[Inc/s]
'Acceleration units'	→	Increments/sec^2	[Inc/s^2]

Only for 'Velocity Mode' , the 'Velocity internal unit' is rpm (revolutions/min) and for conversion are used the 'vl dimension factor ' objects.

Without 'Factor Group' and (606A.0h = -1)

The Position, Velocity and Acceleration resolution depend from Motor resolution defined by mean of 'Motor_Step_Angle' object (2012.1h) and 'Motor_Poles' object (2012.2h).

In this configuration no incremental encoder for feedback purpose is used and then the Position and Velocity objects are all referred only to theoretical values.

Without 'Factor Group' and (606A.0h = 0 or 606A.0h = -2)

The Position, Velocity and Acceleration resolution depend from Motor resolution defined by mean of 'Motor_Step_Angle' object (2012.1h) and 'Motor_Poles' object (2012.2h).

In this configuration is used an incremental encoder for feedback purpose.
For this reason it is also necessary to define the encoder resolution by mean of 'Position_Encoder_Resolution' objects (608F.1h and 608F.2h).

The objects :

- (6044.0h) 'vl_Velocity_actual_value'
- (6064.0h) 'Position_actual_value'
- (606C.0h) 'Velocity_actual_value'
- (60F4.0h) 'Following_error_actual_value'

return the real actual value of the incremental encoder normalized to Motor Resolution and specific unit.

With 'Factor Group' and (606A.0h = -1)

The 'Motor_Step_Angle' object (2012.1h), 'Motor_Poles' object (2012.2h) and 'Velocity Factors' objects (2013.1h , 2013.2h) are used only for Velocity conversion.

The 'Motor_Step_Angle' object (2012.1h), 'Motor_Poles' object (2012.2h) and 'Acceleration Factors' objects (2013.3h , 2013.4h) are used only for Acceleration conversion.

For Position conversion 'Feed constant' and 'Gear Ratio' objects of Factor Group are used with an internal motor resolution of 65536 increments/rev.

In this configuration no incremental encoder for feedback purpose is used and then the Position and Velocity objects are all referred only to theoretical values.

The 'User-defined units' are supported and converted in 'Device Internal unit' using 'Factor Group' objects.

With 'Factor Group' and (606A.0h = 0 or 606A.0h = -2)

The 'Motor_Step_Angle' object (2012.1h), 'Motor_Poles' object (2012.2h) and 'Velocity Factors' objects (2013.1h , 2013.2h) are used only for Velocity conversion.

The 'Motor_Step_Angle' object (2012.1h), 'Motor_Poles' object (2012.2h) and 'Acceleration Factors' objects (2013.3h , 2013.4h) are used only for Acceleration conversion.

For Position conversion 'Feed constant' and 'Gear Ratio' objects of Factor Group are used with an internal motor resolution of 65536 increments/rev.

In this configuration is used an incremental encoder for feedback purpose.
For this reason it is also necessary to define the encoder resolution by mean of 'Position_Encoder_Resolution' objects (608F.1h and 608F.2h).

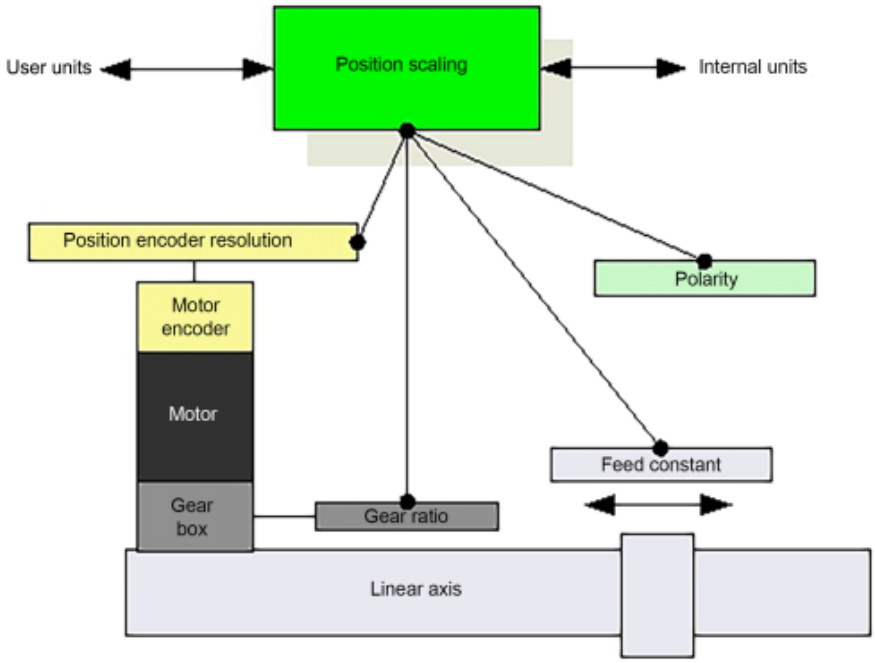
The objects :

- (6044.0h) 'vl_Velocity_actual_value'
- (6064.0h) 'Position_actual_value'
- (606C.0h) 'Velocity_actual_Value'
- (60F4.0h) 'Following_error_actual_value'

return the real value of the incremental encoder normalized to 'User-defined unit'.

3.2.1 Factor group

The factors defined in the factor group set up a relationship between ‘device internal units’ and ‘user-defined units’. The ‘user-defined units’ as defined in this document are used in the corresponding objects representing position, velocity and acceleration values.



The ‘Factor Group’ is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00R80 or superior.

The calculation of the Position, Velocity and Acceleration values is done by the following formulas :

$$Position\ value\left(user\ defined\ unit\right)=\frac{Position\ value\left(device\ internal\ unit^{(1)}\right)*Feed\ constant}{65536\ inc/rev*Gear\ Ratio}$$

$$Velocity\ value\left(user\ defined\ unit\right)=\frac{Velocity\ value\left(device\ internal\ unit^{(2)}\right)}{Velocity\ Factor}$$

$$Acceleration\ value\left(user\ defined\ unit\right)=\frac{Acceleration\ value\left(device\ internal\ unit^{(3)}\right)}{Acceleration\ Factor}$$

$$Feed\ constant=\frac{6092.1\,h}{6092.2\,h}$$

$$Gear\ Ratio=\frac{6091.1\,h}{6091.2\,h}$$

$$Velocity\ Factor=\frac{2013.1\,h}{2013.2\,h}$$

$$Acceleration\ Factor=\frac{2013.3\,h}{2013.4\,h}$$

Note :

(1) Increments (related to 65536 Inc/rev)
(2) Increments/sec (related to 2012.1h and 2012.2h objects)
(3) Increments/sec^2 (related to 2012.1h and 2012.2h objects)

Factor Group related objects :

Index	*Object*	*Name*	*Type*	*Attr.*	*M/O*
2012	RECORD	Motor Parameters		rw	M
2013	RECORD	Motor Factor (1)	UNSIGNED32	rw	O
608F h	ARRAY	Position Encoder Resolution	UNSIGNED32	rw	O
6091 h	ARRAY	Gear Ratio (1)	UNSIGNED32	rw	O
6092 h	ARRAY	Feed Constant (1)	UNSIGNED32	rw	O
60EF.0 h	VAR	Motor Resolution (1)	UNSIGNED32	ro	O

Notes :

(1) available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00R80 or superior

3.2.2 Unit Settings Examples

Example (1)

User Input Information :

User-defined Position Unit	= Inc
User-defined Velocity Unit	= Inc/s
User-defined Acceleration Unit	= Inc/s²
One motor revolution (related to 'Position Unit')	= 400 Inc
EVER drive with 'Factor Group'	= NO
Incremental Encoder (on the motor shaft)	= NO

Objects Setting :

(2012.1h) Motor_Step_Angle	= 2 (400 inc/rev)
(2012.2h) Motor_Poles	= 50
(606A.0h) Sensor_Selection_Code	= -1

Example (2)

User Input Information :

User-defined Position Unit	= Inc
User-defined Velocity Unit	= Inc/s
User-defined Acceleration Unit	= Inc/s²
One motor revolution (related to 'Position Unit')	= 800 Inc
EVER drive with 'Factor Group'	= NO
Incremental Encoder (on the motor shaft)	= YES
Encoder Resolution	= 4000 Inc/rev

Objects Setting :

(2012.1h) Motor_Step_Angle	= 4 (800 inc/rev)
(2012.2h) Motor_Poles	= 50
(606A.0h) Sensor_Selection_Code	= 0 (ENC#0)
(608F.1h) Position_Encoder_Resolution.Enc_Incs	= 4000
(608F.2h) Position_Encoder_Resolution.Mot_revs	= 1

Example (3)

User Input Information :

User-defined Position Unit	= Inc
User-defined Velocity Unit	= Inc/s
User-defined Acceleration Unit	= Inc/s²
One motor revolution (related to 'Position Unit')	= 400 Inc
EVER drive with 'Factor Group'	= YES
Incremental Encoder (on the motor shaft)	= NO
Gear ratio	= 1

Objects Setting :

(2012.1h) Motor_Step_Angle	= 2 (400 inc/rev)
(2012.2h) Motor_Poles	= 50
(606A.0h) Sensor_Selection_Code	= -1
(6092.1h) Feed_Constant.Feed	= 400
(6092.2h) Feed_Constant.Shaft_Revs	= 1
(6091.1h) Gear_Ratio.Motor_Shaft_Revs	= 1
(6091.2h) Gear_Ratio.Driving_Shaft_Revs	= 1
(2013.1h) Velocity_Factor.Numerator	= 1
(2013.2h) Velocity_Factor.Denominator	= 1
(2013.3h) Acceleration_Factor.Numerator	= 1
(2013.4h) Acceleration_Factor.Denominator	= 1

Notes :

- In this example, the objects 2012.1h and 2012.2h are used only for velocity and acceleration conversions.

Example (4)

User Input Information :

User-defined Position Unit	= Inc
User-defined Velocity Unit	= Inc/s
User-defined Acceleration Unit	= Inc/s²
One motor revolution (related to 'Position Unit')	= 800 Inc
EVER drive with 'Factor Group'	= YES
Incremental Encoder (on the motor shaft)	= YES
Encoder Resolution	= 4000 Inc/rev
Gear ratio	= 1

Objects Setting :

(2012.1h) Motor_Step_Angle	= 4 (800 inc/rev)
(2012.2h) Motor_Poles	= 50
(606A.0h) Sensor_Selection_Code	= 0 (ENC#0)
(6092.1h) Feed_Constant.Feed	= 800
(6092.2h) Feed_Constant.Shaft_Revs	= 1
(6091.1h) Gear_Ratio.Motor_Shaft_Revs	= 1
(6091.2h) Gear_Ratio.Driving_Shaft_Revs	= 1
(2013.1h) Velocity_Factor.Numerator	= 1
(2013.2h) Velocity_Factor.Denominator	= 1
(2013.3h) Acceleration_Factor.Numerator	= 1
(2013.4h) Acceleration_Factor.Denominator	= 1
(608F.1h) Position_Encoder_Resolution.Enc_Incs	= 4000
(608F.2h) Position_Encoder_Resolution.Mot_revs	= 1

Notes :

- In this example, the objects 2012.1h and 2012.2h are used only for velocity and acceleration conversions.

Example (5)

User Input Information :

User-defined Position Unit	= mm
User-defined Velocity Unit	= rpm
User-defined Acceleration Unit	= mm/s²
One motor revolution (related to 'Position Unit')	= 1500 mm
EVER drive with 'Factor Group'	= YES
Incremental Encoder (on the motor shaft)	= YES
Encoder Resolution	= 4000 Inc/rev
Gear ratio	= 1

Objects Setting :

(2012.1h) Motor_Step_Angle	= 4 (800 inc/rev)
(2012.2h) Motor_Poles	= 50
(606A.0h) Sensor_Selection_Code	= 0 (ENC#0)
(6092.1h) Feed_Constant.Feed	= 1500
(6092.2h) Feed_Constant.Shaft_Revs	= 1
(6091.1h) Gear_Ratio.Motor_Shaft_Revs	= 1
(6091.2h) Gear_Ratio.Driving_Shaft_Revs	= 1
(2013.1h) Velocity_Factor.Numerator	= 800
(2013.2h) Velocity_Factor.Denominator	= 60
(2013.3h) Acceleration_Factor.Numerator	= 800
(2013.4h) Acceleration_Factor.Denominator	= 1500
(608F.1h) Position_Encoder_Resolution.Enc_Incs	= 4000
(608F.2h) Position_Encoder_Resolution.Mot_revs	= 1

Notes :

- In this example, the objects 2012.1h and 2012.2h are used only for velocity and acceleration conversions.

3.3 Interpolated Position Mode (ip)

The interpolated Position mode is used to control multiple coordinated axles or a single axle with the need for time-interpolation of set set-point data. This modality uses the SYNC message to synchronize the interpolation points. The Slim Lines Drives support **only the synchronous linear interpolation**, therefore the data structure used for interpolation has only one field and is referred to the position set-point (60C1h object).

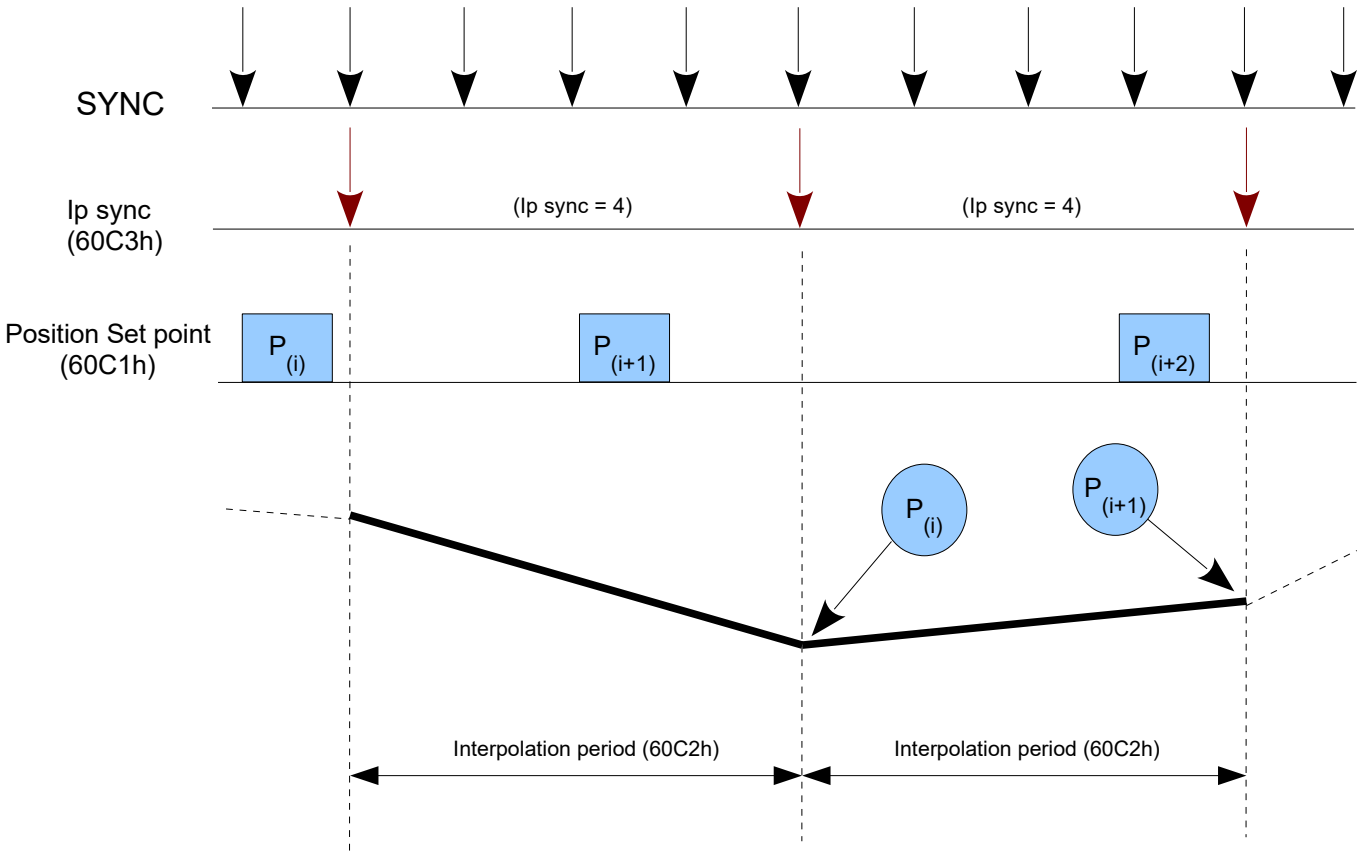
To obtain an accurate interpolation it is necessary to specify :

- The interpolation period (60C2h object)
- The ip-sync (60C3h object)

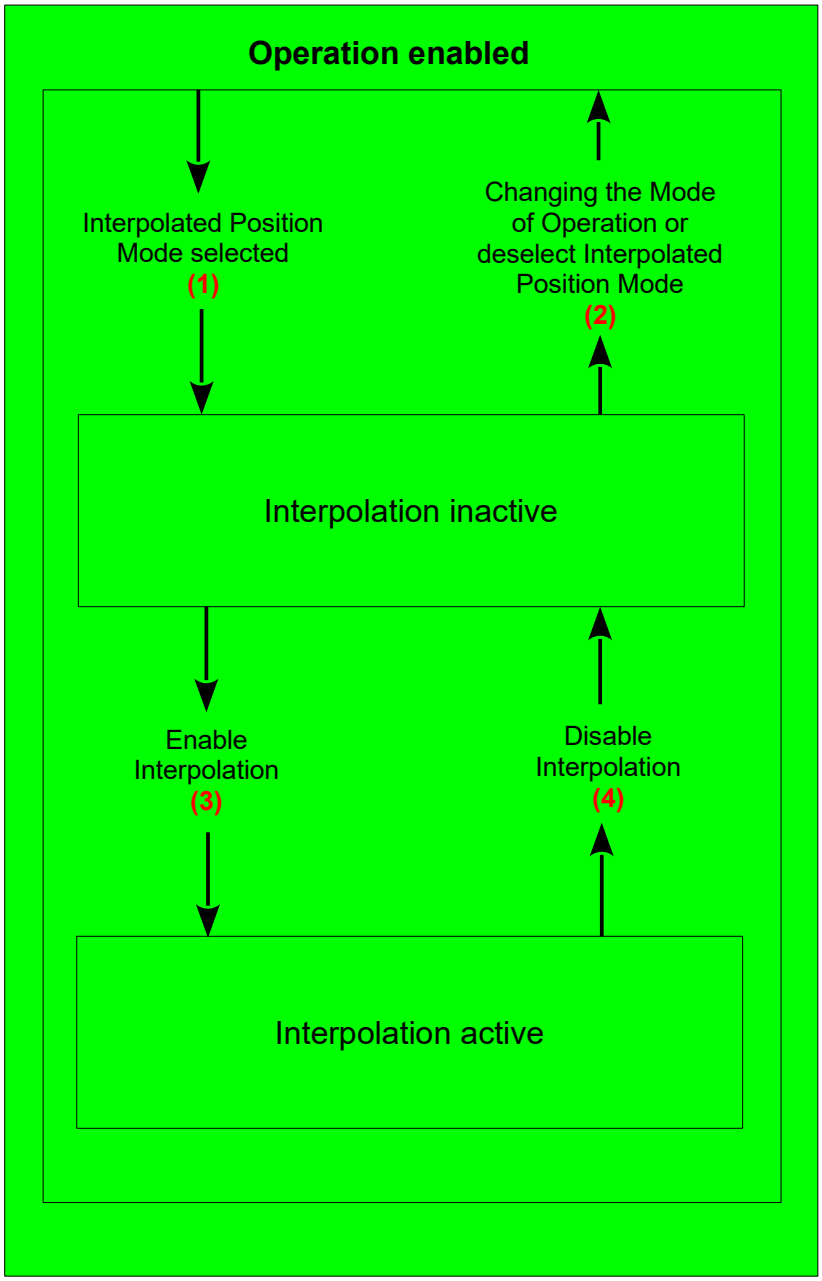
The interpolation period is directly related to the ip-sync. The ip-sync is the event that executes the position set-point. With the 60C3h object is specified how many SYNC have to be received to generate an ip-sync.

The position set point is sent by means of PDO but to execute it is necessary to receive the SYNC number specified in the 60C3h object.

To get a good real time interpolation it is necessary that the master guarantees the interpolation period and the precision of the positions sent by PDO.



3.3.1 Internal States and State transitions



Interpolation inactive

This state is entered when the device is in state OPERATION ENABLED and the interpolated position mode is selected. The drive will accept input data but it does not move the axles.

Interpolation active

This state is entered when the device is in state OPERATION ENABLED and the interpolated position mode is selected and enabled. The drive will accept input data and it moves the axles.

State Transition (1)

NO IP-MODE SELECTED => IP-MODE INACTIVE
Event : Enter in the state OPERATIONAL ENABLE with *Controlword* and select ip mode with *modes of operation*

State Transition (2)

IP-MODE INACTIVE => NO IP-MODE SELECTED
Event : Leave the state OPERATION ENABLE with *Controlword* or select any other mode with *modes of operation* if it is allowed inside the state OPERATION ENABLE

State Transition (3)

IP-MODE INACTIVE => IP-MODE ACTIVE
Event : Set bit *enable ip mode* (bit4) of the *Controlword* while in ip mode and OPERATION ENABLED

State Transition (4)

IP-MODE ACTIVE => IP-MODE INACTIVE
Event : Reset bit *enable ip mode* (bit4) of the *Controlword* while in ip mode and OPERATION ENABLED

3.3.2 Interpolated Position mode related objects

Index	*Object*	*Name*	*Type*	*Attr.*	*M/O*
6040.0 h	VAR	Controlword	UNSIGNED16	rw	M
6041.0 h	VAR	Statusword	UNSIGNED16	ro	M
60C1.0 h	ARRAY	Interpolation data record	INTEGER32	rw	O
60C2.0 h	RECORD	Interpolation time period	Interpolation time period record	rw	O
60C3.0 h	ARRAY	Interpolation sync definition	UNSIGNED8	rw	O

3.4 Profile Position Mode (pp)

A *Target Position* (object 607A.0h) is applied to the trajectory generator; it generates a *Position Demand Value* (object 6062.0h) for the position control loop.

The setting of set-points is controlled by the *New_set_point* bit and the *Change_set_immediately* bit of the *Controlword* and the *Set_point_acknowledge* bit of the *Statusword*.

If the *Change_set_immediately* bit is set to 1, a ***Single set-point*** is expected by the device.
If the *Change_set_immediately* bit is set to 0, a ***Set of set-points*** is expected by the device.

After a set-point is sent to the slave device, the master device signals that the set-point is valid by a rising edge of the *New_set_point* bit of the *Controlword*. The slave device sets to 1 the *Set_point_acknowledge* bit of the Statusword to indicate that the new set-point has been received. Afterwards the slave device sets to 0 the *Set_point_acknowledge* bit when the device is ready to accept new set-points.

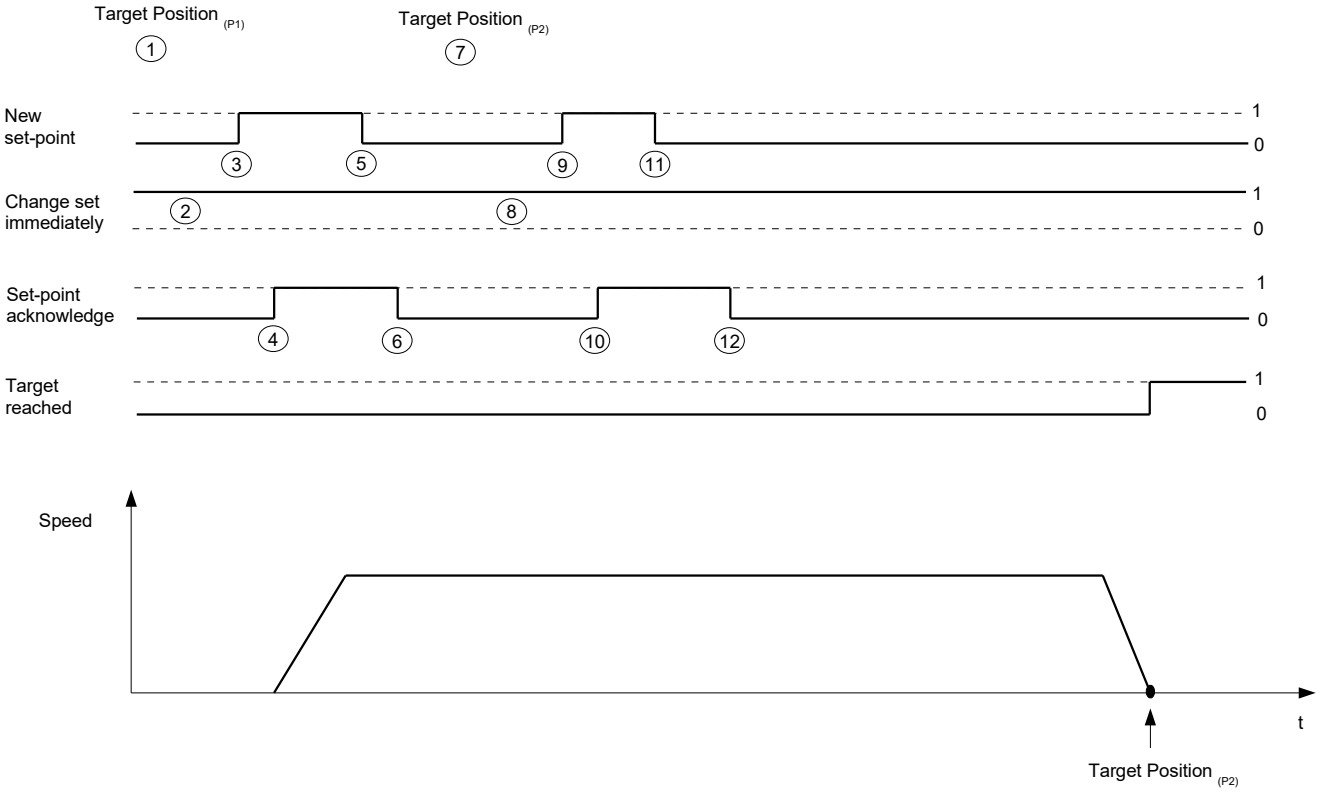
If one set-point is still in progress and new one is received from the slave device, two methods are supported :

Single set-point (*Change_set_immediately* = 1, bit #5 of Controlword)
The new set-point shall be processed immediately.

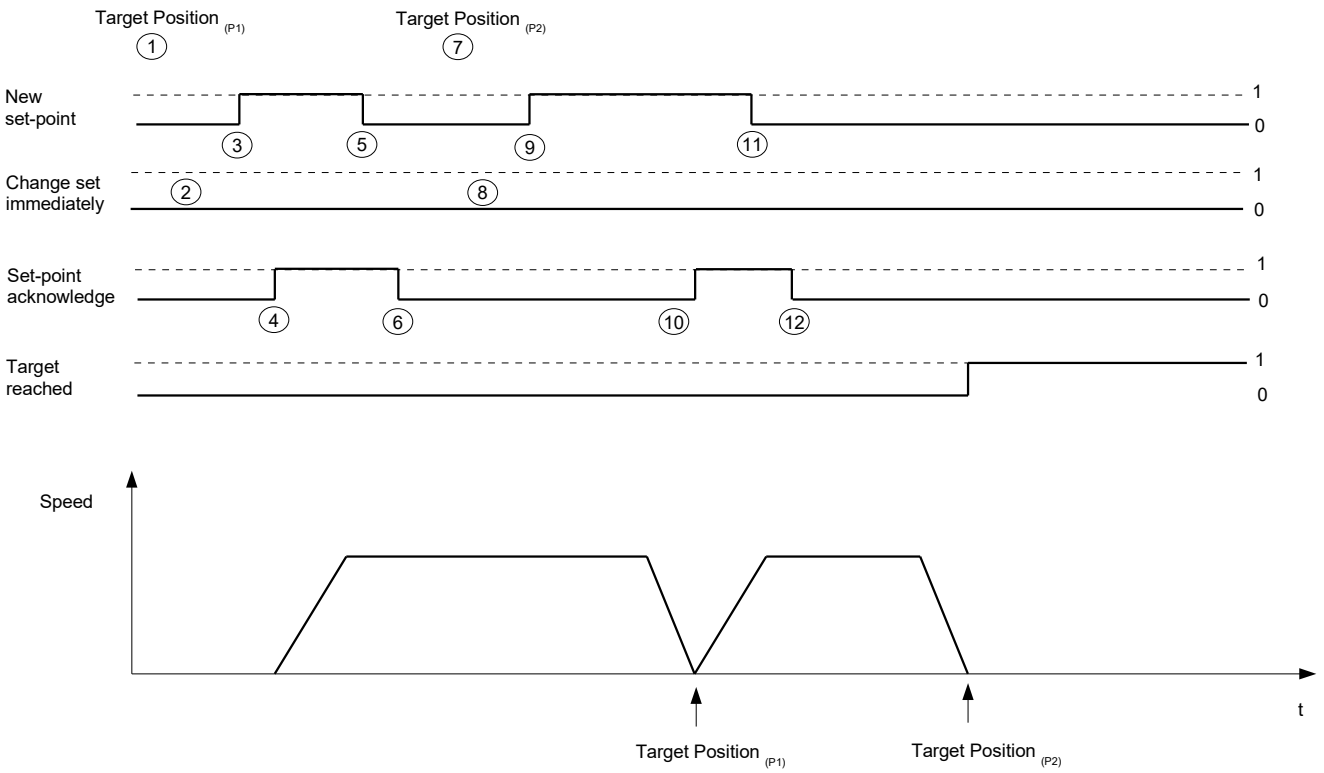
Set of set-points (*Change_set_immediately* = 0, bit #5 of Controlword)
The new set-point shall be processed only after the previous has been reached.
Up to two set-points can be set up. If all set-points available are busy (*Set_point_acknowledge* bit is 1) the reaction of the slave device depends on the *Change_set_immediately* bit. If it is set to 1, the new set-point shal be processed immediately as Single set-point.

The Target_reached bit of the Statusword shall remain to 0 until all set-points are processed.

The trajectory generator support only linear ramp (trapezoidal profile), with separate parameters for acceleration and deceleration.



(Single set-point)



(Set of set-points)

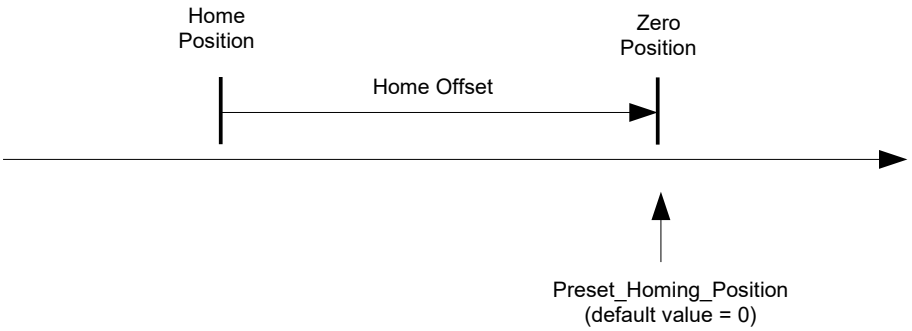
3.4.1 Profile Position mode related objects

Index	*Object*	*Name*	*Type*	*Attr.*	*M/O*
6040.0 h	VAR	Controlword	UNSIGNED16	rw	M
6041.0 h	VAR	Statusword	UNSIGNED16	ro	M
6062.0 h	VAR	Position demand value	INTEGER32	ro	O
6063.0 h	VAR	Position actual value*	INTEGER32	ro	O
6064.0 h	VAR	Position actual value	INTEGER32	ro	M
6065.0 h	VAR	Following error window	UNSIGNED32	rw	O
6066.0 h	VAR	Following error time out	UNSIGNED16	rw	O
6067.0 h	VAR	Position window	UNSIGNED32	rw	O
6068.0 h	VAR	Position window time	UNSIGNED16	rw	O
607A.0 h	VAR	Target position	INTEGER32	rw	M
6081.0 h	VAR	Profile velocity	UNSIGNED32	rw	M
6083.0 h	VAR	Profile acceleration	UNSIGNED32	rw	M
6084.0 h	VAR	Profile deceleration	UNSIGNED32	rw	O
6085.0 h	VAR	Quick stop deceleration	UNSIGNED32	rw	O
6086.0 h	VAR	Motion profile type	INTEGER16	rw	M
60F4.0 h	VAR	Following error actual value	INTEGER32	ro	O
2010.0 h	VAR	Min profile velocity	UNSIGNED16	rw	M

3.5 Homing mode (hm)

This method is used from a drive to seek the home position (also called, the datum reference point or zero point). There are various methods of achieving this, limit switches at the ends of travel or a home switch (zero point switch) in mid-travel, most of methods also use the index (zero) pulse train from an incremental encoder. The user can specify the speeds, acceleration and the method of homing.

It is possible to specify a *home offset* at the end of the seeking to displace zero in the coordinate system for the home position. The home offset is the difference between the zero position for the application and the machine home position. If the final home position must be a value different from zero, the user can define the *Preset_Homing_Position object :*



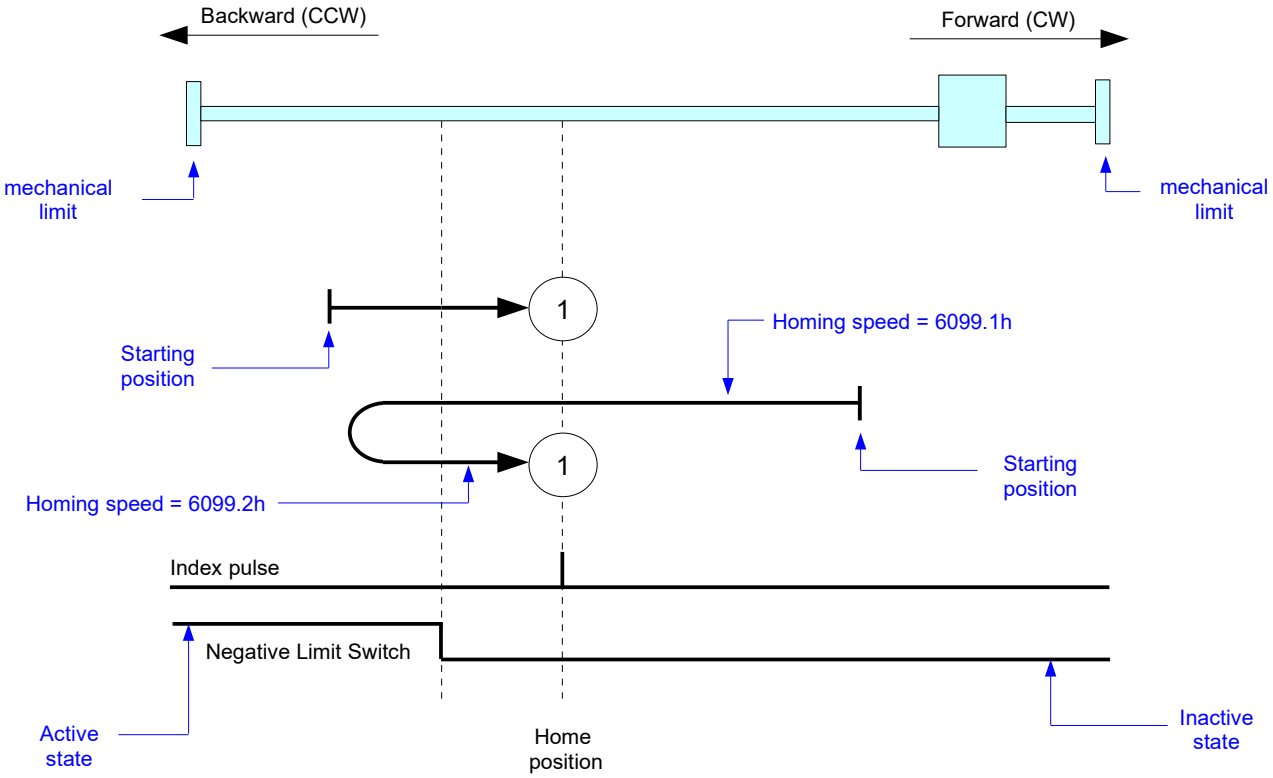
By choosing a method of homing, are defined the homing signal (positive limit switch, negative limit switch, home switch), the direction of actuation and where appropriate and the position of the index pulse. The sequence of the homing operation is described by the method.

3.5.1 Homing mode related objects

Index	*Object*	*Name*	*Type*	*Attr.*	*M/O*
6040.0 h	VAR	Controlword	UNSIGNED16	rw	M
6041.0 h	VAR	Statusword	UNSIGNED16	ro	M
607C.0 h	VAR	Home offset	INTEGER32	rw	O
6098.0 h	VAR	Homing method	INTEGER8	rw	M
6099 h	ARRAY	Homing speeds	UNSIGNED32	rw	M
609A.0 h	VAR	Homing acceleration	UNSIGNED32	rw	O
6085.0 h	VAR	Quick stop deceleration	UNSIGNED32	rw	O
2080.0 h	VAR	Preset Homing Position	INTEGER32	rw	O
2081.0 h	VAR	Drive Homing Inputs Setting	UNSIGNED32	rw	O

3.5.2 Homing method 1

The initial direction of movement is leftward if the negative limit switch is inactive or rightward if the negative limit switch is active. The home position is at the first index pulse to the right of the position where the Negative limit switch becomes inactive.



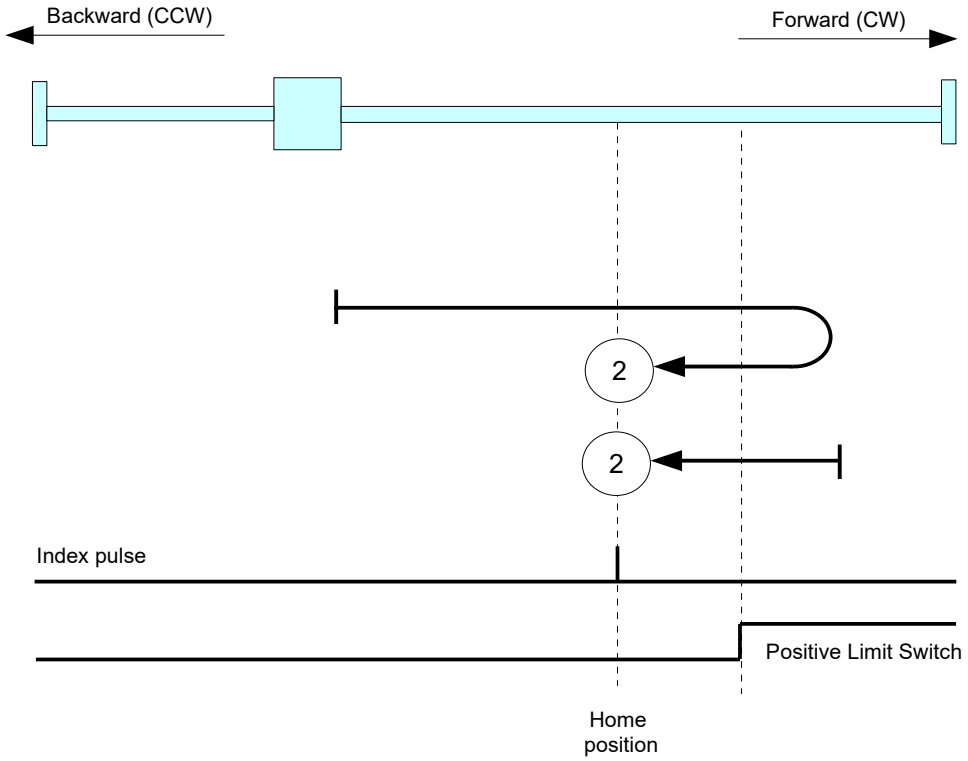
The Home Switch and Positive Limit Switch are not used.

Note:

- This Homing method is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior

3.5.3 Homing method 2

The initial direction of movement is rightward if the positive limit switch is inactive or leftward if the positive limit switch is active. The home position is at the first index pulse to the left of the position where the Positive limit switch becomes inactive.



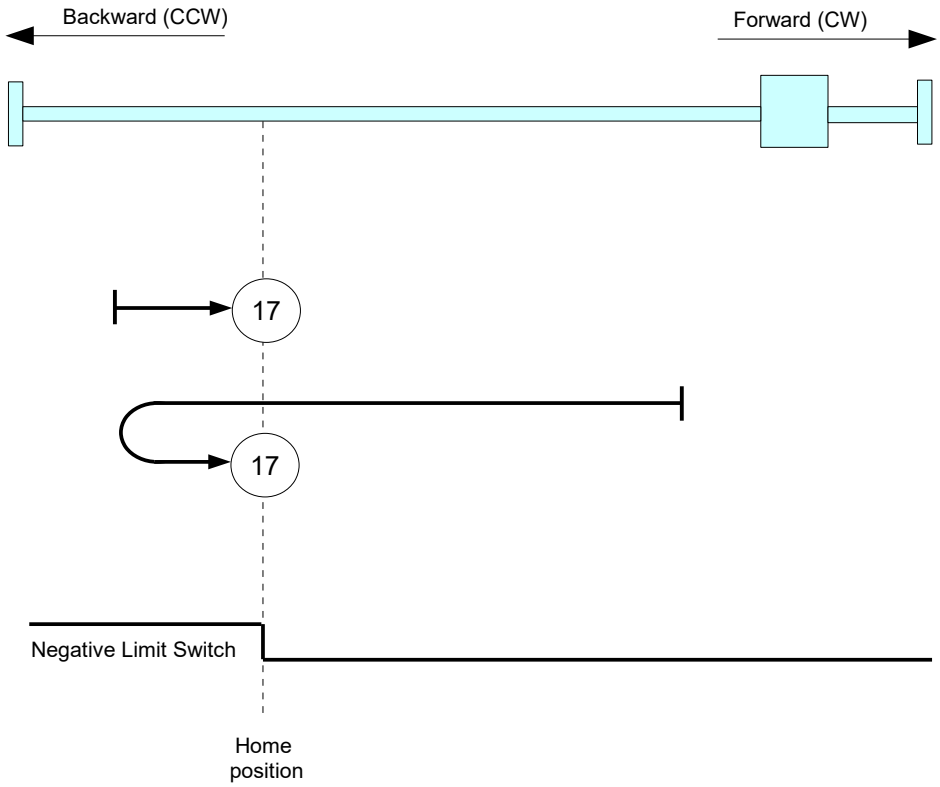
The Home switch and Negative Limit Switch are not used.

Note:

- This Homing method is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior

3.5.4 Homing method 17

The initial direction of movement is leftward if the negative limit switch is inactive or rightward if the negative limit switch is active. The home position is on the transition from active to inactive state of Negative limit switch.



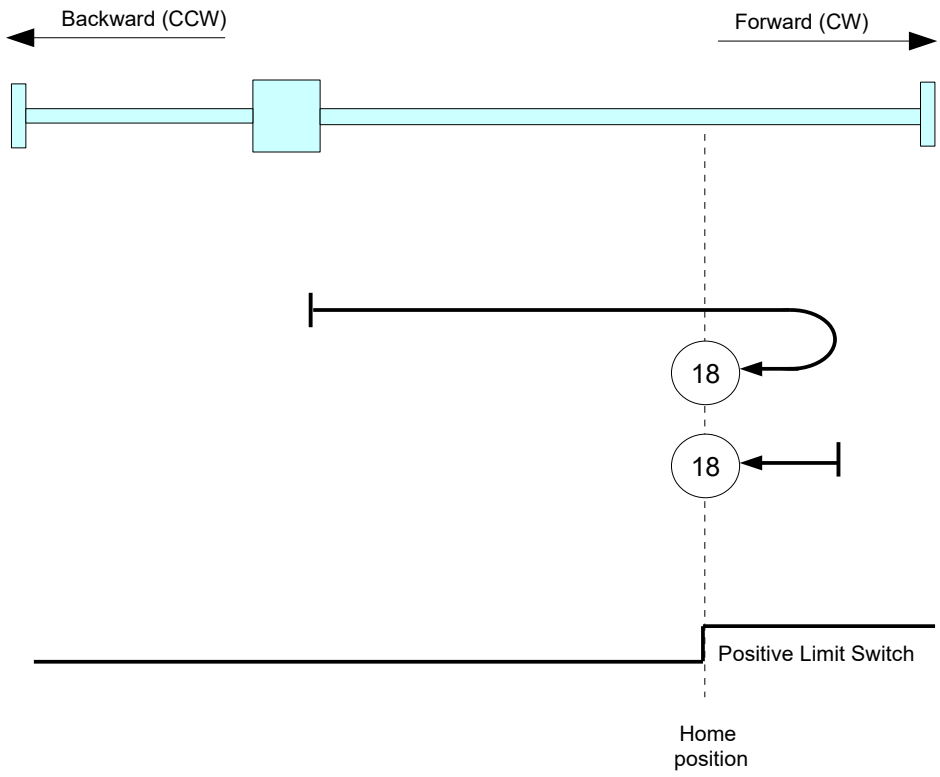
The Index pulse, Home Switch and Positive Limit switch are not used.

Note:

- This Homing method is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior

3.5.5 Homing method 18

The initial direction of movement is rightward if the positive limit switch is inactive or leftward if the positive limit switch is active. The home position is on the transition from active to inactive state of Positive limit switch.



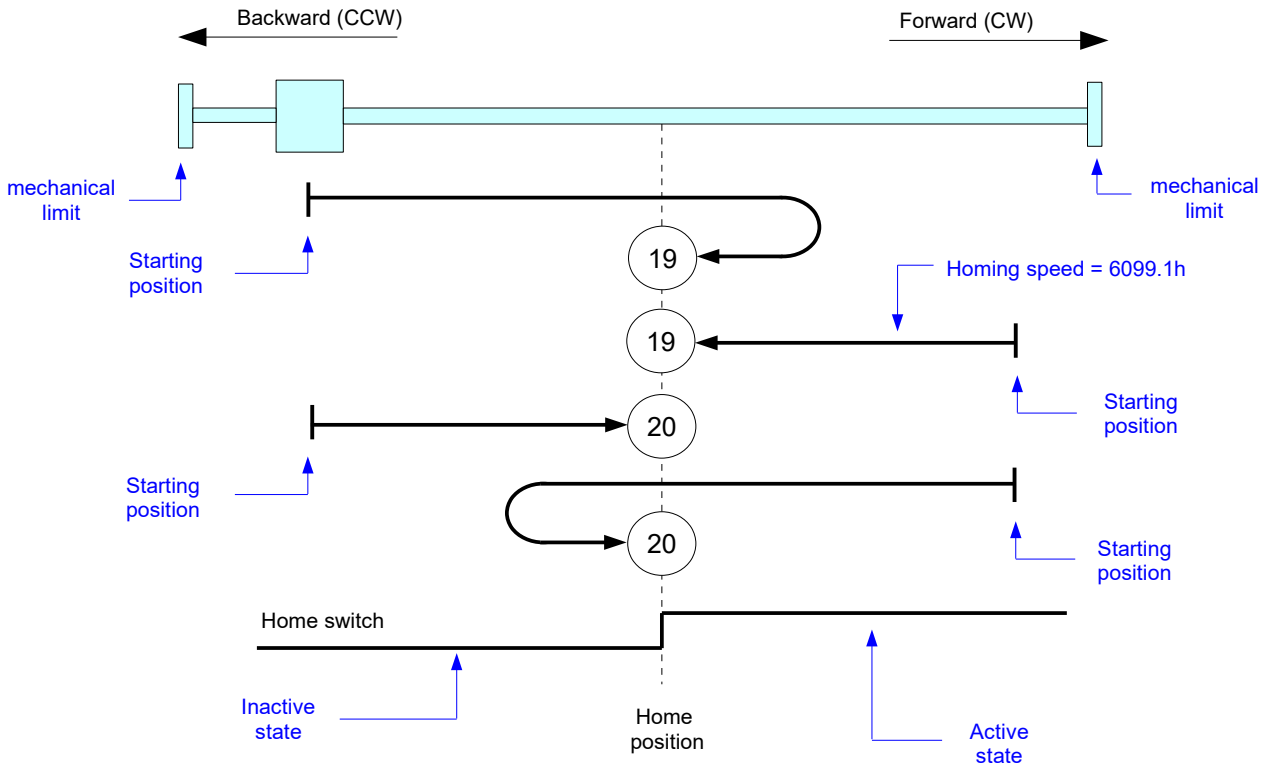
The Index pulse, Home switch and Negative Limit Switch are not used.

Note:

- This Homing method is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior

3.5.6 Homing methods 19 and 20

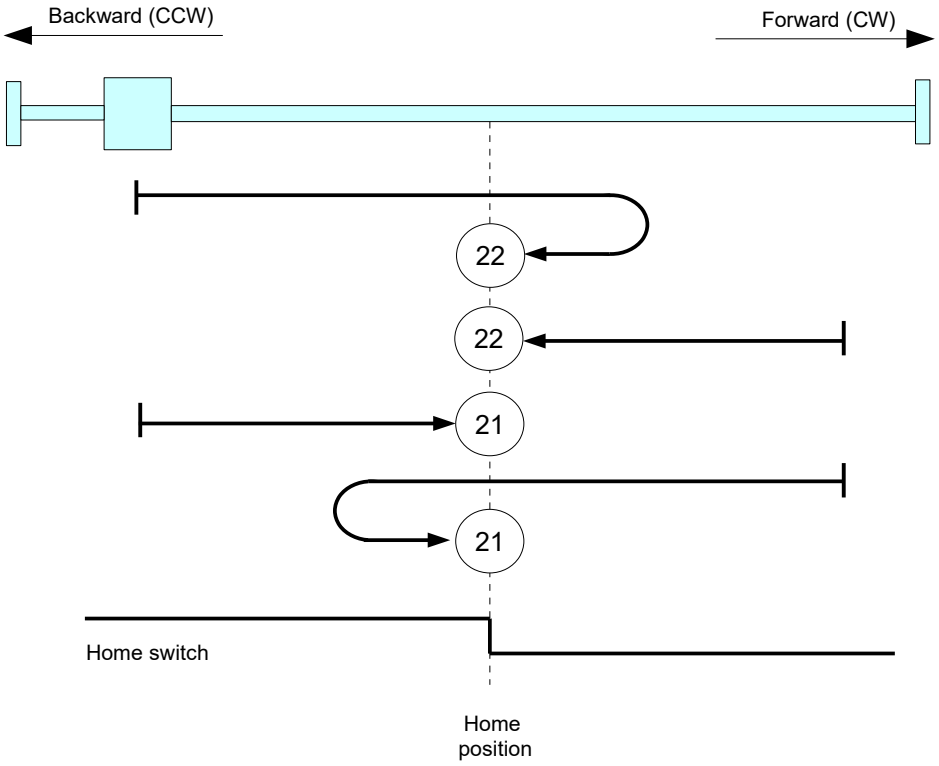
The initial direction of movement is dependent on the state of the home switch. The home position is on the point where the home switch changes its state. The point at which the reversal direction of movement takes place is anywhere after the change of state of the home switch.



The Index pulse, Negative Limit Switch and Positive Limit Switch are not used.

3.5.7 Homing methods 21 and 22

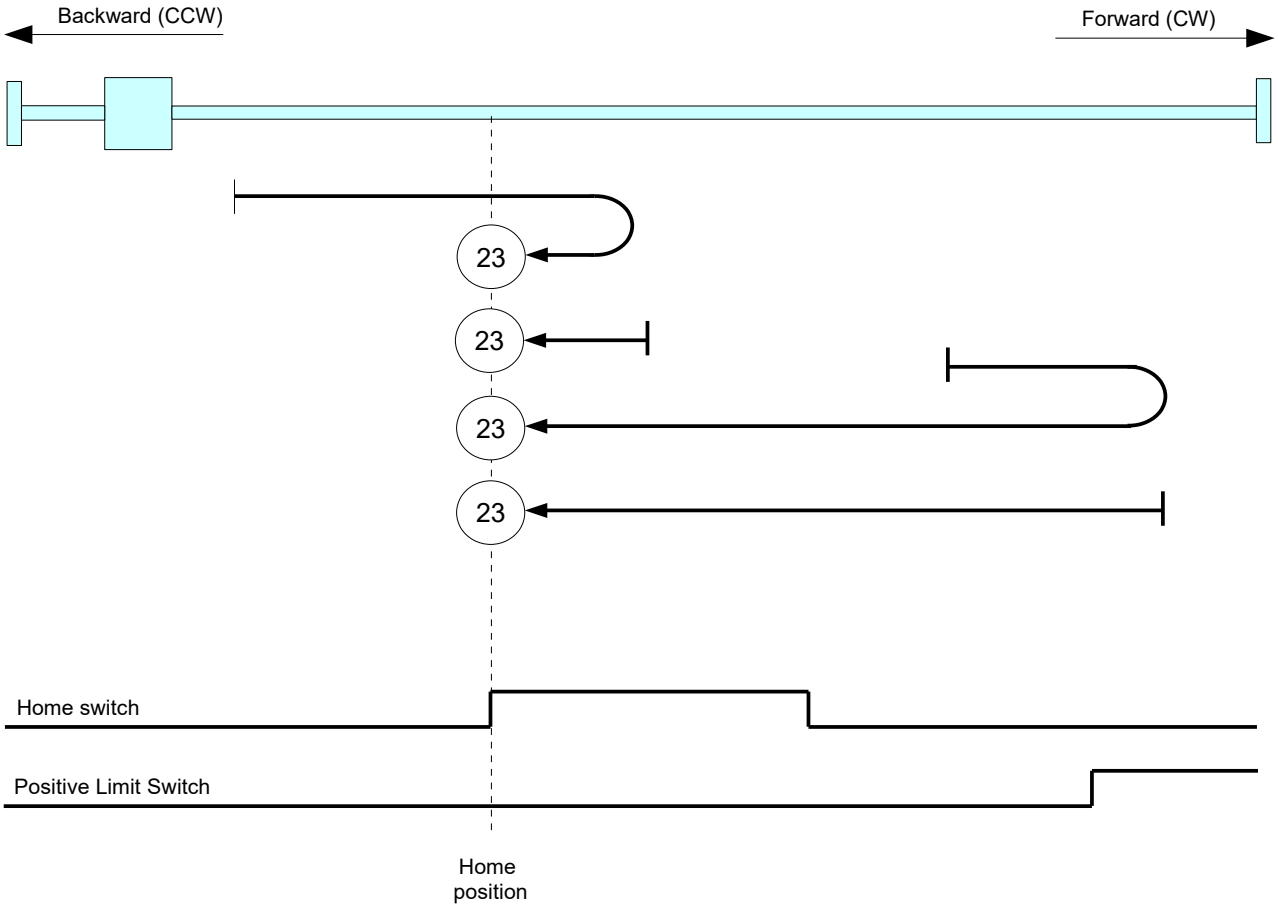
The initial direction of movement is dependent on the state of the home switch. The home position is on the point where the home switch changes its state. The point at which the reversal direction of movement takes place is anywhere after the change of state of the home switch.



The Index pulse, Negative Limit Switch and Positive Limit Switch are not used.

3.5.8 Homing method 23

This method uses a home switch which is present only on a portion of the travel. The initial and final directions of the movements are dependent on the state of the Home switch and Positive Limit Switch.

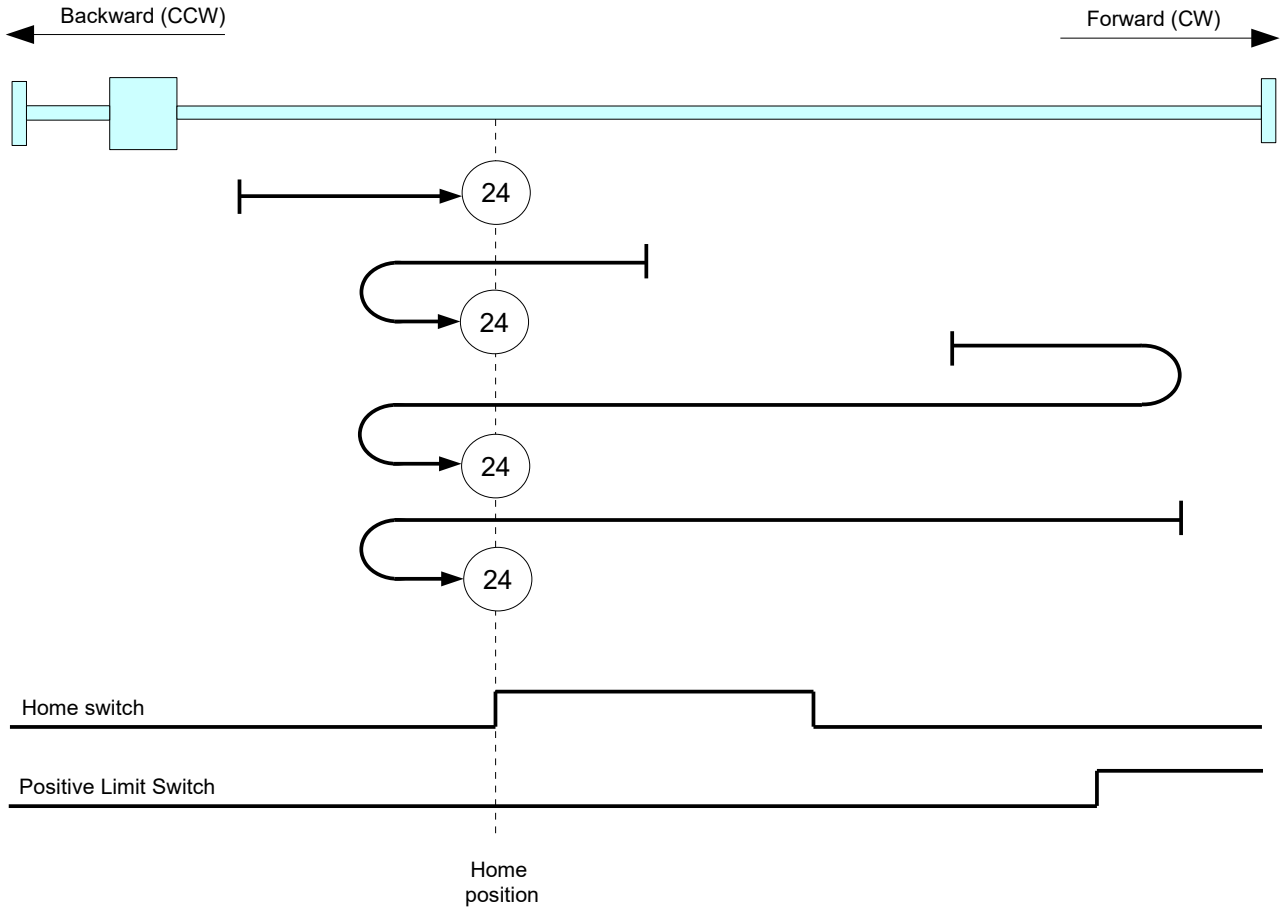


The Index pulse and Negative Limit Switch are not used.

Note:
- This Homing method is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V02r86 or superior

3.5.9 Homing method 24

This method uses a home switch which is present only on a portion of the travel. The initial and final directions of the movements are dependent on the state of the Home switch and Positive Limit Switch.

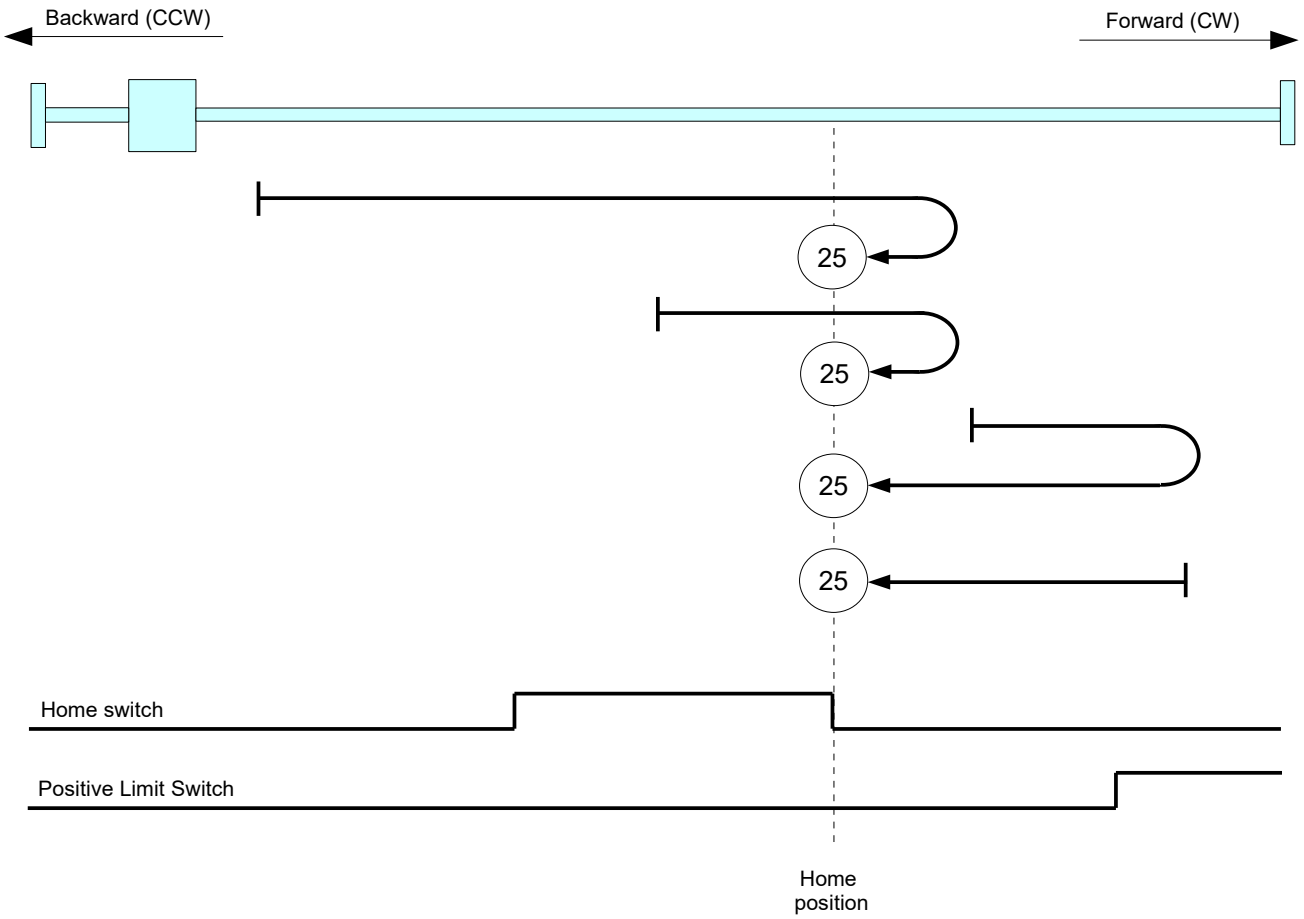


The Index pulse and Negative Limit Switch are not used.

Note:
- This Homing method is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V02r86 or superior

3.5.10 Homing method 25

This method uses a home switch which is present only on a portion of the travel. The initial and final directions of the movements are dependent on the state of the Home switch and Positive Limit Switch.

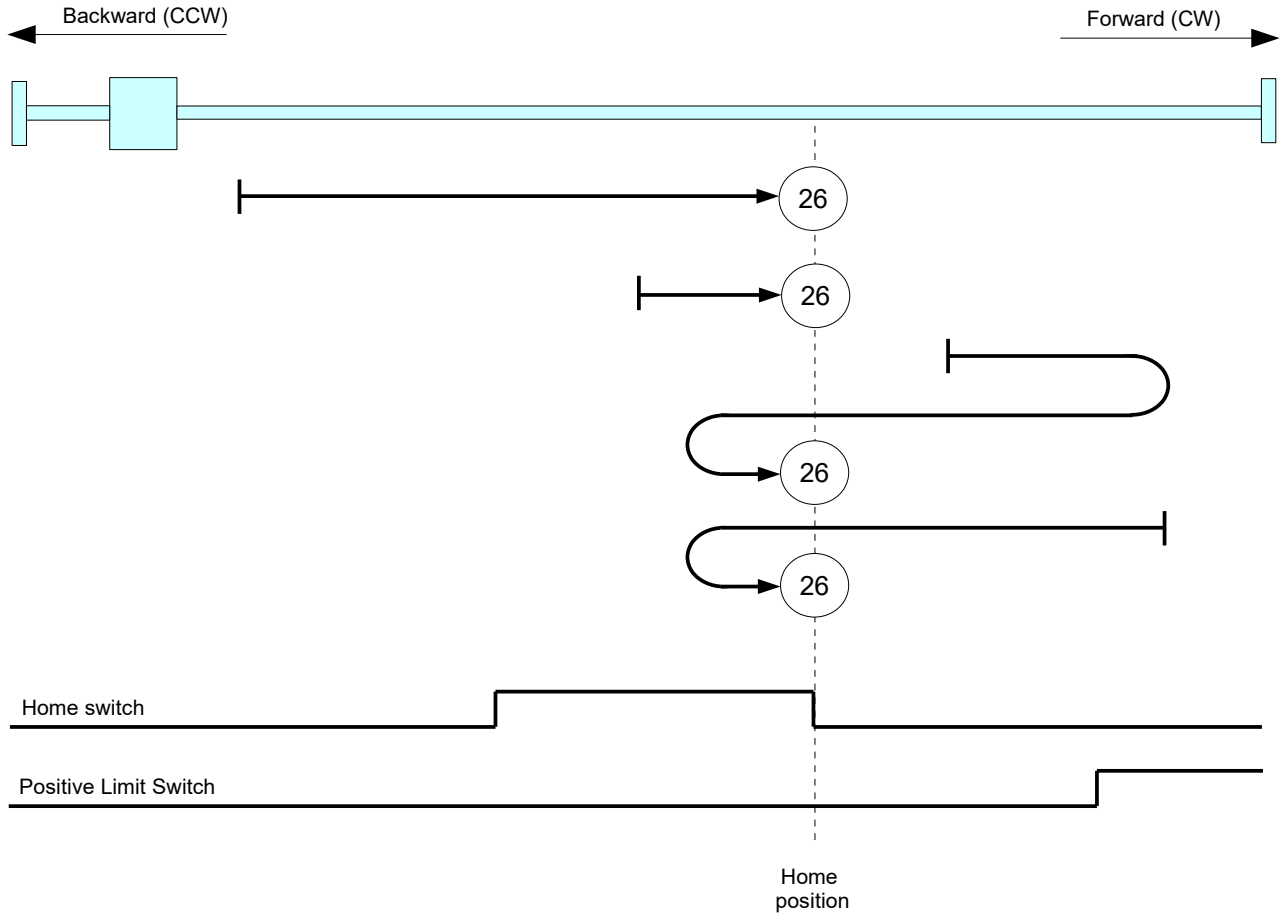


The Index pulse and Negative Limit Switch are not used.

Note:
- This Homing method is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V02r86 or superior

3.5.11 Homing method 26

This method uses a home switch which is present only on a portion of the travel. The initial and final directions of the movements are dependent on the state of the Home switch and Positive Limit Switch.

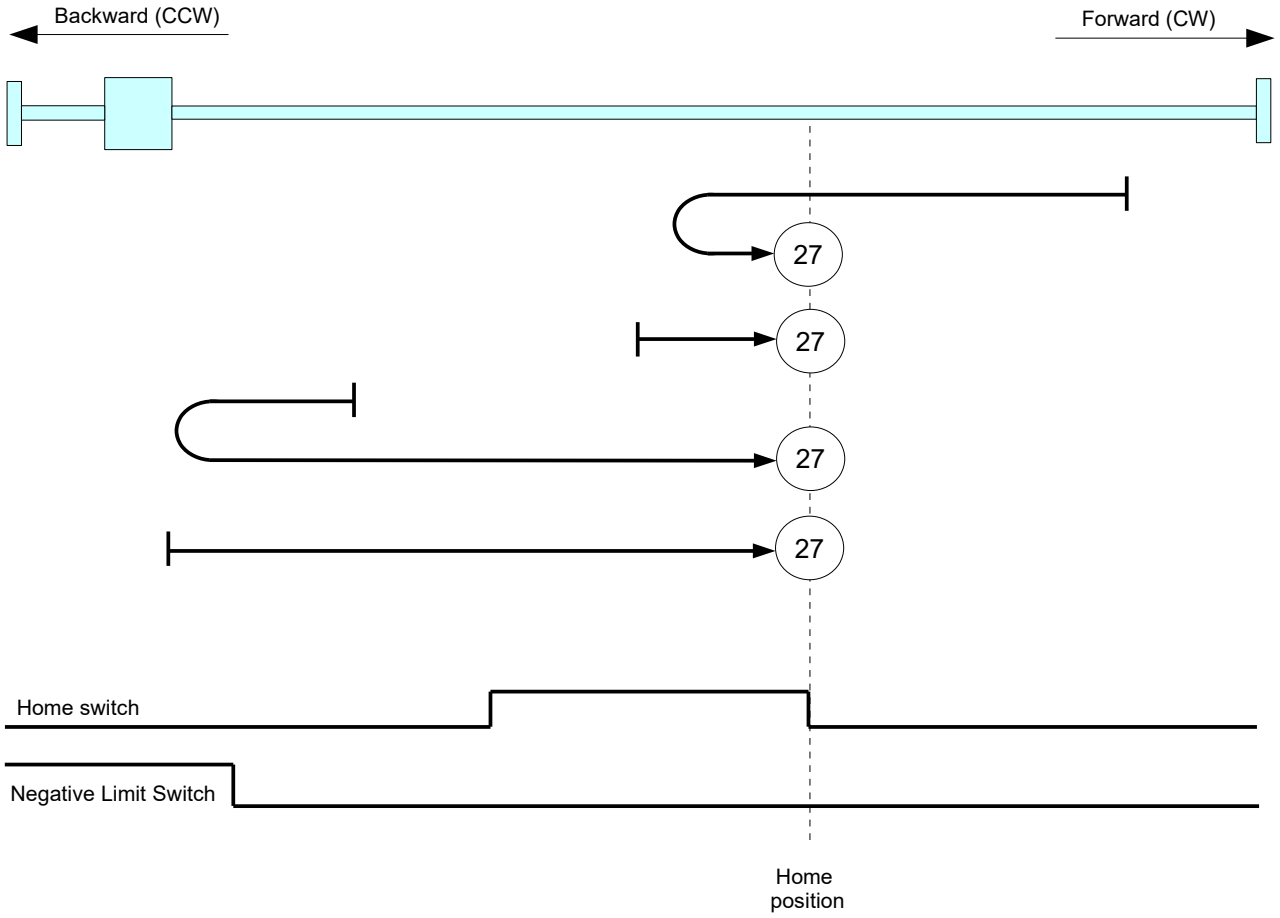


The Index pulse and Negative Limit Switch are not used.

Note:
- This Homing method is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V02r86 or superior

3.5.12 Homing method 27

This method uses a home switch which is present only on a portion of the travel. The initial and final directions of the movements are dependent on the state of the Home switch and Negative Limit Switch.

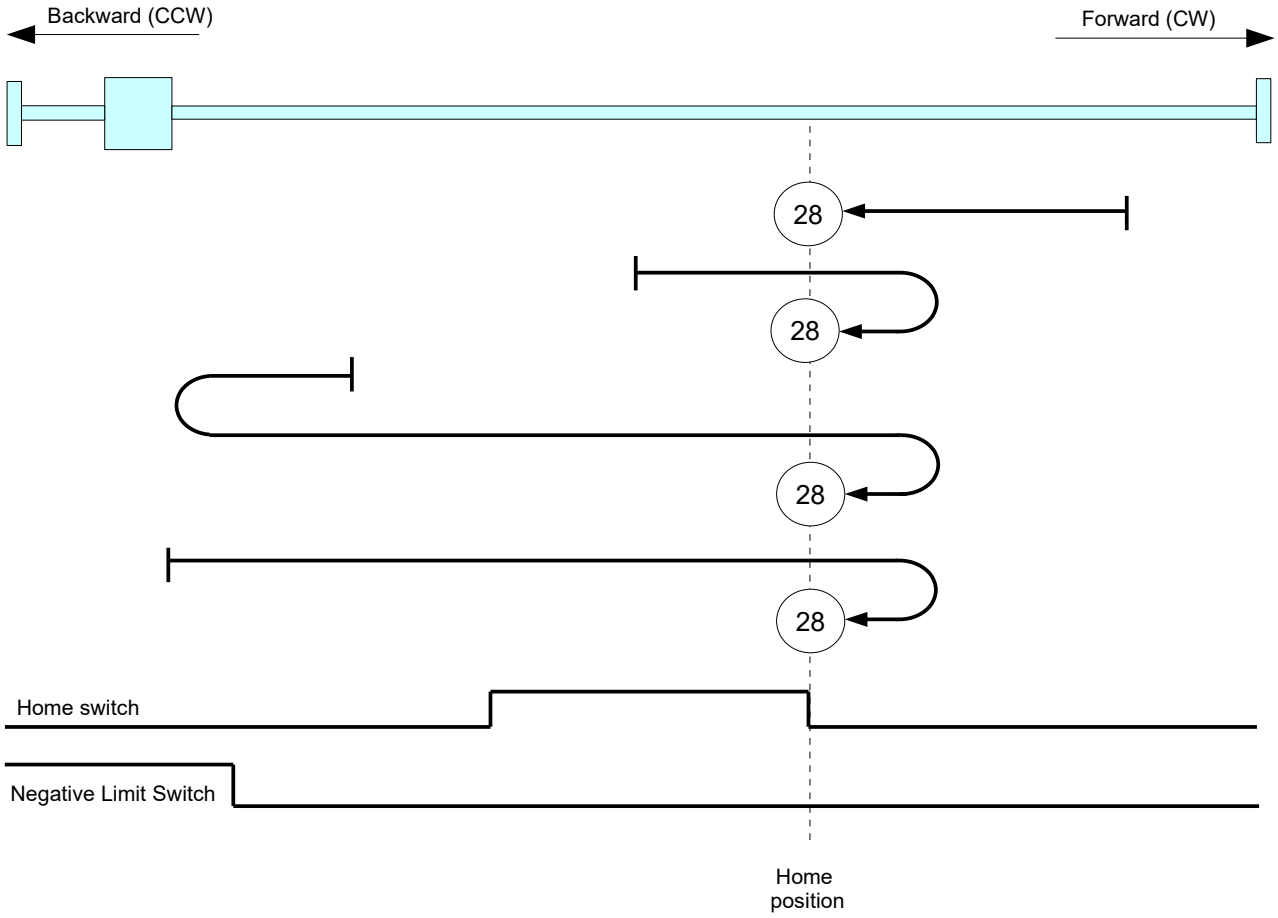


The Index pulse and Positive Limit Switch are not used.

Note:
- This Homing method is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V02r86 or superior

3.5.13 Homing method 28

This method uses a home switch which is present only on a portion of the travel. The initial and final directions of the movements are dependent on the state of the Home switch and Negative Limit Switch.



The Index pulse and Positive Limit Switch are not used.

Note:
- This Homing method is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V02r86 or superior

3.5.14 Homing method 35

In this method, no homing movement is executed. All position values (objects 6062h,6063h,6064h) are set to *Preset Homing Position* (default value = 0).

3.5.15 Homing method 37

In this method, no homing movement is executed. All position values (objects 6062h,6063h,6064h) are set to *Home Offset* (default value = 0).

Note:

- This Homing method is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior

3.6 Profile Velocity mode (pv)

This mode is used to move the motor only by mean of motor velocity. A target velocity is applied to the trajectory generator and this generates a velocity demand value.

The trajectory generator support only linear ramp (trapezoidal profile), with separate parameters for acceleration and deceleration.

3.6.1 Profile Velocity mode related objects

Index	*Object*	*Name*	*Type*	*Attr.*	*M/O*
6040.0 h	VAR	Controlword	UNSIGNED16	rw	M
6041.0 h	VAR	Statusword	UNSIGNED16	ro	M
6083.0 h	VAR	Profile acceleration	UNSIGNED32	rw	M
6084.0 h	VAR	Profile deceleration	UNSIGNED32	rw	O
6085.0 h	VAR	Quick stop deceleration	UNSIGNED32	rw	O
606B.0 h	VAR	Velocity demand value	INTEGER32	ro	M
606C.0 h	VAR	Velocity actual value	INTEGER32	ro	M
606D.0 h	VAR	Velocity window	UNSIGNED16	rw	O
606E.0 h	VAR	Velocity window time	UNSIGNED16	rw	O
60FF.0 h	VAR	Target velocity	INTEGER32	rw	M

3.7 Cyclic Synchronous Position mode (csp)

With this mode, the trajectory generator is located in the control device, not in the drive device. In cyclic synchronous manner, it provides a target position to the drive device, which performs position control and velocity control.

3.7.1 Cyclic Synchronous Position mode related objects

Index	*Object*	*Name*	*Type*	*Attr.*	*M/O*
6040.0 h	VAR	Controlword	UNSIGNED16	rw	M
6041.0 h	VAR	Statusword	UNSIGNED16	ro	M
6064.0 h	VAR	Position actual value	INTEGER32	ro	M
6065.0 h	VAR	Following error window	UNSIGNED32	rw	O
6066.0 h	VAR	Following error time out	UNSIGNED16	rw	O
606C.0 h	VAR	Velocity actual value	INTEGER32	ro	O
607A.0 h	VAR	Target position	INTEGER32	rw	M
6085.0 h	VAR	Quick stop deceleration	UNSIGNED32	rw	O
6086.0 h	VAR	Motion profile type	INTEGER16	rw	O
60C2.0 h	RECORD	Interpolation time period	Interpolation time period record	rw	O
60F4.0 h	VAR	Following error actual value	INTEGER32	ro	O

3.8 Cyclic Synchronous Velocity mode (csv)

With this mode, the trajectory generator is located in the control device, not in the drive device. In cyclic synchronous manner, it provides a target velocity to the drive device, which performs velocity control.

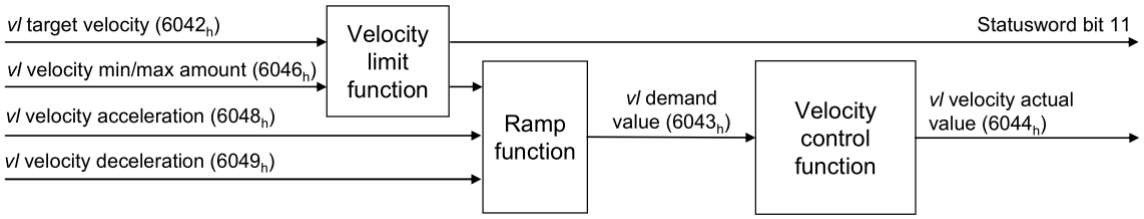
3.8.1 Cyclic Synchronous Velocity mode related objects

Index	*Object*	*Name*	*Type*	*Attr.*	*M/O*
6040.0 h	VAR	Controlword	UNSIGNED16	rw	M
6041.0 h	VAR	Statusword	UNSIGNED16	ro	M
6064.0 h	VAR	Position actual value	INTEGER32	ro	M
606C.0 h	VAR	Velocity actual value	INTEGER32	ro	O
6085.0 h	VAR	Quick stop deceleration	UNSIGNED32	rw	O
6086.0 h	VAR	Motion profile type	INTEGER16	rw	O
60C2.0 h	RECORD	Interpolation time period	Interpolation time period record	rw	O
60FF.0 h	VAR	Target velocity	INTEGER32	rw	M

3.9 Velocity mode (vl)

This mode is used for applications that use a velocity set-point and a controlword for switching the drive device on and off.

All drive devices using this profile and supporting the velocity mode shall implement the mandatory objects and there functionality as shown in figure below :

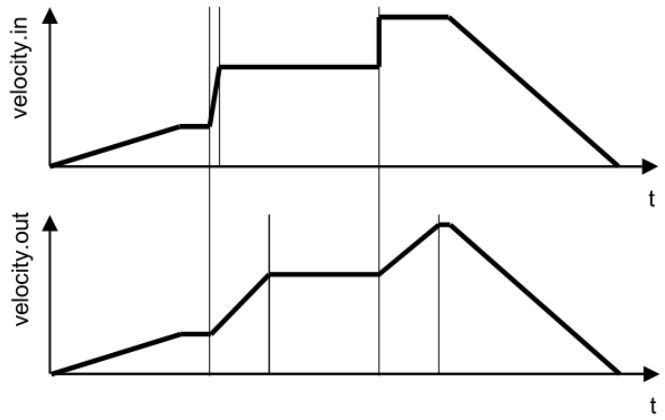


Velocity limit function

The limits in the velocity limit function may be given in user-specific units by including the *vl* dimension factor in the velocity limit or in rotations per minute (rpm). The limit-value message is generated if the input value of the speed limit results in a value outside the speed limit's operating range. The limit-value message is mapped in the Statusword (bit11).

Ramp function

The velocity output is equal to the input as long as the changes are below as defined in *vl* velocity acceleration, *vl* velocity deceleration vl and velocity quickstop.



Velocity control function

On the basis of the *vl* velocity demand, the velocity control function provides the *vl* control-effort.

Factor function

The factor function multiplies the input variables by the assigned factors. The factor shall have a value of 1, if it is not implemented.

Note:
- Velocity mode is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior

3.9.1 Velocity mode related objects

Index	*Object*	*Name*	*Type*	*Attr.*	*M/O*
6040.0 h	VAR	Controlword	UNSIGNED16	rw	M
6041.0 h	VAR	Statusword	UNSIGNED16	ro	M
6042.0 h	VAR	vl Target Velocity	INTEGER16	rw	M
6043.0 h	VAR	vl Velocity Demand	INTEGER16	ro	M
6044.0 h	VAR	vl Control Effort (vl Velocity Actual Value)	INTEGER16	ro	M
6046 h	ARRAY	vl Velocity Min Max Amount	UNSIGNED 32	rw	M
6048 h	RECORD	vl Velocity Acceleration	vl velocity acceleration deceleration	rw	M
6049 h	RECORD	vl Velocity Deceleration	vl velocity acceleration deceleration	rw	M
604C h	ARRAY	vl Dimension Factor	INTEGER32	rw	O

Note:
- Velocity mode is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior

3.10 Touch Probe functionality

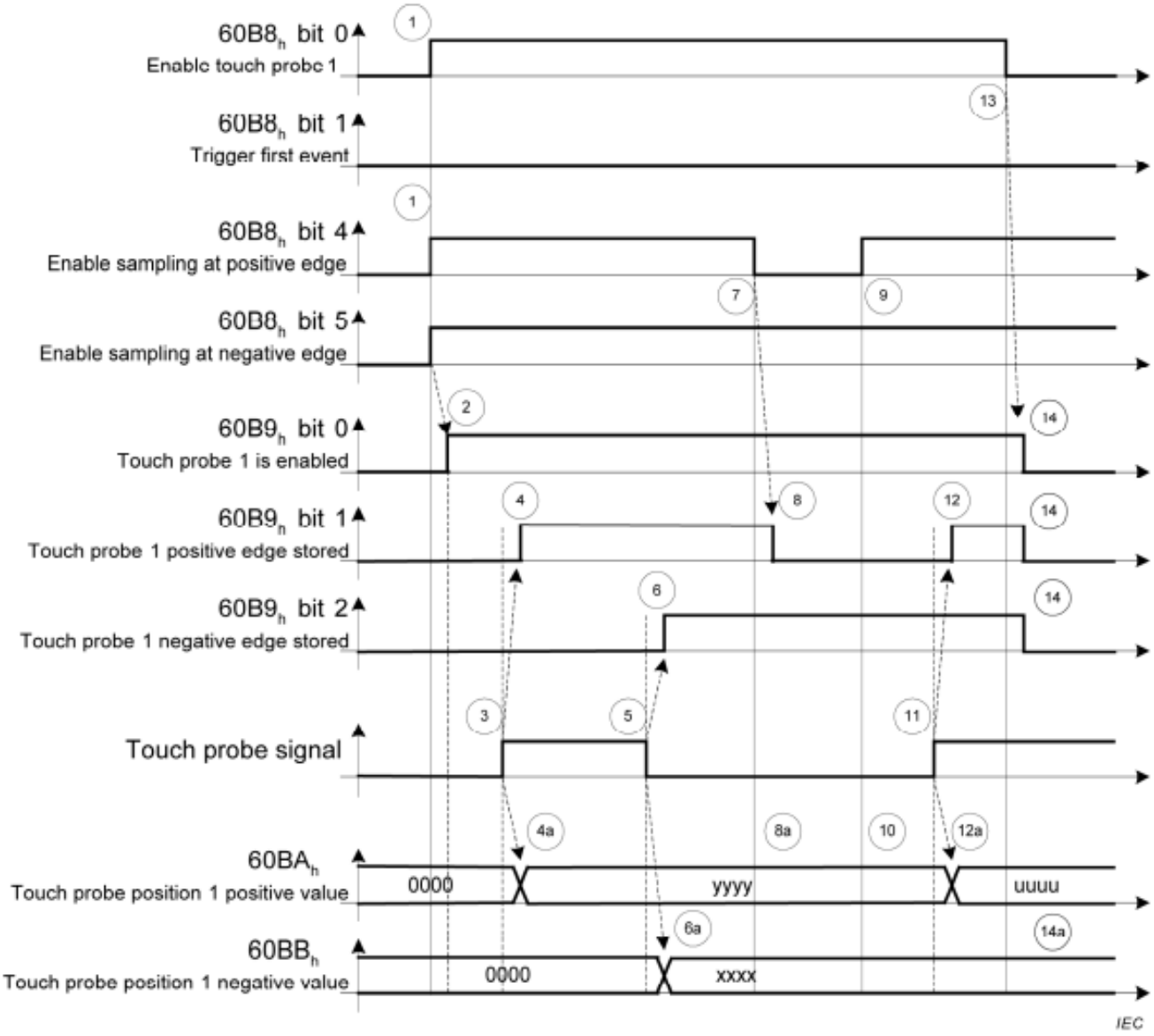
The Touch Probe functionality shall provide the position value of the touch probe 1 and/or touch probe 2 at positive/negative edge.

3.10.1 Touch Probe functionality related objects

Index	*Object*	*Name*	*Type*	*Attr.*	*M/O*
60B8.0h	VAR	Touch Probe Function	UNSIGNED16	rw	M
60B9.0h	VAR	Touch Probe Status	UNSIGNED16	ro	M
60BA.0h	VAR	Touch Probe 1 Positive Edge Position	INTEGER32	ro	M
60BB.0h	VAR	Touch Probe 1 Negative Edge Position	INTEGER32	ro	M
60BC.0h	VAR	Touch Probe 2 Positive Edge Position	INTEGER32	ro	M
60BD.0h	VAR	Touch Probe 2 Negative Edge Position	INTEGER32	ro	M
60D0h	ARRAY	Touch Probe Source	INTEGER16	rw	R
60D5.0h	VAR	Touch Probe 1 Positive Edge Counter	UNSIGNED16	ro	O
60D6.0h	VAR	Touch Probe 1 Negative Edge Counter	UNSIGNED16	ro	O
60D7.0h	VAR	Touch Probe 2 Positive Edge Counter	UNSIGNED16	ro	O
60D8.0h	VAR	Touch Probe 2 Negative Edge Counter	UNSIGNED16	ro	O
2082.0h	VAR	Touch Probe 1 Filter	UNSIGNED32	rw	O
2083.0h	VAR	Touch Probe 2 Filter	UNSIGNED32	rw	O

(*) This functionality is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r11 or superior

3.10.2 Touch Probe example



Number	Touch probe behavior	
(1)	$60B8_h$, bit 0 = 1_b	Enable touch probe 1
	$60B8_h$, bit 1, 4, 5	Configure and enable touch probe 1 positive and negative edge
(2)	→ $60B9_h$, bit 0 = 1_b	Status "Touch probe 1 enabled" is set
(3)	External touch probe signal has positive edge	
(4)	→ $60B9_h$, bit 1 = 1_b	Status "Touch probe 1 positve edge stored" is set
(4a)	→ $60BA_h$	Touch probe position 1 positive value is stored
(5)	External touch probe signal has negative edge	
(6)	→ $60B9_h$, bit 2 = 1_b	Status "Touch probe 1 negative edge stored" is set
(6a)	→ $60BB_h$	Touch probe position 1 negative value is stored
(7)	$60B8_h$, bit 4 = 0_b	Sample positive edge is disabled
(8)	→ $60B9_h$, bit 0 = 0_b	Status "Touch probe 1 positve edge stored" is reset
(8a)	→ $60BA_h$	Touch probe position 1 positive value is not changed
(9)	$60B8_h$, bit 4 = 1_b	Sample positive edge is enabled
(10)	→ $60BA_h$	Touch probe position 1 positive value is not changed
(11)	External touch probe signal has positive edge	
(12)	→ $60B9_h$, bit 1 = 1_b	Status "Touch probe 1 positve edge stored" is set
(12a)	→ $60BA_h$	Touch probe position 1 positive value is stored
(13)	$60B8_h$, bit 0 = 0_b	Touch probe 1 is disabled
(14)	→ $60B9_h$, bit 0, 1, 2 = 0_b	Status bits are reset
(14a)	→ $60BA_h$, $60BB_h$	Touch probe position 1 positive/negative value are not changed

4.0 Object Dictionary

The following tables show whole CANopen objects implemented by Slim Line drives.

CiA Draft Standard 301 (V4.01) :

Index (hex)	SubIndex (hex)	Name	Type	Attr.	Description
1000	0	Device Type	Unsigned32	ro	Device Type : 00000000H
1001	0	Error Register	Unsigned8	ro	Error Register (only Bit#0 used)
1002	0	Manufacturer_Specific_Error	Unsigned32	ro	Manufacturer Specific Error
1005	0	COB-ID SYNC	Unsigned32	ro	Cob-Id SYNC
1008	0	Manufacturer device name	Visible String	ro	Manufacturer device name
100A	0	Manufacturer software version	Visible String	ro	Manufacturer software version
100C	0	Guard Time	Unsigned16	rw	Nodeguarding time
100D	0	Life Time Factor	Unsigned8	rw	Nodeguarding Life Time
1010	1	Save Parameters	Unsigned32	rw	Save all Parameters in nv memory
1011	1	Restore all Parameters	Unsigned32	rw	Restore all default Parameters
1014	0	COB_ID Emergency	Unsigned32	ro	Cob-Id EMCY
1017	0	Producer_Heartbeat_Time	Unsigned16	rw	Producer Heartbeat Time
1018	1	Vendor ID	Unsigned32	ro	Vendor - ID (4BH)
1018	2	Product Code	Unsigned32	ro	Drive Hardware Code
1018	3	Revision Number	Unsigned32	ro	Drive Hardware Revision Number
1018	4	Serial_Number	Unsigned32	ro	Drive Serial Number
1200	0 ÷ 2	1st Server SDO Parameters	SDO Par.	ro	Server SDO #1 Parameters
1400	1	RX_PDO1_Cob_Id	PDO CommPar	rw	Cob Id RX PDO 1
1400	2	RX_PDO1_Tx_Type	PDO CommPar	ro	Transmission Type Rx PDO #1
1401	1	RX_PDO2_Cob_Id	PDO CommPar	rw	Cob Id RX PDO 2
1401	2	RX_PDO2_Tx_Type	PDO CommPar	ro	Transmission Type Rx PDO #2
1402	1	RX_PDO3_Cob_Id(1)	PDO CommPar	rw	Cob Id RX PDO 3
1402	2	RX_PDO3_Tx_Type(1)	PDO CommPar	ro	Transmission Type Rx PDO #3
1403	1	RX_PDO4_Cob_Id(1)	PDO CommPar	rw	Cob Id RX PDO 4
1403	2	RX_PDO4_Tx_Type(1)	PDO CommPar	ro	Transmission Type Rx PDO #4
1600÷1603	0 ÷ 4	Receive PDOs (1 ÷ 4) mapping	PDO Mapping	ro/rw(1)	Mapping RX PDOs
1800	1	TX_PDO1_Cob_Id	PDO CommPar	rw	Cob Id TX PDO 1
1800	2	TX_PDO1_Tx_Type	PDO CommPar	ro	Transmission Type Tx PDO #1
1800	5	TX_PDO1_Event_Timer	PDO CommPar	rw	Timer Tx PDO #1
1801	1	TX_PDO2_Cob_Id	PDO CommPar	rw	Cob Id TX PDO 2
1801	2	TX_PDO2_Tx_Type	PDO CommPar	ro	Transmission Type Tx PDO #2
1801	5	TX_PDO2_Event_Timer	PDO CommPar	rw	Timer Tx PDO #2
1802	1	TX_PDO3_Cob_Id(1)	PDO CommPar	rw	Cob Id TX PDO 3
1802	2	TX_PDO3_Tx_Type(1)	PDO CommPar	ro	Transmission Type Tx PDO #3
1802	5	TX_PDO3_Event_Timer(1)	PDO CommPar	rw	Timer Tx PDO #3
1803	1	TX_PDO4_Cob_Id(1)	PDO CommPar	rw	Cob Id TX PDO 4
1803	2	TX_PDO4_Tx_Type(1)	PDO CommPar	ro	Transmission Type Tx PDO #4
1803	5	TX_PDO4_Event_Timer(1)	PDO CommPar	rw	Timer Tx PDO #4
1A00÷1A03	0 ÷ 4	Transmit PDOs (1 ÷ 4) mapping	PDO Mapping	ro/rw(1)	Mapping TX PDOs

CiA Draft Standard Proposal 401 (V1.4) :

Index (hex)	Subindex (hex)	Name	Type	Attr.	Description
6000	**1**	B0_Digital_Inputs	Unsigned8	ro	Bank #0 Digital Inputs
6100	**2**	B1_Digital_Inputs	Unsigned8	ro	Bank #1 Digital Inputs
6200	**1**	B0_Digital_Outputs	Unsigned8	rw	Bank #0 Digital Outputs
6300	**2**	B1_Digital_Outputs	Unsigned8	rw	Bank #1 Digital Outputs
6404	**1 ÷ 2**	Analog_In[0÷1]	Integer16	ro	Analog Inputs
6414	**1 ÷ 2**	Analog_Out[0÷1]	Unsigned16	rw	Analog Outputs

CiA Draft Standard Proposal 402 (V1.1) :

Index (hex)	Subindex (hex)	Name	Type	Attr.	Description
603F	0	Error_Code	Unsigned16	ro	Error Code
6040	0	Controlword	Unsigned16	rw	Controlword
6041	0	Statusword	Unsigned16	ro	Statusword
6042	0	vl Target Velocity(2)	Integer16	rw	vl Target Velocity
6043	0	vl Velocity Demand(2)	Integer16	ro	vl Velocity Demand
6044	0	vl Control Effort(2) (vl Velocity Actual value)(2)	Integer16	ro	vl Control Effort (vl Velocity Actual value)
6046	0	vl Velocity min max amount(2)	Unsigned8	ro	Number of entries
6046	1	vl Velocity min amount(2)	Unsigned32	rw	vl Velocity min amount
6046	2	vl Velocity max amount(2)	Unsigned32	rw	vl Velocity max amount
6048	0	vl Velocity acceleration(2)	Unsigned8	ro	Number of entries
6048	1	vl Velocity acceleration(2)	Unsigned32	rw	Delta speed
6048	2	vl Velocity acceleration(2)	Unsigned16	rw	Delta time
6049	0	vl Velocity deceleration(2)	Unsigned8	ro	Number of entries
6049	1	vl Velocity deceleration(2)	Unsigned32	rw	Delta speed
6049	2	vl Velocity deceleration(2)	Unsigned16	rw	Delta time
604C	0	vl dimension factor(2)	Unsigned8	ro	Number of entries
604C	1	vl dimension factor(2)	Integer32	rw	vl dimension factor numerator
604C	2	vl dimension factor(2)	Integer32	rw	vl dimension factor denominator
605A	0	Quick_stop_option_code	Integer16	rw	Quick stop option
605D	0	Halt_option_code	Integer16	rw	Halt option code
605E	0	Fault_reaction_option_code	Integer16	rw	Fault_reaction_option_code
6060	0	Modes_of_operation	Integer8	rw	Modes of operation
6061	0	Modes_of_operation_display	Integer8	ro	Modes of operation display
6062	0	Position_demand_value	Integer32	ro	Position demand value
6063	0	Position_Actual_Value*	Integer32	ro	Motor Actual Position (inc)
6064	0	Position_Actual_Value	Integer32	ro	Motor Actual Position
6065	0	Following_error_window	Unsigned32	rw	Following error window
6066	0	Following_error_time_out	Unsigned16	rw	Following error timeout
6067	0	Position_window	Unsigned32	rw	Position window
6068	0	Position_window_time	Unsigned16	rw	Position window time
606A	0	Sensor_selection_code	Integer16	rw	Sensor selection code
606B	0	Velocity_demand_value	Integer32	ro	Velocity demand value
606C	0	Velocity_Actual_Value	Integer32	ro	Motor Actual Velocity
606D	0	Velocity window	Unsigned16	rw	Velocity window
606E	0	Velocity window time	Unsigned16	rw	Velocity window time
6078	0	Current_Actual_Value	Unsigned16	ro	Motor Actual Current
607A	0	Target_Position	Integer32	rw	Target Position
607C	0	Home_Offset	Integer32	rw	Home Offset
607F	0	Max_Profile_Velocity	Unsigned32	rw	Motor Maximum Velocity
6081	0	Profile_Velocity	Unsigned32	rw	Profile Velocity
6083	0	Profile_acceleration	Unsigned32	rw	Profile acceleration
6084	0	Profile_deceleration	Unsigned32	rw	Profile deceleration
6085	0	Quick_Stop_deceleration	Unsigned32	rw	Quick Stop Deceleration
6086	0	Motion profile type	Integer16	rw	Motion profile type
6089	0	Position_notation_index (5)	Integer8	rw	Position notation index
608A	0	Position_dimension_index (5)	Unsigned8	rw	Position dimension index
608B	0	Velocity_notation_index (5)	Integer8	rw	Velocity notation index
608C	0	Velocity_dimension_index (5)	Unsigned8	rw	Velocity dimension index
608D	0	Acceleration_notation_index (5)	Integer8	rw	Acceleration notation index
608E	0	Acceleration_dimension_index (5)	Unsigned8	rw	Acceleration dimension index
608F	0	Position_encoder_resolution	Unsigned8	ro	Number of entries
608F	1	Position_encoder_resolution	Unsigned32	rw	Encoder increments
608F	2	Position_encoder_resolution	Unsigned32	rw	Motor revolutions
6091	0	Gear_Ratio(4) (Highest sub-index supported)	Unsigned8	ro	Highest sub-index supported
6091	1	Gear_Ratio(4) (Motor_Shaft_Revolutions)	Unsigned32	rw	Motor Shaft Revolutions

6091	**2**	Gear_Ratio[4] (Driving_Shaft_Revolutions)	Unsigned32	rw	Driving Shaft Revolutions
6092	**0**	Feed_Constant[4] (Highest sub-index supported)	Unsigned8	ro	Highest sub-index supported
6092	**1**	Feed_Constant[4] (Feed)	Unsigned32	rw	Feed
6092	**2**	Feed_Constant[4] (Shaft Revolutions)	Unsigned32	rw	Shaft Revolutions
6098	**0**	Homing_method	Integer8	rw	Homing method
6099	**0**	Homing_speeds	Unsigned8	ro	Number of entries
6099	**1**	Homing_speeds	Unsigned32	rw	Speed during search for Home Switch
6099	**2**	Homing_speeds	Unsigned32	rw	Speed during search for Zero
609A	**0**	Homing_acceleration	Unsigned32	rw	Homing acceleration
60B8	**0**	Touch_Probe_Function[3]	Unsigned16	rw	Touch Probe Function
60B9	**0**	Touch_Probe_Status[3]	Unsigned16	ro	Touch Probe Status
60BA	**0**	Touch_Probe1_Positive_Edge_Position[3]	Integer32	ro	Touch Probe 1 Positive Edge Position
60BB	**0**	Touch_Probe1_Negative_Edge_Position[3]	Integer32	ro	Touch Probe 1 Negative Edge Position
60BC	**0**	Touch_Probe2_Positive_Edge_Position[3]	Integer32	ro	Touch Probe 2 Positive Edge Position
60BD	**0**	Touch_Probe2_Negative_Edge_Position[3]	Integer32	ro	Touch Probe 2 Negative Edge Position
60C1	**0**	Interpolation_data_record	Unsigned8	ro	Number of entries
60C1	**1**	Interpolation_data_record	Integer32	rw	Position set point
60C2	**0**	Interpolation_time_period	Unsigned8	ro	Number of entries
60C2	**1**	Interpolation_time_period	Unsigned8	rw	Interpolation time units
60C2	**2**	Interpolation_time_period	Signed8	rw	Interpolation time index
60C3	**0**	Interpolation_sync_definition	Unsigned8	ro	Number of entries
60C3	**1**	Interpolation_sync_definition	Unsigned8	rw	Synchronize on group
60C3	**2**	Interpolation_sync_definition	Unsigned8	rw	Ip sync every n event
60D0	**0**	Touch_Probe_Source[3]	Unsigned8	rw	Touch Probe Source (sub index supported)
60D0	**1**	Touch_Probe1_Source[3]	Integer16	rw	Touch Probe 1 Source
60D0	**2**	Touch_Probe2_Source	Integer16	rw	Touch Probe 2 Source
60D5	**0**	Touch_Probe1_Positive_Edge_Counter[3]	Unsigned16	ro	Touch Probe 1 Positive Edge Counter
60D6	**0**	Touch_Probe1_Negative_Edge_Counter[3]	Unsigned16	ro	Touch Probe 1 Negative Edge Counter
60D7	**0**	Touch_Probe2_Positive_Edge_Counter[3]	Unsigned16	ro	Touch Probe 2 Positive Edge Counter
60D8	**0**	Touch_Probe2_Negative_Edge_Counter[3]	Unsigned16	ro	Touch Probe 2 Negative Edge Counter
60EF	**0**	Motor_Resolution[4]	Unsigned32	ro	Motor Resolution
60F4	**0**	Following_error_actual_value	Integer32	ro	Following error actual value
60FD	**0**	Digital_Inputs[1]	Unsigned32	ro	Digital Inputs
60FF	**0**	Target_Velocity	Integer32	rw	Target Velocity
6502	**0**	Supported_drive_modes	Unsigned32	ro	Supported drive modes

CANopen Manufacturer Specific :

Index (hex)	Subindex (hex)	Name	Type	Attr.	Description
2004	0	Dips	Unsigned16	ro	Dips Switches Status
2005	1	Min_Current	Unsigned16	rw	Motor Minimum Current
2005	2	Max_Current	Unsigned16	rw	Motor Maximum Current
2005	3	Boost_Current	Unsigned16	rw	Motor Boost Current
2005	4	Nominal_Current	Unsigned16	rw	Motor Nominal Current
2005	6	Current_Gain_Kp (7)	Float	rw	Current Gain Kp
2005	6	Motor_R (8)	Unsigned16	rw	Motor R
2005	7	Current_Gain_Ki (7)	Float	rw	Current Gain Ki
2005	7	Motor _L (8)	Unsigned16	rw	Motor L
2005	8	Current_Actual_Value (5)	Unsigned16	ro	Motor Actual Current
2005	8	Motor_R_detected (8)	Unsigned16	ro	Motor R detected
2005	9	Motor_L_detected (8)	Unsigned16	ro	Motor L detected
2007	0	Encoder_Actual_Value[0]	Integer32	rw	Encoder #0 Actual Position
2008	0	Encoder_Actual_Value[1]	Integer32	rw	Encoder #1 Actual Position
2009	0	Drive_Voltage_Actual_Value	Unsigned16	ro	Drive Voltage Actual Value
200A	0	Drive_Temperature_Actual_Value	Unsigned16	ro	Drive Temperature Actual Value
2010	0	Min_Profile_Velocity	Unsigned16	rw	Minimum Ramp Velocity
2012	0	Motor_Parameters	Unsigned8	ro	Highest sub-index supported
2012	1	Motor_Parameters (Motor_Step_Angle)	Unsigned16	rw	Motor Step Angle
2012	2	Motor_Parameters (Motor_Poles)	Unsigned16	rw	# of Motor Poles
2013	0	Motor_Factor(4)	Unsigned8	ro	Highest sub-index supported
2013	1	Motor_Factor(4) (Velocity_Factor_Numerator)	Unsigned32	rw	Velocity Factor Numerator
2013	2	Motor_Factor(4) (Velocity_Factor_Denominator)	Unsigned32	rw	Velocity Factor Denominator
2013	3	Motor_Factor(4) (Acceleration_Factor_Numerator)	Unsigned32	rw	Acceleration_Factor_Numerator
2013	4	Motor_Factor(4) (Acceleration_Factor_Denominator)	Unsigned32	rw	Acceleration_Factor_Denominator
2023	0	CAN_Busoff_Counter	Unsigned16	ro	# of CAN Bus Off Occurred
2027	0	RotoSwitches	Unsigned16	ro	RotoSwitches Status
2080	0	Preset_Homing_Position	Integer32	rw	Preset Homing Position
2081	0	Drive_Homing_Inputs_Setting	Unsigned32	rw	Drive Homing Inputs Setting
2082	0	Touch_Probe1_Filter(3)	Unsigned32	rw	Touch_Probe1_Filter
2083	0	Touch_Probe2_Filter(3)	Unsigned32	rw	Touch_Probe2_Filter
2084	0	DS402_Working_Settings	Unsigned32	rw	DS402_Working_Settings
2100	0	Homing_Sequence_Pause	Unsigned32	rw	Homing Sequence Pause
2200	1	Drive_Inputs_Level	Unsigned16	rw	Drive Digital Inputs Working Level
2200	2	Drive_Working_Settings	Unsigned16	rw	Drive Working Settings
2200	3	Drive_Inputs_Setting	Unsigned32	rw	Drive Digital Inputs Working Setting
2200	5	Drive_CANopen_Config	Unsigned16	rw	CANopen Protocol Configuration
2200	6	Analog_Input0_K_Filter	Unsigned16	rw	K filter for analog input0 sampling
2200	E	Analog_Input0_Type	Unsigned16	ro	Analog Input0 Type
2200	F	Analog_Input1_Type	Unsigned16	ro	Analog Input1 Type
2200	10	Analog_Input1_K_Filter	Unsigned16	rw	K filter for analog input1 sampling
2200	11	Drive_Working_Settings_Extended	Unsigned16	rw	Drive Working Settings Extended
2200	1A	Drive_CANopen_Config_Extended	Unsigned16	rw	CANopen Protocol Configuration
2210	3	Counters_Filter	Unsigned16	rw	Filter for Counter #0 and Counter #1
2211	1	Encoder_Frequency[0]	Unsigned32	ro	Encoder #0 Frequency
2211	2	Encoder_Frequency[1]	Unsigned32	ro	Encoder #1 Frequency
2230	1	Feedback_Max_Displacement (5)	Unsigned32	rw	Feedback Max Displacement
2230	2	Feedback_Actual_Displacement(5)	Integer32	ro	Feedback Actual displacement
2230	3	Feedback_Source_PPR(5)	Unsigned32	rw	Feedback Encoder PPR
2230	4	Feedback_Kp(5)	Unsigned32	rw	Feedback Kp
2230	5	Feedback_Kv(5)	Unsigned32	rw	Feedback Kv
2230	6	Feedback_Kffw_Acc(5)	Unsigned32	rw	Feedback Kffw Acc
2230	7	Feedback_Kffw_Dec(5)	Unsigned32	rw	Feedback Kffw Dec
2230	8	(6) Feedback_Tolerance(5)	Unsigned32	rw	Feedback Tolerance
2230	9	(6) Feedback_Timeout(5)	Unsigned16	rw	Feedback timeout

Index (hex)	Subindex (hex)	Name	Type	Attr.	Description
2230	B	Feedback_Ki[5]	Unsigned32	rw	Feedback Ki
2230	C	Feedback_Ki_Limit[5]	Unsigned16	rw	Feedback Ki limit
2230	D	Feedback_Kalfas[5]	Unsigned16	rw	Feedback Kalfas
2230	E	[6] Feedback_Tau_Torque[5]	Unsigned16	rw	Feedback Tau Torque
2230	F	Feedback_Iq_Min[5]	Unsigned16	rw	Feedback Iq min
2230	10	Feedback_Ka[5]	Unsigned32	rw	Feedback Ka
2230	11	[6] Feedback_Kt[5]	Unsigned16	rw	Feedback Kt
2230	12	Feedback_Boost_Current[5]	Unsigned16	rw	Feedback Boost Current
2230	14	[6] Feedback_Switch_Kff[5]	Unsigned16	rw	Feedback Switch Kff
2230	15	Feedback_Kfbw_Acc[5]	Unsigned32	rw	Feedback Kfbw Acc
2230	16	Feedback_Kfbw_Dec[5]	Unsigned32	rw	Feedback Kfbw Dec
2A00	0	BiSS_Encoder_Actual_Value	Signed32	ro	BiSS_Encoder_Actual_Value
2A02	0	BiSS_Encoder_Status	Unsigned16	ro	BiSS_Encoder_Status
2A04	0	BiSS_Encoder_Config	Unsigned32	rw	Biss_Encoder_Config
2A06	0	BiSS_Encoder_RxErr	Unsigned16	rw	BiSS_Encoder_RxErr
2A08	0	BiSS_Encoder_Offset_Value	Signed32	rw	BiSS_Encoder_Offset_Value
2A0A	0	BiSS_Encoder_Internal_Value	Signed32	ro	BiSS_Encoder_Internal_Value
4000	7	Node Id	Unsigned16	ro	Drive NodeId
4000	8	Baud Rate	Unsigned16	ro	CAN Baud Rate
4004	0	Firmware_Version	Unsigned16	ro	Drive Firmware Version
4004	1	Firmare_Checksum	Unsigned16	ro	Drive Firmware Checksum
4004	2	Boot_Version	Unsigned16	ro	Drive Boot Version
4004	6	Drive_Type	Unsigned16	ro	Drive Type
4004	7	Configuration_Code	Unsigned16	ro	Firmware Configuration Code

Notes :

(1) Available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r67 or superior
(2) Available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r70 or superior
(3) Available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r11 or superior
(4) Available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior
(5) Available only for SW1,SM1A,DMD105 Drives
(6) Not yet handled
(7) Available only on SW1 drive type 9x60 (DMD02)
(8) Available only for CSMD1,SB4,SW4,SM4,SW5 Drives

Name: **Device Type**
Index.Sub: **1000.0H**
Type: Unsigned32
Access: ro
Unit: **--**
Range: **--**
Default Value: 00040192H
Store Supported: --

Description: This object contains information about the device type.

Notes:

Name: **Error register**
Index.Sub: **1001.0H**
Data Type: Unsigned8
Access: ro
Unit: **---**
Range: ---
Default Value: 0
Store Supported: No

Description: This object is an error register for the drive. More than one bit at time could be set to 1, meaning that more than one fault is active. Bit0 is set to 1 if one or more faults are active, is reset to 0 if all faults are cleared.

See **§2.8** for more details.

Bit		Description	
0	**M**	**Generic error**	
1	**O**	Current	
2	**O**	Voltage	
3	**O**	Temperature	
4	**O**	Communication error	
5	**O**	Device profile specific	
6	**O**	Reserved (always 0)	(always 0)
7	**O**	**Manufacturer specific**	

Name: **Manufacturer Specific Error Register**
Index.Sub: **1002.0H**
Data Type: Unsigned32
Access: ro
Unit: **---**
Range: ---
Default Value: 0
Store Supported: No

Description: This object is the specific fault of the SW1 drive. See **§2.8** for more details.
This object is not available in EtherCAT fieldbus.

Name: **COB-ID SYNC**
Index.Sub: **1005.0H**
Type: Unsigned32
Access: ro
Unit: **--**
Range: **--**
Default Value: 80h
Store Supported: --

Description: This object defines the COB-ID of Synchronization Object(SYNC) see **§2.6** for details.

Notes: This object is not available in EtherCAT fieldbus.

Name: **Manufacturer Device Name**
Index.Sub: **1008.0H**
Type: Visible String
Access: ro
Unit: **--**
Range: **--**
Default Value: **--**
Store Supported: --

Description: Contains the manufacturer device name.

Notes:

Name: **Manufacturer Software Version**
Index.Sub: **100A.0H**
Type: Visible String
Access: ro
Unit: **--**
Range: **--**
Default Value: **--**
Store Supported: --

Description: Contains the manufacturer software version description.

Notes:

Name: **Guard Time**
Index.Sub: **100C.0H**
Type: Unsigned16
Access: rw
Unit: Milliseconds
Range: 0 ÷ 65535
Default Value: 0
Store Supported: No

Description: The objects at index 100Ch and 100Dh include the guard time in milliseconds and the life time factor.The life time factor multiplied with the guard time gives the life time for the Life Guarding Protocol. It is 0 if not used.

Notes: This object is not available in EtherCAT fieldbus. This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r74 or superior.

Name: **Life Time Factor**
Index.Sub: **100D.0H**
Type: Unsigned8
Access: rw
Unit: --
Range: 0 ÷ 255
Default Value: 0
Store Supported: No

Description: The life time factor multiplied with the guard time gives the life time for the node guarding protocol. It is 0 if not used.

Notes: This object is not available in EtherCAT fieldbus.
This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r74 or superior.

Name: **Store_Parameters (Save Parameters)**
Index.Sub: **1010.1H**
Type: Unsigned32
Access: rw
Unit: **--**
Range:

Value	Domains
65766173H	Whole domains

Default Value: --
Store Supported: --

Description: This object supports the saving of drive parameters in non volatile memory. On reception of the correct signature the drive stores the current parameters values in non volatile memory. At the next drive switch on the parameters starting value will be equal to the value stored in non volatile memory.

Notes: The storing process takes about 4-5 (worst case) seconds to be completed. If a further store parameters command is sent before the completion of the previous one, the drive will not answer to the communication interface until the previous storing process is completed. The non volatile ram is really written only if data have changed compared to the previous store process.

!!!WARNING!!! : The non volatile memory can be written for a limited number of times (typically 100,000 times), when reached that limit F4 errors (see B Appendix) can occur, and the drive should be sent to EVER for reparation.

Name: **Restore_Parameters**
Index.Sub: **1011.1H**
Type: Unsigned32
Access: rw
Unit: **--**
Range: valid signature = 64616F6CH ('load' in ASCII hex)
Default Value: --
Store Supported: **--**

Description: Restore the drive's parameters default value at reception of the correct signature. At the next drive switch on the parameters starting value will be equal to the factory default value.

Notes: The restoring process takes about 4-5 (worst case) seconds to be completed. If a further restore parameters command is sent before the completion of the previous one, the drive will not answer to the communication interface until the previous storing process is completed. The non volatile ram is really written only if data have changed compared to the previous store process.

!!!WARNING!!! : The non volatile memory can be written for a limited number of times (typically 100,000 times), when reached that limit F4 errors (see B Appendix) can occur, and the drive should be sent to EVER for reparation.

Name: **COB-ID Emergency**
Index.Sub: **1014.0H**
Type: Unsigned32
Access: ro
Unit: **--**
Range: **--**
Default Value: 80h + NodeId
Store Supported: --

Description: This object defines the COB-ID of the Emergency Object (EMCY).

Notes: This object is not available in EtherCAT fieldbus.

Name: **Producer_Heartbeat_Time**
Index.Sub: **1017.0H**
Type: Unsigned16
Access: rw
Unit: ms
Range: 0 ÷ 65535
Default Value: 500
Store Supported: Yes

Description: This object defines the cycle time of the heartbeat message. If is 0 the heartbeat transmission is disabled.

Notes: This object is not available in EtherCAT fieldbus.

Name: **Vendor_ID**
Index.Sub: **1018.1H**
Type: Unsigned32
Access: ro
Unit: **--**
Range: **--**
Default Value: 4BH
Store Supported: No

Description: EVER Vendor ID assigned by CiA.

Notes:

Name: **Product_Code**
Index.Sub: **1018.2H**
Type: Unsigned32
Access: ro
Unit: **--**
Range: **--**
Default Value: --
Store Supported: No

Description: Drive Hardware Code.

Notes:

Name: **Revision_Number**
Index.Sub: **1018.3H**
Type: Unsigned32
Access: ro
Unit: **--**
Range: **--**
Default Value: --
Store Supported: No

Description: Drive Hardware Revision Number.

Notes:

Name: **Serial_Number**
Index.Sub: **1018.4H**
Type: Unsigned32
Access: ro
Unit: **--**
Range: **--**
Default Value: --
Store Supported: No

Description: This buffer contains the drive's serial number. If the serial number has not been stored in the drive this object return FFFFFFFFH.

Notes:

Name: **Server SDO Parameters (Cob-Id Client → Server)**
Index.Sub: **1200.1H**
Type: Unsigned32
Access: ro
Unit: **--**
Range: **--**
Default Value: 600h + NodeId
Store Supported: --

Description: This object defines the COB-ID of the SDO requests sent by the master to the drive.

Notes: This object is not available in EtherCAT fieldbus.

Name: **Server SDO Parameters (Cob-Id Server → Client)**
Index.Sub: **1200.2H**
Type: Unsigned32
Access: ro
Unit: **--**
Range: **--**
Default Value: 580h + NodeId
Store Supported: --

Description: This object defines the COB-ID of the SDO answers sent by the drive to the master.

Notes: This object is not available in EtherCAT fieldbus.

Name: **RX_PDO1_Cob_Id**
Index.Sub: **1400.1H**
Type: Unsigned32
Access: rw
Unit: **--**
Range: 1h÷7FFh,
80000000h (PDO disabled)
Default Value: 200h+NodeId
Store Supported: No

Description: This is the Cob-Id of receive PDO #1. Can be changed anytime but cannot be stored in NVRAM.

Notes: This object is not available in EtherCAT fieldbus.

Name: **RX_PDO2_Cob_Id**
Index.Sub: **1401.1H**
Type: Unsigned32
Access: rw
Unit: **--**
Range: 1h÷7FFh,
80000000h (PDO disabled)
Default Value: 300h+NodeId
Store Supported: No

Description: This is the Cob-Id of receive PDO #2. Can be changed anytime but cannot be stored in NVRAM.

Notes: This object is not available in EtherCAT fieldbus.

Name: **RX_PDO3_Cob_Id**
Index.Sub: **1402.1H**
Type: Unsigned32
Access: rw
Unit: **--**
Range: 1h÷7FFh,
80000000h (PDO disabled)
Default Value: 400h+NodeId
Store Supported: No

Description: This is the Cob-Id of receive PDO #3. Can be changed anytime but cannot be stored in NVRAM.

Notes: This object is not available in EtherCAT fieldbus.
This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r67 or superior

---

Name: **RX_PDO4_Cob_Id**
Index.Sub: **1403.1H**
Type: Unsigned32
Access: rw
Unit: **--**
Range: 1h÷7FFh,
80000000h (PDO disabled)
Default Value: 500h+NodeId
Store Supported: No

Description: This is the Cob-Id of receive PDO #4. Can be changed anytime but cannot be stored in NVRAM.

Notes: This object is not available in EtherCAT fieldbus.
This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r67 or superior

---

Name: **RX_PDO_x_Transmission_Type**
Index.Sub: **1400.2H** (RX PDO #1)
1401.2H (RX PDO #2)
1402.2H (RX PDO #3)
1403.2H (RX PDO #4)
Type: Unsigned8
Access: rw
Unit: **--**
Range: 0÷255
Default Value: 1
Store Supported: No

Description: These are the Transmission Type of receive PDOs. Can be changed anytime but cannot be stored in NVRAM. The allowed values are:

Value	*Description*
0÷240	PDO Synchronous. Handled after the reception of each SYNC message.
254/255	PDO Asynchronous. Handled as soon as after the reception.

Notes: This object is not available in EtherCAT fieldbus.

The 1402.2H and 1403.2H are available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r67 or superior.

Name: **TX_PDO1_Cob_Id**
Index.Sub: **1800.1H**
Type: Unsigned32
Access: rw
Unit: **--**
Range: 1h÷7FFh,
80000000h (PDO disabled)
Default Value: 180h+NodeId
Store Supported: No

Description: This is the Cob-Id of transmit PDO #1. Can be changed anytime but cannot be stored in NVRAM.

Notes: This object is not available in EtherCAT fieldbus.

Name: **TX_PDO_x_Transmission_Type**
Index.Sub: **1800.2H** (TX PDO #1)
1801.2H (TX PDO #2)
1802.2H (TX PDO #3)
1803.2H (TX PDO #4)
Type: Unsigned8
Access: rw
Unit: **--**
Range: 0÷255
Default Value: 1
Store Supported: No

Description: These are the Transmission Type of transmit PDOs. Can be changed anytime but cannot be stored in NVRAM. The allowed values are:

Value	*Description*
0	PDO Synchronous and Acyclic. Transmitted after the reception of each SYNC message
1÷240	PDO Synchronous. Transmitted after the reception of n SYNC messages (where n = Value)
254/255	PDO Asynchronous. Transmitted only when it changes (if Event_Timer = 0) or periodically every Event_Timer

Notes: This object is not available in EtherCAT fieldbus.

The 1802.2H and 1803.2H are available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r67 or superior.

Name: **TX_PDO1_Event_Timer**
Index.Sub: **1800.5H**
Type: Unsigned16
Access: rw
Unit: ms
Range: 0÷65535
Default Value: 100
Store Supported: yes

Description: This is the event timer of transmit PDO #1. If value equal to 0 the PDO is transmitted only when it changes. If value > 0 the PDO is transmitted every event timer milliseconds.

Notes: This object is not available in EtherCAT fieldbus.

Name: **TX_PDO2_Cob_Id**
Index.Sub: **1801.1H**
Type: Unsigned32
Access: rw
Unit: **--**
Range: 1h÷7FFh,
80000000h (PDO disabled)
Default Value: 280h+NodeId
Store Supported: No

Description: This is the Cob-Id of transmit PDO #2. Can be changed anytime but cannot be stored in NVRAM.

Notes: This object is not available in EtherCAT fieldbus.

Name: **TX_PDO2_Event_Timer**
Index.Sub: **1801.5H**
Type: Unsigned16
Access: rw
Unit: ms
Range: 0÷65535
Default Value: 100
Store Supported: yes

Description: This is the event timer of transmit PDO #2. If value equal to 0 the PDO is transmitted only when it changes. If value > 0 the PDO is transmitted every event timer milliseconds.

Notes: This object is not available in EtherCAT fieldbus.

Name: **TX_PDO3_Cob_Id**
Index.Sub: **1802.1H**
Type: Unsigned32
Access: rw
Unit: **--**
Range: 1h÷7FFh,
80000000h (PDO disabled)
Default Value: 380h+NodeId
Store Supported: No

Description: This is the Cob-Id of transmit PDO #3. Can be changed anytime but cannot be stored in NVRAM.

Notes: This object is not available in EtherCAT fieldbus.

This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r67 or superior.

Name: **TX_PDO3_Event_Timer**
Index.Sub: **1802.5H**
Type: Unsigned16
Access: rw
Unit: ms
Range: 0÷65535
Default Value: 100
Store Supported: yes

Description: This is the event timer of transmit PDO #3. If value equal to 0 the PDO is transmitted only when it changes. If value > 0 the PDO is transmitted every event timer milliseconds.

Notes: This object is not available in EtherCAT fieldbus.

This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r67 or superior.

Name: **TX_PDO4_Cob_Id**
Index.Sub: **1803.1H**
Type: Unsigned32
Access: rw
Unit: **--**
Range: 1h÷7FFh,
80000000h (PDO disabled)
Default Value: 480h+NodeId
Store Supported: No

Description: This is the Cob-Id of transmit PDO #4. Can be changed anytime but cannot be stored in NVRAM.

Notes: This object is not available in EtherCAT fieldbus.

This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r67 or superior.

Name: **TX_PDO4_Event_Timer**
Index.Sub: **1803.5H**
Type: Unsigned16
Access: rw
Unit: ms
Range: 0÷65535
Default Value: 100
Store Supported: yes

Description: This is the event timer of transmit PDO #4. If value equal to 0 the PDO is transmitted only when it changes. If value > 0 the PDO is transmitted every event timer milliseconds.

Notes: This object is not available in EtherCAT fieldbus.
This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r67 or superior.

Name: **Dips**
Index.Sub: **2004.0H**
Type: Unsigned16
Access: ro
Unit: --
Range: 0 ÷ 4095 (0FFFH)
Default Value: --
Store Supported: No

Description: This object contains the current status of drive's dip switches

Notes: This object is not available in EtherCAT fieldbus.

Name: **Min_Current**
Index.Sub: **2005.1H**
Type: Unsigned16
Access: rw
Unit: mA
Range: 0 ÷ (max drive current)
Default Value: 0
Store Supported: Yes

Description: It sets the motor's reduced current. The drive automatically reduces the current after a current reduction time at the end of the movement

Notes: The current set is the Irms current. The peak current is Irms * 1.4.

Name: **Max_Current**
Index.Sub: **2005.2H**
Type: Unsigned16
Access: rw
Unit: mA
Range: 0 ÷ (max drive current)
Default Value: 0
Store Supported: Yes

Description: It sets the motor's current when running at constant speed. During acceleration and deceleration, the *Boost_Current* is automatically set.

Notes: The current set is the Irms current. The peak current is Irms * 1.4.

Name: **Boost_Current**
Index.Sub: **2005.3H**
Type: Unsigned16
Access: rw
Unit: mA
Range: 0 ÷ (max drive current)
Default Value: 0
Store Supported: Yes

Description: It sets the boost current in the motor. The boost current is enabled when the motor accelerates and decelerates.

Notes: The current set is the Irms current. The peak current is Irms * 1.4.

Name: **Nominal_Current**
Index.Sub: **2005.4H**
Type: Unsigned16
Access: rw
Unit: mA
Range: 0 ÷ (max drive current)
Default Value: 65535
Store Supported: Yes

Description: This object defines the Nominal current (rms) of the motor. This value is used for motor R,L detection procedure.

Notes: This object is available only for 'CSMD1,SB4,SW4,SM4,SW5 Drives'

Name: **Current_Gain_Kp**
Index.Sub: **2005.6H**
Type: Float
Access: rw
Unit: **%**
Range: 0.01 ÷ 100.0
Default Value: 50.0
Store Supported: Yes

Description: This object defines the proportional gain of the PI controller of the current regulation.

Notes: **This object is available only on SW1 drive type 9x60 (DMD02).**

Name: **Motor_R**
Index.Sub: **2005.6H**
Type: Unsigned32
Access: rw
Unit: ohm $x10^{-3}$
Range: Unsigned32
Default Value: 380
Store Supported: Yes

Description: This object sets the motor R (resistance). This value is used when bit9 of 'Drive_working_Settings_extended' object is equal to 0.

Notes: This object is available only for 'CSMD1,SB4,SW4,SM4,SW5 Drives'

Name: **Motor_L**
Index.Sub: **2005.7H**
Type: Unsigned32
Access: rw
Unit: H $x10^{-6}$
Range: Unsigned32
Default Value: 2420
Store Supported: Yes

Description: This object sets the motor L (inductance). This value is used when bit9 of 'Drive_working_Settings_extended' object is equal to 0.

Notes: This object is available only for 'CSMD1,SB4,SW4,SM4,SW5 Drives'

Name: **Current_Gain_Ki**
Index.Sub: **2005.7H**
Type: Float
Access: rw
Unit: **%**
Range: 0.01 ÷ 100.0
Default Value: 25.0
Store Supported: Yes

Description: This object defines the integral gain of the PI controller of the current regulation.

Notes: **This object is available only on SW1 drive type 9x60 (DMD02).**

Name: **Current_Actual_Value**
Index.Sub: **2005.8H**
Type: Unsigned16
Access: ro
Unit: mA
Range: 0 ÷ (max drive current)
Default Value: 0
Store Supported: No

Description: This object contains the current value of the current supplied to the motor (Irms).

Notes: This object is available only for SW1,SM1A,DMD105 Drives.

Name: **Motor_R_detected**
Index.Sub: **2005.8H**
Type: Unsigned32
Access: rw
Unit: ohm $x10^{-3}$
Range: Unsigned32
Default Value: **--**
Store Supported: No

Description: This object returns the motor R (resistance) when bit9 of 'Drive_working_Settings_extended' object is equal to 1.

Notes: This object is available only for 'CSMD1,SB4,SW4,SM4,SW5 Drives'

Name: **Motor_L_detected**
Index.Sub: **2005.9H**
Type: Unsigned32
Access: rw
Unit: H $x10^{-6}$
Range: Unsigned32
Default Value: --
Store Supported: No

Description: This object returns the motor L (inductance) when bit9 of 'Drive_working_Settings_extended' object is equal to 1.

Notes: This object is available only for 'CSMD1,SB4,SW4,SM4,SW5 Drives'

Name: **Encoder_Actual_Value[0÷1]**
Index.Sub: **2007.0H,2008.0H**
Type: Integer32
Access: rw
Unit: Increments
Range: -2147483648 ÷ 2147483647
Default Value: 0
Store Supported: No

Description: This object contains the encoder #0 and #1 current position. The position can be cleared writing a 0 in the object.

Notes: The number of encoders available depends on the version of the drive currently in use.

Name: **Drive_Voltage_Actual_Value**
Index.Sub: **2009.0H**
Type: Unsigned16
Access: ro
Unit: Volts
Range: Minimum Drive Voltage ÷ Maximum Drive Voltage
Default Value: --
Store Supported: No

Description: This object contains the current voltage that powers the drive.

Notes:

Name: **Drive_Temperature_Actual_Value**
Index.Sub: **200A.0H**
Type: Unsigned16
Access: ro
Unit: °C
Range: 0 ÷ Maximum Drive Temperature
Default Value: --
Store Supported: No

Description: This object contains the current drive’s temperature.

Notes:

Name: **Min_Profile_Velocity**
Index.Sub: **2010.0H**
Type: Unsigned16
Access: rw
Unit[1] : *Inc/s*
Unit[2] : User-defined velocity unit [3]
Range[1] : 250 Inc/s ÷ 3000 Inc/s
Range[2] : 0 ÷ '*Max*' [7]
Default Value: 250
Store Supported: No

Description: This objects defines the minimum motor velocity for the acceleration and deceleration ramp.

Notes:

[1] *for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower'*

[2] *for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior'*

[3] *See 'Factor Group' chapter*

[4] This object can only be set with the motor at a standstill and cannot be higher than the *Max_Profile_Velocity* object.

[5] If the value to be set is lower than the minimum value of range, the minimum value is stored.

[6] If the value to be set is higher than the maximum value of range, the maximum value is stored.

[7] '*Max*'[2] value is calculated according to this formula :

*Max(User-defined velocity unit) = 150000 * (2013.2h / 2013.1h) * (60EF.0h / 65536)*

See 'Factor Group' chapter for more details.

[8] The 'device internal velocity unit' is 'Increments/sec'

Name: **Motor_Parameters (Highest sub-index supported)**
Index.Sub: **2012.0H**
Type: Unsigned8
Access: ro
Unit: --
Range: 2
Default Value: 2
Store Supported: No

Description: Highest sub-index supported

Notes:

Name: **Motor_Parameters (Motor_Step_Angle)**
Index.Sub: **2012.1H**
Type: Unsigned16
Access: rw
Unit: --
Range: 1; 2; 4; 8; 16; 32; 64; 128; 65535 *(3)*
Default Value: 1
Store Supported: Yes

Description: This object is used to calculate the motor resolution.

- If 'Motor_Step_Angle' value is different from 65535, the number of motor Increments per revolution are computed as follow :

  *Motor_Resolution = (Motor_Poles*4) * Motor_Step_Angle;*

  If 'Motor_Poles' = 50 (standard typical value) :

Motor_Step_Angle = 1	→	Motor_Resolution = 200	Inc/rev
Motor_Step_Angle = 2	→	Motor_Resolution = 400	Inc/rev
Motor_Step_Angle = 4	→	Motor_Resolution = 800	Inc/rev
Motor_Step_Angle = 8	→	Motor_Resolution = 1600	Inc/rev
Motor_Step_Angle = 16	→	Motor_Resolution = 3200	Inc/rev
Motor_Step_Angle = 32	→	Motor_Resolution = 6400	Inc/rev
Motor_Step_Angle = 64	→	Motor_Resolution = 12800	Inc/rev
Motor_Step_Angle = 128	→	Motor_Resolution = 25600	Inc/rev

- If 'Motor_Step_Angle' value is equal to 65535 *(3)*, the number of motor Increments per revolution is :

Motor_Step_Angle = 65535 *(3)*	→	Motor_Resolution = 65536	Inc/rev

Notes: *(1)* This object can only be set with the motor at a standstill.
(2) If the value set is not valid the default value is stored.
(3) only for CSMD1,SB4,SW4,SM4,SW5 Drives
For Titanio Line Drives it is suggested to set this object = 65535 to have Position, Speed and Acceleration based on the same internal unit.

Name: **Motor_Parameters (Motor_Poles)**
Index.Sub: **2012.2H**
Type: Unsigned16
Access: rw
Unit: # of motor poles
Range: 6; 8; 10; 12; 15; 25; 45; 50; 100
Default Value: 50 (standard typical value)
Store Supported: Yes

Description: It sets the number of motor poles. The number of poles can be computed as follow:
Take the Step Angle value from motor data sheet (typically 1.8°) and then perform the following computation:

$$\frac{360°}{StepAngle\ (motor\ data\ sheet)} * \frac{1}{4}$$

For a motor with Step Angle (motor data sheet) = 1.8° the number of poles are 50.

Notes: This object can only be set with the motor at a standstill.
If the value set is not valid the default value is stored.

Name: **Motor_Factor (Highest sub-index supported)**
Index.Sub: **2013.0H**
Type: Unsigned8
Access: ro
Unit: --
Range: 4
Default Value: 4
Store Supported: No

Description: Highest sub-index supported

Notes: (1) This object is available only for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior'.

(2) See 'Factor Group' chapter for more details.

Name: **Motor_Factor (Velocity_Factor_Numerator)**
Index.Sub: **2013.1H**
Type: Unsigned32
Access: rw
Unit: --
Range: 1÷4294967295
Default Value: 1
Store Supported: Yes

Description: This object defines the 'Numerator' of the ratio for Velocity factor used to convert User-defined velocity unit to Internal velocity unit and vice versa.

$$Velocity\ factor = \frac{2013.1h}{2013.2h}$$

The Internal velocity unit is Hz (Inc/s).

According to 'Sensor_Selection_Code' object setting, the 'Inc/s' depend on 'Motor_Step_Angle' or 'Position_Encoder_Resolution' object.

Notes: (1) This object is available only for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior'.

(2) See 'Factor Group' chapter for more details.

Name: **Motor_Factor (Velocity_Factor_Denominator)**
Index.Sub: **2013.2H**
Type: Unsigned32
Access: rw
Unit: **--**
Range: 1÷4294967295
Default Value: 1
Store Supported: Yes

Description: This object defines the 'Denominator' of the ratio for Velocity factor used to convert User-defined velocity unit to Internal velocity unit and vice versa.

$$Velocity\ factor = \frac{2013.1h}{2013.2h}$$

The Internal velocity unit is Hz (Inc/s).

According to 'Sensor_Selection_Code' object setting, the 'Inc/s' depend on 'Motor_Step_Angle' or 'Position_Encoder_Resolution' object.

Notes: (1) This object is available only for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior'.

(2) See 'Factor Group' chapter for more details.

Name: **Motor_Factor (Acceleration_Factor_Numerator)**
Index.Sub: **2013.3H**
Type: Unsigned32
Access: rw
Unit: --
Range: 1÷4294967295
Default Value: 1
Store Supported: Yes

Description: This object defines the 'Numerator' of the ratio for Acceleration factor used to convert user-defined Acceleration (and Deceleration) unit to Internal Acceleration (and Deceleration) unit and vice versa.

$$Acceleration\,factor = \frac{2013.3\,h}{2013.4\,h}$$

The Internal Acceleration unit is 'Inc/s²'.

According to 'Sensor_Selection_Code' object setting, the 'Inc/s²' depend on 'Motor_Step_Angle' or 'Position_Encoder_Resolution' object.

Notes: (1) This object is available only for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior'

(2) See 'Factor Group' chapter for more details.

Name: **Motor_Factor (Acceleration_Factor_Denominator)**
Index.Sub: **2013.4H**
Type: Unsigned32
Access: rw
Unit: **--**
Range: 1÷4294967295
Default Value: 1
Store Supported: Yes

Description: This object defines the 'Denominator' of the ratio for Acceleration factor used to convert user-defined Acceleration (and Deceleration) unit to Internal Acceleration (and Deceleration) unit and vice versa.

$$Acceleration\,factor = \frac{2013.3\,h}{2013.4\,h}$$

The Internal Acceleration unit is 'Inc/s²'

According to 'Sensor_Selection_Code' object setting, the 'Inc/s²' depend on 'Motor_Step_Angle' or 'Position_Encoder_Resolution' object.

Notes: (1) This object is available only for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior'.

(2) See 'Factor Group' chapter for more details.

Name: **RotoSwitches**
Index.Sub: **2027.0H**
Type: Unsigned16
Access: ro
Unit: --
Range: 0 ÷ 65535 (FFFFH)
Default Value: --
Store Supported: No

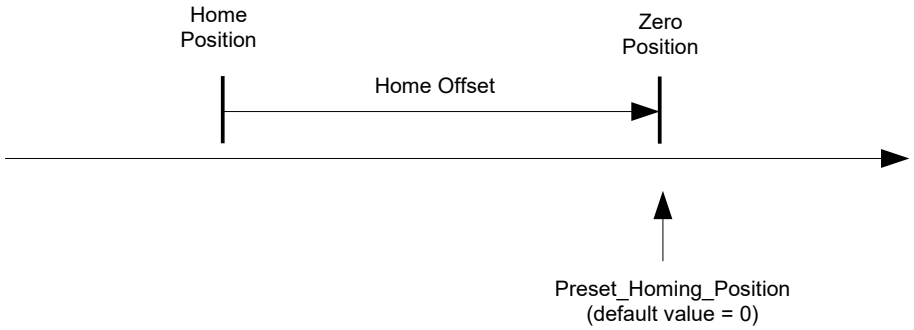
Description: This object contains the current status of drive's Rotoswitches

ROTOSWITCHES															
bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
SW4 Rotoswitch Position				**SW3 Rotoswitch Position**				**SW2 Rotoswitch Position**				**SW1 Rotoswitch Position**			
bit3	bit2	bit1	bit0	bit3	bit2	bit1	bit0	bit3	bit2	bit1	bit0	bit3	bit2	bit1	bit0

Notes: **This object is available only on SW1 drive type 9x60 (DMD02).**
This object is not available in EtherCAT fieldbus.

Name: **Preset_Homing_Position**
Index.Sub: **2080.0H**
Type: Integer32
Access: rw
Unit[1] : Increments
Unit[2] : User-defined position unit [3]
Range: Integer32
Default Value: 0
Store Supported: Yes [4]

Description: This object is used during the Homing procedure and contains the desired position value for the Homing Zero Position. All position values (objects 6062h,6063h,6064h) are set to *Preset_Homing_Position* value.



Notes: *[1] for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower'*

[2] for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior'

[3] See 'Factor Group' chapter

[4] 'Store supported' is available for this object *only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r36 or superior.*

Name: **Drive_Homing_Inputs_Setting**
Index.Sub: **2081.0H**
Type: Unsigned32
Access: rw
Unit: --
Range: 0 ÷ FFFFFFFFH
Default Value: 00000000H
Store Supported: Yes [4]

Description: This object is used to parametrize drive digital inputs bank # 0 allocation for Homing procedure. For any function can be assigned one digital input that have to be specified in the four bits of the object concerning that function.

	Bit 31÷28	Bit 27÷24	Bit 23÷20	Bit 19÷16	Bit 15÷12	Bit 11÷8	Bit 7÷4	Bit 0÷3
					Positive Limit Switch	Negative Limit Switch	Index Pulse	Home Switch
Range[1]	**0÷F**	**0÷F**	**0÷F**	**0÷F**	**0,1,4,5**	**0,1,4,5**	**0,1,4,5**	**0,1,4,5**
Range[2]	**0÷F**	**0÷F**	**0÷F**	**0÷F**	**0,1,2,3,5,6,7**	**0,1,2,3,5,6,7**	**0,1,2,3,5,6,7**	**0,1,2,3,5,6,7**
Default Value					0	0	0	0

Notes: [1] for SW1,SM1A,DMD105 Drives
[2] for CSMD1,SB4,SW4,SM4,SW5 Drives

[3] See also *D Appendix Multiplexed I/O Allocations*

[4] 'Store supported' is available for this object *only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r36 or superior.*

Name: Touch_Probe1_Filter
Index.Sub: 2082.0H
Type: Unsigned32
Access: rw
Unit: microseconds
Range: 0 ÷ FFFFFFFFH
Default Value: 00080008H
Store Supported: Yes [2]

Description: This object is used to set the filters of Touch Probe1.

	Touch Probe 1 Filter	
	(Digital Input High Level)	**(Digital Input Low Level)**
	Bit 31÷16	**Bit 15÷0**
Range	**0÷65535**	**0÷65535**
Default Value	8	8

Notes: [1] This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r11 or superior.

[2] 'Store supported' is available for this object *only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r36 or superior.*

Name: Touch_Probe2_Filter
Index.Sub: 2083.0H
Type: Unsigned32
Access: rw
Unit: microseconds
Range: 0 ÷ FFFFFFFFH
Default Value: 00080008H
Store Supported: Yes [2]

Description: This object is used to set the filters of Touch Probe2.

	Touch Probe 2 Filter	
	(Digital Input High Level)	**(Digital Input Low Level)**
	Bit 31÷16	**Bit 15÷0**
Range	**0÷65535**	**0÷65535**
Default Value	8	8

Notes: [1] This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r11 or superior.

[2] 'Store supported' is available for this object *only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r36 or superior.*

Name: DS402_Working_Settings
Index.Sub: 2084.0H
Type: Unsigned32
Access: rw
Unit: --
Range: 0 ÷ FFFFFFFFH
Default Value: 0
Store Supported: Yes (2)

Description: This object is used to parameterize DS402 working general configuration.

Bit #	Name	Description	Default Value
0	PP_mode_Position_Align	0 = Not align Position in 'PP mode' 1 = Align Position in 'PP mode' 2 = Not used 3 = Not used	0
1			
2	Torque_Enable_Mode	0 = STO/Torque Enable handled as Fault 1 = STO/Torque Enable not handled as Fault	0
3	Limit_Switch_Mode	0 = Fault on Limit Switch 1 = No Fault on Limit Switch	0
4	Digital_Input_Actual_Value	0 = Filtered and polarized Digital Inputs on bit 16÷31 of 60FD Object 1 = Direct value of Digital Inputs on bit 16÷31 of 60FD Object	0
5	Reserved	Reserved	0
6	Reserved	Reserved	0
7	Reserved	Reserved	0
8	Reserved	Reserved	0
9	Reserved	Reserved	0
10	Reserved	Reserved	0
11	Reserved	Reserved	0
12	Reserved	Reserved	0
13	Reserved	Reserved	0
14	Reserved	Reserved	0
15	Reserved	Reserved	0
16	Reserved	Reserved	0
17	Reserved	Reserved	0
18	Reserved	Reserved	0
19	Reserved	Reserved	0
20	Reserved	Reserved	0
21	Reserved	Reserved	0
22	Reserved	Reserved	0
23	Reserved	Reserved	0
24	Reserved	Reserved	0
25	Reserved	Reserved	0
26	Reserved	Reserved	0
27	Reserved	Reserved	0
28	Reserved	Reserved	0
29	Reserved	Reserved	0
30	Reserved	Reserved	0
31	Reserved	Reserved	0

Bit Explanation:

PP_mode_Position_Align (bit1,bit0):

This setting is used only for 'Profile Position mode' and it is relevant when 'Sensor_Selection_Code' object is equal to (0) or (-2).

When the value is (1), with motor at a standstill and before to start a movement, the 'Position demand value' is aligned to 'Position actual value'.

When the value is (0), the Position align function is not active.

Torque_Enable_Mode:

This setting is used only for drives equipped with STO or Torque Enable input

When the value is (1), the intervention of STO/Torque Enable input is not considered a Fault and the state machine will be kept to 'Switch On Disabled' until the input is off.

When the value is (0), the intervention of STO/Torque Enable input is considered a Fault.

Limit_Switch_Mode:

When the value is (1), the intervention of a Limit Switch input is not considered a Fault

When the value is (0), the intervention of a Limit Switch input is considered a Fault.

Digital_Input_Actual_Value:

When the value is (1), the status of B0_Digital_Inputs on bits 16÷31 of 60FDh object is direct, not filtered and not polarized.

When the value is (0), the status of B0_Digital_Inputs on bits 16÷31 of 60FDh object is filtered and polarized.

Notes:

(1) This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r28 or superior.

(2) 'Store supported' is available for this object *only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V02r29 or superior.*

Name: **Drive_Inputs_Level**
Index.Sub: **2200.1H**
Type: Unsigned16
Access: rw
Unit: --
Range: 0 ÷ FFFFH
Default Value: **----**
Store Supported: Yes

Description: This object is used to parametrize drive digital inputs working level.

Bit #	Name	Description	Default Value
0	Forward_Limit_Switch_Level	1 = Active high level 0 = Active low level	0
1	Backward_Limit_Switch_Level	1 = Active high level 0 = Active low level	0
2	Fast_Stop_Level	1 = Active high level 0 = Active low level	0
3	Reserved	Reserved	--
4	Reserved	Reserved	--
5	Reserved	Reserved	--
6	Reserved	Reserved	--
7	Reserved	Reserved	--
8	Reserved	Reserved	--
9	Reserved	Reserved	--
10	Reserved	Reserved	--
11	Reserved	Reserved	--
12	Reserved	Reserved	--
13	Reserved	Reserved	--
14	Reserved	Reserved	--
15	Reserved	Reserved	--

Bit Explanation:

Forward_Limit_Switch_Level: When this bit is set (1) the drive will consider the forward limit switch intervention when the corresponding digital input (see *Drive_Inputs_Setting* object) will close. When this bit is reset (0) the drive will consider the forward limit switch intervention when the corresponding digital input (see *Drive_Inputs_Settings* object) will open.

Backward_Limit_Switch_Level: When this bit is set (1) the drive will consider the backward limit switch intervention when the corresponding digital input (see *Drive_Inputs_Setting* object) will close. When this bit is reset (0) the drive will consider the backward limit switch intervention when the corresponding digital input (see *Drive_Inputs_Setting* object) will open.

Fast_Stop_Level: When this bit is set (1) the drive will consider the fast stop input intervention when the corresponding digital input (see *Drive_Inputs_Setting* object) will close. When this bit is reset (0) the drive will consider the fast stop input intervention when the corresponding digital input (see *Drive_Inputs_Setting* object) will open.

Notes: See also *Drive_Working_Settings* object.

Name: **Drive_Working_Settings**
Index.Sub: **2200.2H**
Type: Unsigned16
Access: rw
Unit: --
Range: 0 ÷ FFFFH
Default Value: **--**
Store Supported: Yes

Description: This object is used to parametrize drive working modalities.

Bit #	Name	Description	Default Value
0	**Forward_Limit_Switch_Check**	1 = Forward limit switch check enabled 0 = Forward limit switch check disabled	0
1	**Backward_Limit_Switch_Check**	1 = Backward limit switch check enabled 0 = Backward limit switch check disabled	0
2	**Fast_Stop_From_Input** (2)	1 = Fast Stop Input enabled 0 = Fast Stop Input disabled	0
3	Reserved	Reserved	0
4	**Feedback_Motor_Check** (1)	1 = Motor feedback enabled 0 = Motor feedback disabled	0
5	Reserved	Reserved	--
6	Reserved	Reserved	--
7	**Disable_Digital_Outputs_FW_Handling**	1 = Disable Digital_Outputs handling by firmware (All DO user free) 0 = Enable Digital_Outputs handling by firmware (Not all DO user free)	0
8	Reserved	Reserved	--
9	Reserved	Reserved	--
10	**Motor_Rotation_Direction**	1 = counter clockwise rotation when motor move forward 0 = clockwise rotation when motor move forward	0
11	Reserved	Reserved	--
12	Reserved	Reserved	--
13	**Feedback_Error_Motor_Action** (1)	1 = Stop Motor when feedback error 0 = Don't stop the motor when feedback error	0
14	Reserved	Reserved	--
15	Reserved	Reserved	--

Bit Explanation:

Forward_Limit_Switch_Check:

When this bit is set (1), outside of the Homing modality, the drive will check continuously for the intervention of forward limit switch (see *Drive_Inputs_Setting* and *Drive_Inputs_Level* objects). If the limit switch intervenes and the motor is running forward :
- the motor will stop with the action indicated in the *Fault_reaction_option_code* object.
- the *Internal_limit_active* bit and *Fault* bit of *Statusword* object will be set immediately
- an emergency message will be send

When this bit is reset (0) the drive will not check for the forward limit switch at all.

Backward_Limit_Switch_Check:

When this bit is set (1), outside of the homing modality, the drive will check continuously for the intervention of backward limit switch (see *Drive_Inputs_Setting* and *Drive_Inputs_Level* objects). If the limit switch intervenes and the motor is running backward :
- the motor will stop with the action indicated in the *Fault_reaction_option_code* object.
- the *Internal_limit_active* bit and *Fault* bit of *Statusword* object will be set immediately
- an emergency message will be send

When this bit is reset (0) the drive will not check for the backward limit switch at all.

Fast_Stop_From_Input[2]:

Fast_Stop_From_Input: When this bit is set (1) the drive will check continuously for the intervention of fast stop input (see Drive_Inputs_Setting and Drive_Inputs_Level objects). If the fast stop is on the state machine is kept to 'Switch On Disabled' until the input goes off.

When this bit is reset (0) the drive will not check for the fast stop input at all.

Disable_Digital_Outputs_FW_Handling:

When this bit is set (1) the firmware will not set/reset automatically the digital outputs assigned to Fault and Ready/Busy functions. Whole digital outputs are available to the user.

When this bit is reset (0) the firmware will set/reset automatically the digital outputs assigned to Fault and Ready/Busy functions. The remaining digital outputs are available to the user.

Motor_Rotation_Direction:

When this bit is set (1) the motor will rotate counter clockwise when the motion direction is forward (*Position_Actual_Value* object increase) and clockwise when the motion direction is backward (*Position_Actual_Value* object decrease).

When this bit is reset (0) the drive motor will rotate clockwise when the motion direction is forward (*Position_Actual_Value* object increase) and counter clockwise when the motion direction is backward (*Position_Actual_Value* object decrease).

Feedback_Motor_Check[1]:

When this bit is set (1) the drive will enable the feedback feature to close the motion control loop. This will optimize the motor efficiency.

When this bit is reset (0) the feedback feature is disabled.

Feedback_Error_Motor_Action[1]:

This bit setting is relevant if feedback feature is enabled (Feedback_Motor_Check = 1).

When this bit is set (1) the drive will stop any movement in progress if a feedback error is detected.

When this bit is reset (0) the drive will not take any action, if a feedback error is detected.

Notes:

[1] available only for CSMD1,SB4,SW4,SM4,SW5 Drives.
[2] available only for CSMD1,SB4,SW4,SM4,SW5 Drives starting from firmware release V02r63.

Name: **Drive_Inputs_Setting**
Index.Sub: **2200.3H**
Type: Unsigned32
Access: rw
Unit: --
Range: 0 ÷ FFFFFFFFH
Default Value: **----**
Store Supported: Yes

Description: This object is used to parametrize drive digital inputs bank # 0 allocation. For any function can be assigned one digital input that have to be specified in the four bits of the object concerning that function.

See also *Drive_Working_Settings* object.

	Bit 31÷28	Bit 27÷24	Bit 23÷20	Bit 19÷16	Bit 15÷12	Bit 11÷8	Bit 7÷4	Bit 3÷0
						Fast Stop	Backward Limit Switch	Forward Limit Switch
Range						**0÷7**	**0÷7**	**0÷7**
Default Value	0	0	0	0	0	2	1	0

Notes: See also *D Appendix Multiplexed I/O Allocations*

Name: **Drive_CANopen_Config**
Index.Sub: **2200.5H**
Type: Unsigned16
Access: rw
Unit: --
Range: 0000H ÷ FFFFH

	Bit 15÷12	Bit 11÷8	Bit 7÷4	Bit 0÷3
	RX PDO1 Config	TX PDO1 Config	RX PDO2 Config	TX PDO2 Config
Default Value	0	0	4	1

TX PDOx Config:

00 → Mapping #0
01 → Mapping #1
02 → Mapping #2
03 → Mapping #3
……

RX PDOx Config:

00 → Mapping #0
01 → Mapping #1
02 → Mapping #2
03 → Mapping #3
…….

Default Value: 0041H
Store Supported: Yes

Description: This object permits to change the mapping of RX/TX PDO. See §2.5 for details.

Notes: This object is not available in EtherCAT fieldbus.

'RX PDO1 Config' and 'TX PDO1 Config' are available only for SW1,DMD105,SM1A Drives with firmware V02r18 or superior , and only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r67 or superior.

Name: **Analog_Input0_K_Filter**
Index.Sub: **2200.6H**
Type: Unsigned16
Access: rw
Unit: **--**
Range: 0 ÷ 10000
Default Value: --
Store Supported: Yes

Description: It contains the value of the K constant used for filtering the analog input #0. The formula for filtering the analog inputs is the following:

$$Analog\_Input\_Filtered_{(n)} = Analog\_Input\_Filtered_{(n-1)} + \frac{(Analog\_Input_{(n)} - Analog\_Input\_Filtered_{(n-1)})}{Analog\_Inputs\_K\_Filter}$$

High value of K lead to a more filtered analog input value. With K = 1 the filter is disabled and the Analog_In[0] objects returns the instant value of analog input #0 without any filtering. With K = 0 the analog input #0 is disabled.

Notes:

Name: **B0_Digital_Inputs_Polarity**
Index.Sub: **2200.7H**
Type: Unsigned16
Access: rw
Unit: **--**
Range: 0 (all inputs have normal polarity) ÷ 2^n-1 (all inputs have inverted polarity, where n is the number of digital inputs available)
Default Value: 0

Bit #	Input	Description	Default Value
0	B0_In0	1 = inverted polarity 0 = normal polarity	0
1	B0_In1	1 = inverted polarity 0 = normal polarity	0
2	B0_In2	1 = inverted polarity 0 = normal polarity	0
3	B0_In3	1 = inverted polarity 0 = normal polarity	0
n	B0_Inn	1 = inverted polarity 0 = normal polarity	0

Store Supported: Yes

Description: It contains the polarity of all the digital inputs on bank 0 of the drive. When the polarity is normal, the input status (B0_Digital_Inputs) is 1 if the voltage is supplied to the input (5V or 24V) and 0 if no voltage is supplied to the input. When the polarity is inverted, the input status (B0_Digital_Inputs) is 0 if the voltage is supplied to the input (5V or 24V) and 1 if no voltage is supplied to the input.

Notes:

Name: **Analog_Input0_Type**
Index.Sub: **2200.EH**
Type: Unsigned16
Access: ro
Unit: --
Range: 0 ; 1

Value	Description
0	Differential ±10V (JMP600 – position 1)
1	Potentiometer (JMP600 – position 2)

Default Value: --
Store Supported: No

Description: This object contains Analog Input0 Type (see JMP600 on hardware manual).

Notes: Only for boards DMD00 (SW1x4080) and DMD01 (SW1D2x42).

Name: **Analog_Input1_Type**
Index.Sub: **2200.FH**
Type: Unsigned16
Access: ro
Unit: --
Range: 0 ; 1

Value	Description
0	Differential ±10V (JMP600 – position 1)
1	Potentiometer (JMP600 – position 2)

Default Value: --
Store Supported: No

Description: This object contains Analog Input1 Type (see JMP600 on hardware manual).

Notes: Only for boards DMD00 (SW1x4080) and DMD01 (SW1D2x42).

Name: **Analog_Input1_K_Filter**
Index.Sub: **2200.10H**
Type: Unsigned16
Access: rw
Unit: **--**
Range: 0 ÷ 10000
Default Value: --
Store Supported: Yes

Description: It contains the value of the K constant used for filtering the analog input #1. The formula for filtering the analog inputs is the following:

$$Analog\_Input\_Filtered_{(n)} = Analog\_Input\_Filtered_{(n-1)} + \frac{(Analog\_Input_{(n)} - Analog\_Input\_Filtered_{(n-1)})}{Analog\_Inputs\_K\_Filter}$$

High value of K lead to a more filtered analog input value. With K = 1 the filter is disabled and the Analog_In[1] objects returns the instant value of analog input #1 without any filtering. With K = 0 the analog input #1 is disabled.

Notes:

Name: **Drive_Working_Settings_Extended**
Index.Sub: **2200.11H**
Type: WORD
Access: rw
Unit: --
Range: 0 ÷ FFFFH
Default Value: 0H
Store Supported: Yes

Description: This register is used to parametrize drive working modalities.

Bit #	Name	Description	Default Value
0	Reserved	Reserved	--
1	Reserved	Reserved	--
2	Reserved	Reserved	--
3	Reserved	Reserved	--
4	Reserved	Reserved	--
5	Reserved	Reserved	--
6	**Encoder0_Rotation_Direction**	**1 = The encoder #0 counting is reversed** **0 = The encoder #0 counting is not reversed**	**0**
7	**Encoder1_Rotation_Direction**	**1 = The encoder #1 counting is reversed** **0 = The encoder #1 counting is not reversed**	**0**
8	Reserved	Reserved	--
9	**Motor R,L detection** [1]	**1 = The motor R,L parameters are automatically detected** **0 = The motor R,L parameters are defined by objects**	1
10	Reserved	Reserved	--
11	Reserved	Reserved	--
12	Reserved	Reserved	--
13	Reserved	Reserved	--
14	Reserved	Reserved	--
15	Reserved	Reserved	--

Bit Explanation:

EncoderX_Rotation_Direction:

When this bit is set (1) the Encoder #X counting is reversed.

When this bit is reset (0) the Encoder #X counting is not reversed.

Motor R,L detection[1]:

When this bit is set (1) the motor R,L parameters are detected by automatic calbration. For this procedure the object 2005.4h must be set with Nominal Current of the motor.

When this bit is reset (0) the motor R,L parameters are defined by mean of objects 2005.6h and 2005.7h.

Notes:

[1] available only for CSMD1,SB4,SW4,SM4,SW5 Drives.

Name: **B1_Digital_Inputs_Polarity**
Index.Sub: **2200.16H**
Type: Unsigned16
Access: rw
Unit: **--**
Range: 0 (all inputs have normal polarity) ÷ 2^n-1 (all inputs have inverted polarity, where n is the number of digital inputs available)
Default Value: 0

Bit #	Input	Description	Default Value
0	B1_In0	1 = inverted polarity 0 = normal polarity	0
1	B1_In1	1 = inverted polarity 0 = normal polarity	0
2	B1_In2	1 = inverted polarity 0 = normal polarity	0
3	B1_In3	1 = inverted polarity 0 = normal polarity	0
n	B1_Inn	1 = inverted polarity 0 = normal polarity	0

Store Supported: Yes

Description: It contains the polarity of all the digital inputs on bank 1 of the drive. When the polarity is normal, the input status (B1_Digital_Inputs) is 1 if the voltage is supplied to the input (5V or 24V) and 0 if no voltage is supplied to the input. When the polarity is inverted, the input status (B1_Digital_Inputs) is 0 if the voltage is supplied to the input (5V or 24V) and 1 if no voltage is supplied to the input.

Notes:

Name: **Drive_CANopen_Config_Extended**
Index.Sub: **2200.1AH**
Type: Unsigned16
Access: rw
Unit: --
Range: 0000H ÷ FFFFH

	Bit 15÷12	Bit 11÷8	Bit 7÷4	Bit 0÷3
	RX PDO3 Config	TX PDO3 Config	RX PDO4 Config	TX PDO4 Config
Default Value	0	0	0	0

TX PDOx Config:

00 → Mapping #0
01 → Mapping #1
02 → Mapping #2
03 → Mapping #3
……

RX PDOx Config:

00 → Mapping #0
01 → Mapping #1
02 → Mapping #2
03 → Mapping #3
…….

Default Value: 0000H
Store Supported: Yes

Description: This object permits to change the mapping of RX/TX PDO. See §2.5 for details.

Notes: This object is not available in EtherCAT fieldbus.

This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r70 or superior.

Name: **Counters_Filter**
Index.Sub: **2210.3H**
Type: Unsigned16
Access: rw
Unit: --
Range: 0 ÷ FF H

	Bit 15÷12	Bit 7÷4	Bit 0÷3
	Not used	Counter #1 Filter	Counter #0 Filter
Range	//	0÷F	0÷F
Default Value	00	0	0

Value	Filter (us)
0	1,6
1	1,9
2	2,2
3	2,6
4	3,2
5	5,1
6	6,0
7	7,2
8	8,2
9	10,0
A	15,0
B	20,0
C	25,0
D	35,0
E	50,0
F	80,0

Default Value: 00H
Store Supported: Yes

Description: This object is used to set the filters of hardware counter #0 (B0_In0,B0_In1) and counter #1 (B0_In2,B0_In3). See appendix B.

Notes: This object is available only for SW1,SM1A,DMD105 Drives

Name: **Encoder_Frequency[0÷1]**
Index.Sub: **2211.1H,2211.2H**
Type: Unsigned32
Access: ro
Unit: Inc/s
Range: **- -**
Default Value: 0
Store Supported: No

Description: This object contains the encoder # 0 and #1 frequency.

Notes: The number of encoders available depends on the version of the drive currently in use.

Name: **Feedback_Max_Displacement**
Index.Sub: **2230.1H**
Type: Unsigned32
Access: rw
Unit: Inc
Range: Unsigned32
Default Value: 10000
Store Supported: Yes

Description: This object contains the maximum allowed displacement between motor and encoder for the feedback feature. The value is expressed in Increments, referred to the motor resolution of 65536 inc/rev.

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives

---

Name: **Feedback_Actual_Displacement**
Index.Sub: **2230.2H**
Type: Integer32
Access: ro
Unit: Inc
Range: Integer32
Default Value: --
Store Supported: No

Description: This object stores the actual displacement between motor and encoder used for feedback feature expressed in Increments, referred to the motor resolution of 65536 inc/rev.

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives

---

Name: **Feedback_Source_PPR**
Index.Sub: **2230.3H**
Type: Unsigned32
Access: rw
Unit: Inc
Range: Unsigned32
Default Value: 1600
Store Supported: Yes

Description: Pulses per revolution of incremental encoder used for the Feedback feature.

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives

---

Name: **Feedback_Kp**
Index.Sub: **2230.4H**
Type: Unsigned32
Access: rw
Unit: --
Range: Unsigned32
Default Value: 20000
Store Supported: Yes

Description: Constant used for Feedback feature. Kp is used for position correction. Increasing Kp value the drive will supply more current to compensate the position error.

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives

---

Name: **Feedback_Kv**
Index.Sub: **2230.5H**
Type: Unsigned32
Access: rw
Unit: --
Range: Unsigned32
Default Value: 8000
Store Supported: Yes

Description: Constant used for Feedback feature. Kv is used for speed correction. Increasing Kv value the drive will supply more current to compensate speed error.

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives

---

Name: **Feedback_Kffw_Acc**
Index.Sub: **2230.6H**
Type: Unsigned32
Access: rw
Unit: --
Range: Unsigned32
Default Value: 14000
Store Supported: Yes

Description: Constant used for Feedback feature. The Kffw_Acc is used for acceleration when the motor moves forward. Increasing Kffw_Acc value the drive will supply more current during acceleration.

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives

---

Name: **Feedback_Kffw_Dec**
Index.Sub: **2230.7H**
Type: Unsigned32
Access: rw
Unit: --
Range: Unsigned32
Default Value: 14000
Store Supported: Yes

Description: Constant used for Feedback feature. The Kffw_Dec is used for deceleration when the motor moves forward. Increasing Kffw_Dec value the drive will supply more current during acceleration.

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives

---

Name: **Feedback_Ki**
Index.Sub: **2230.BH**
Type: Unsigned32
Access: rw
Unit: --
Range: Unsigned32
Default Value: 1000
Store Supported: Yes

Description: Constant used for Feedback feature. Ki is used for integral correction.

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives

---

Name: **Feedback_Ki_Limit**
Index.Sub: **2230.CH**
Type: Unsigned16
Access: rw
Unit: mA
Range: 0 ÷ (max drive current)
Default Value: 1500
Store Supported: Yes

Description: This value is used for Feedback feature to limit integral correction

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives

---

Name: **Feedback_Kalfas**
Index.Sub: **2230.DH**
Type: Unsigned16
Access: rw
Unit: --
Range: Unsigned16
Default Value: 0
Store Supported: Yes

Description: Constant used for Feedback feature. It is Phase advanced constant.

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives

---

Name: **Feedback_Iq_min**
Index.Sub: **2230.FH**
Type: Unsigned16
Access: rw
Unit: mA
Range: Unsigned16
Default Value: 500
Store Supported: Yes

Description: This value is used for Feedback feature and define the minimum current of the regulator.

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives

---

Name: **Feedback_Ka**
Index.Sub: **2230.10H**
Type: Unsigned32
Access: rw
Unit: --
Range: Unsigned32
Default Value: 0
Store Supported: Yes

Description: Constant used for Feedback feature during acceleration phase.

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives

---

Name: **Feedback_Boost_Current**
Index.Sub: **2230.12H**
Type: Unsigned16
Access: rw
Unit: mA
Range: 0 ÷ (max drive current)
Default Value: 1500
Store Supported: Yes

Description: This value is used for Feedback feature and define Maximum current supplied to the motor.

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives

---

Name: **Feedback_Kfbw_Acc**
Index.Sub: **2230.15H**
Type: Unsigned32
Access: rw
Unit: --
Range: Unsigned32
Default Value: 14000
Store Supported: Yes

Description: Constant used for Feedback feature. The Kfbw_Acc is used for acceleration when the motor moves backward. Increasing Kfbw_Acc value the drive will supply more current during acceleration.

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives

---

Name: **Feedback_Kfbw_Dec**
Index.Sub: **2230.16H**
Type: Unsigned32
Access: rw
Unit: --
Range: Unsigned32
Default Value: 14000
Store Supported: Yes

Description: Constant used for Feedback feature. The Kfbw_Dec is used for deceleration when the motor moves backward. Increasing Kfbw_Dec value the drive will supply more current during acceleration.

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives

---

Name: **BiSS_Encoder_Actual_Value**
Index.Sub: **2A00.0H**
Type: Signed32
Access: ro
Unit: Increments
Range: -2147483648 ÷ 2147483647
Default Value: 0
Store Supported: No

Description: This object contains the BiSS_Encoder_Internal_Value normalized to 32 bits and subtracted by the BiSS_Encoder_Offset_Value.

BiSS_Encoder_Actual_Value = BiSS_Encoder_Internal_Value - BiSS_Encoder_Offset_Value

Since the BiSS_Encoder_Internal_Value cannot be changed, it is possible to get the desired value acting on the Offset. For Instance if a particular position motor position have to become the 0 position, just set the BiSS_Encoder_Offset_Value = BiSS_Encoder_Internal_Value.

Notes:

This object is available only on drive models (SB4,SM4,SW4,SW5) fitted with BiSS Encoder input and only *with firmware V02r76 or superior.*

Name: **BiSS_Encoder_Status**
Index.Sub: **2A02.0H**
Type: Unsigned16
Access: ro
Unit: --
Range: 0 ÷ 65535
Default Value: 0
Store Supported: No

Description: This object contains the value of diagnostic bits of BiSS Encoder received communication frame. If the Encoder is correctly working the value of this object should be always equal to 3 otherwise there are some communication errors with the Encoder or the Encoder is faulty.

Bit #	Description
0	1 = Ok, Good 0 = Warning Condition
1	1 = Ok, Data Valid 0 = Error (HW Failure detected)
2 ÷ 15	Reserved

Notes:

This object is available only on drive models (SB4,SM4,SW4,SW5) fitted with BiSS Encoder input and only *with firmware V02r76 or superior.*

Name: **BiSS_Encoder_Config**
Index.Sub: **2A04.0H**
Type: Unsigned32
Access: rw
Unit: --
Range: 0 ÷ FFFFFFFFH
Default Value: 0 (BiSS Encoder disabled)
Store Supported: Yes

Description: This object contains the configuration for the BiSS Encoder. For the correct reading of the BiSS Encoder connected to the drive it is necessary to specify the right resolution (# of bits of either single turn or multiturn). For Instance if the connected BiSS Encoder has a resolution of 17 bits single turn and 16 bits multiturn, The *BiSS_Encoder_Config* object must be set equal to 4113 (1011H).

Byte 3	Byte 2	Byte 1	Byte 0
Reserved	Reserved	Multiturn Bits #	SingleTurn Bits #

Notes:

This object is available only on drive models (SB4,SM4,SW4,SW5) fitted with BiSS Encoder input and only *with firmware V02r76 or superior.*

Name: **BiSS_Encoder_RxErr**
Index.Sub: **2A06.0H**
Type: Unsigned16
Access: rw
Unit: --
Range: 0 ÷ 65535
Default Value: 0
Store Supported: No

Description: This object contains the number of BiSS Encoder receive errors. This object should be always 0 otherwise there are some communication errors with the Encoder. Check the cabling and the cable length to solve communication errors. The error counter can be cleared setting it to 0.

Notes:

This object is available only on drive models (SB4,SM4,SW4,SW5) fitted with BiSS Encoder input and only *with firmware V02r76 or superior.*

Name: **BiSS_Encoder_Offset_Value**
Index.Sub: **2A08.0H**
Type: Signed32
Access: rw
Unit: Increments
Range: -2147483648 ÷ 2147483647
Default Value: 0
Store Supported: Yes

Description: This object contains the offset used to calculate the *BiSS_Encoder_Actual_Value* object.

Notes:

This object is available only on drive models (SB4,SM4,SW4,SW5) fitted with BiSS Encoder input and only *with firmware V02r76 or superior.*

Name: **BiSS_Encoder_Internal_Value**
Index.Sub: **2A0A.0H**
Type: Signed32
Access: ro
Unit: Increments
Range: -2147483648 ÷ 2147483647
Default Value: --
Store Supported: No

Description: This object contains the actual BiSS Encoder value normalized to 32 bits. Regardless of the Encoder multiturn and singleturn resolution, this object contains in the low word the single turn position and in the high word the number of turns.

Notes:

This object is available only on drive models (SB4,SM4,SW4,SW5) fitted with BiSS Encoder input and only *with firmware V02r76 or superior.*

Name: **Node_Id**
Index.Sub: **4000.7H**
Type: Unsigned16
Access: rw
Unit: --
Range: 1 ÷ 127
Default Value: 1
Store Supported: Yes

Description: This object contains the drive's NodeId. On SW1 drives this object can be changed only via dip-switches/rotoswitches. On SM1A drives can be changed using the Slim Line Drives CAN Monitor Software Tool or following the procedure explained at §2.1.2.

Notes: This object is not available in EtherCAT fieldbus.

Name: **Baud_Rate**
Index.Sub: **4000.8H**
Type: Unsigned16
Access: rw
Unit: --
Range: 0 ÷ 3 (0 = 1M, 1 = 500k, 2 = 250k, 3 = 125k)
Default Value: 1
Store Supported: Yes

Description: This object contains the drive's Baud Rate for the CAN interface. On SW1 drives this object can be changed only via dip-switches/rotoswitches. On SM1A drives can be changed using the Slim Line Drives CAN Monitor Software Tool or following the procedure explained at §2.1.2.

Notes: This object is not available in EtherCAT fieldbus.

Name: **Firmware_Version**
Index.Sub: **4004.0H**
Type: Unsigned16 (high byte = version ; low byte = release)
Access: ro
Unit: **--**
Range: **--**
Default Value: --
Store Supported: No

Description:

Notes:

Name: **Firmware_Checksum**
Index.Sub: **4004.1H**
Type: Unsigned16
Access: ro
Unit: **--**
Range: 0 ÷ FFFFH
Default Value: --
Store Supported: No

Description: This buffer contains the checksum of the firmware stored in drive.

Notes: This object is not available in EtherCAT fieldbus.

Name: **Boot_Version**
Index.Sub: **4004.2H**
Type: Unsigned16 (high byte = version ; low byte = release)
Access: ro
Unit: **--**
Range: **--**
Default Value: --
Store Supported: No

Description:

Notes: This object is not available in EtherCAT fieldbus.

Name: Drive_Type
Index.Sub: 4004.6H
Type: Unsigned16 (high byte = Drive Type ; low byte = hardware revision)
Access: ro
Unit: --
Range: --
Default Value: --
Store Supported: No

Description: It contains the information about the drive type and hardware version. The MSB contains the drive board family, while the LSB contains the hardware version.

MSB Value	Description
01H	M5A Board
10H	DMD00 Board (SW1x4080)
11H	DMD01 Board (SW1D2x42)
12H	DMD02 Board (SW1A9x60)
13H	DMD10 Board (SM1A)
14H	DMD30 Board (SW1xx085)
15H	DMD31 Board (SW1xx085)
30H	CSMD1
31H	TWD
32H	IMD00 (SB4A)
34H	ISD00 (SW5)
35H	IMD01 (SW4)
37H	IMD02 (SB4D)
80H	DLD00 Board
81H	DLD01 Board

Notes: This object is not available in EtherCAT fieldbus.

Name: Configuration_Code
Index.Sub: 4004.7H
Type: Unsigned16
Access: ro
Unit: --
Range: --
Default Value: --
Store Supported: No

Description: This object contains the firmware configuration code.

Notes: This object is not available in EtherCAT fieldbus.

Name: **Warning_Inserted**
Index.Sub: **59A2.0H**
Type: Unsigned16
Access: ro
Unit: --
Range: --
Default Value: --
Store Supported: No

Description: This object contains the state of the internal warning of the firmware.

Bit #	Name	Description	Default value
0	Warning_Voltage	1 = Drive voltage is near limit 0 = Ok	0
1	Warning_Temperature	1 = Drive temperature is near limit 0 = Ok	0
2	Warning_EEprom_Near_Write_Overrun	1 = EEprom near Write Overrun. 0 = Ok	0
3	Warning_EEprom_Near_EOL	1 = EEprom near End of Life 0 = Ok	0
4	Reserved	Reserved	0
5	Reserved	Reserved	0
6	Reserved	Reserved	0
7	Reserved	Reserved	0
8	Warning_Missing_Inominal	1 = Missing Nominal Current parameter 0 = Ok	0
9	Reserved	Reserved	0
10	Reserved	Reserved	0
11	Reserved	Reserved	0
12	Reserved	Reserved	0
13	Reserved	Reserved	0
14	Reserved	Reserved	0
15	Reserved	Reserved	0

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r17 or superior.

Name: **B0_Digital_Inputs**
Index.Sub: **6000.1H**
Type: Unsigned8
Access: ro
Unit: **--**
Range: 0 (all inputs are open) ÷ 2^n-1 (all inputs are closed, where n is the number of digital inputs available)
Default Value: --
Store Supported: No

Description: It contains the status of all the inputs on bank 0 of the drive.

Notes: See B Appendix.
A numeric value is associated to the input, following the procedure laid out hereafter:

INPUTS	VALUE
Input 0	1
Input 1	2
Input 2	4
Input 3	8
Input 4	16
Input 5	32
Input n	2^n

Name: **Error Code**
Index.Sub: **603F.0H**
Object Code: var
Data Type: Unsigned16
Access: ro
Unit: **---**
Range: ---
Default Value: 0
Store Supported: No

Description: This object captures the code of the last error that occurred in the drive.
See §2.8.1 for emergency codes.

Value (hex)	Description
0000 hex	No error
FF01-FFFF hex	Manufacturer specific

Name: **Controlword**
Index.Sub: **6040.0H**
Data Type: Unsigned16
PDO Mapping: Yes
Access: rw
Unit: **---**
Range: ---
Default Value: ---
Store Supported: No

Description: The controlword controls the state machine, operating modes and manufacturer specific options.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Manufacturer specific					Reserved		Halt	Fault reset	Operation mode specific			Enable Operation	Quick Stop	Enable Voltage	Switch on
O					O		O	M	O			M	M	M	M

Bits 0,1,2,3,7 : Device control commands (see Table 3).

Bits 4,5,6,8 : Operation mode specific: the description is in the chapter of the specific Operation mode.

Bits 9,10 : Reserved

Bits 11,12,13,14,15 : Manufacturer specific.

Interpolated Position Mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Manufacturer specific					Reserved		Halt	Fault reset	Operation mode specific			Enable Operation	QuickStop	Enable Voltage	Switch on
O					O		O	M	O			M	M	M	M
X	X	X	X	X	X	X	**Halt**	X	reserved		**Enable ip mode**	X	X	X	X

Name	Value	Description
enable ip mode (bit4)	0	Interpolated position mode inactive
	1	Interpolated position mode active
Halt (bit8)	0	Execute the instruction of bit 4
	1	Stop axle

Profile Position Mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Manufacturer specific					Reserved		Halt	Fault reset	Operation mode specific			Enable Operation	Quick Stop	Enable Voltage	Switch on
O					O		O	M				M	M	M	M
X	X	X	X	X	X	X	**Halt**	X	**Abs / Rel**	**Change set immediatel y**	**New set point**	X	X	X	X

Name	Value	Description
New set-point (bit4)	0	Does not assume *Target Position*
	1	Assume *Target Position*
Change set immediately (bit5)	0	Finish the actual positioning and then start the next positioning
	1	Interrupt the actual positioning and start the next positioning
Abs / Rel (bit6)	0	*Target Position* is an absolute value
	1	*Target Position* is a relative value
Halt (bit8)	0	Execute positioning
	1	Stop axle with profile deceleration

Homing Mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Manufacturer specific					Reserved		Halt	Fault reset	Operation mode specific			Enable Operation	Quick Stop	Enable Voltage	Switch on
O					O		O	M				M	M	M	M
X	X	X	X	X	X	X	**Halt**	X	reserved		**Homing operation start**	X	X	X	X

Name	Value	Description
Homing operation start	0	Homing mode inactive
	0 → 1	Start homing mode
	1	Homing mode active
	1 → 0	Interrupt homing mode
Halt (bit8)	0	Execute the instruction of bit 4
	1	Stop axle with homing acceleration

Profile Velocity Mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Manufacturer specific					Reserved		Halt	Fault reset	Operation mode specific			Enable Operation	Quick Stop	Enable Voltage	Switch on
O					O		O	M				M	M	M	M
X	X	X	X	X	X	X	**Halt**	X	**X**	**X**	**X**	**X**	X	X	X

Name	Value	Description
Halt (bit8)	0	Execute the motion
	1	Stop axle with profile deceleration

Cyclic Synchronous Position Mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Manufacturer specific					Reserved		Halt	Fault reset	Operation mode specific			Enable Operation	Quick Stop	Enable Voltage	Switch on
O					O		O	M				M	M	M	M
X	X	X	X	X	X	X	**Halt**	X	**X**	**X**	**X**	**X**	X	X	X

Name	Value	Description
Halt (bit8)	0	Execute the motion
	1	Stop axle with profile deceleration

Cyclic Synchronous Velocity Mode

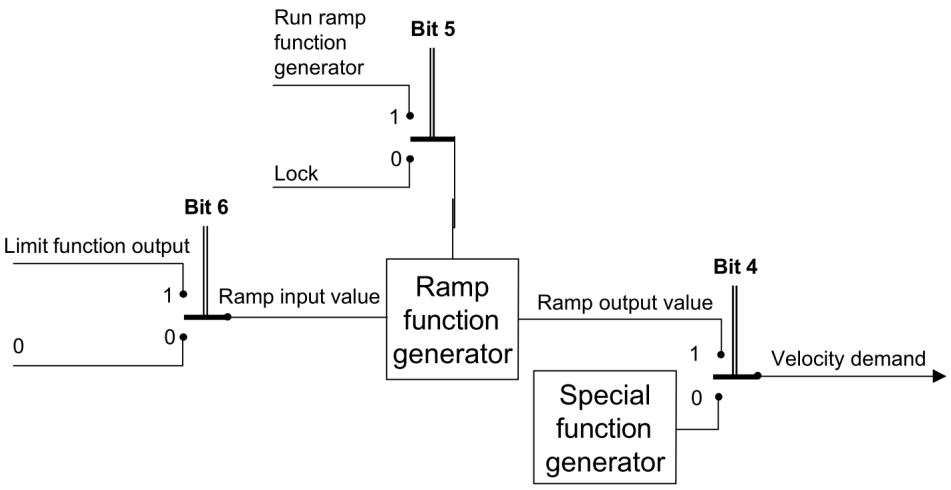
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Manufacturer specific					Reserved		Halt	Fault reset	Operation mode specific			Enable Operation	Quick Stop	Enable Voltage	Switch on
O					O		O	M				M	M	M	M
X	X	X	X	X	X	X	**Halt**	X	**X**	**X**	**X**	**X**	X	X	X

Name	Value	Description
Halt (bit8)	0	Execute the motion
	1	Stop axle with profile deceleration

Velocity Mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Manufacturer specific					Reserved		Halt	Fault reset	Operation mode specific			Enable Operation	Quick Stop	Enable Voltage	Switch on
O					O		O	M	O	O	O	M	M	M	M
X	X	X	X	X	X	X	**Halt**	X	**Reference ramp**	**Unlock ramp**	**Enable ramp**	X	X	X	X

Name	Value	Description
Enable ramp (bit4)	0	Velocity demand value shall be controlled in any other (manufacturer-specific) way
	1	Velocity demand value accords to ramp output value
Unlock ramp (bit5)	0	Ramp output value is locked to current output value
	1	Ramp output value follows ramp input value
Reference ramp (bit6)	0	Ramp input value is set to zero
	1	Ramp input value accords to ramp reference
Halt (bit8)	0	Execute motion
	1	Stop axle



Note :
The Velocity mode is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior

Name: **Statusword**
Index.Sub: **6041.0H**
Data Type: Unsigned16
PDO Mapping: Yes
Access: ro
Unit: **---**
Range: ---
Default Value: ---
Store Supported: No

Description: The Statusword indicates the current state of the drive, the operating state of the mode and manufacturer specific options.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Manuf. specific		Op. mode specific		Int. limit active	Target reached	Remote	Manuf. specific	Warning	Switch on disabled	Quick stop	Voltage enabled	Fault	Operation enabled	Switched on	Ready to switch on
O		O		M	M	M	O	O	M	M	M	M	M	M	M

Bits 0,1,2,3,5,6 : Device state (see table 1)

Bit4 : High voltage is applied to the drive (if bit is set to 1)

Bit5 : If this bit is equal to 0 , the drive is reacting on a quick stop request.

Bit7 : If this bit is set to 1, a drive warning is present. The cause means no error but a state that has to be mentioned. The status of the drive does not change. The cause of the warning may be found by reading the object 59A2h.0h. This bit is set and reset by the device.

Bit8 : This bit may be used by a drive manufacturer to implement any manufacturer specific functionality.

Bit9 : If this bit is set to 1, then parameters may be modified via CAN-network and the drive executes the content of a command message. If this bit is equal to 0, then the drive is in local mode and will not execute the command message. In local mode the drive will accept accesses via SDO.

Bit10 : If this bit is set to 1 then a set-point has been reached. The set-point is dependent on the operating mode. The description is in the chapter of the specific Operation mode. The change of a target value alters this bit.

Bit11 : This bit indicates that an internal limitation is active.

Bits 12,13 : This bits are operation mode specific. The description is in the chapter of the special mode.

Bits 14,15 : These bits may be used by a drive manufacturer to implement any manufacturer specific functionality.

Interpolated Position Mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Manuf. specific		Operation mode specific		Internal limit active	Target reached	Remote	Manuf. specific	Warning	Switch on disabled	Quick stop	Voltage enabled	Fault	Operation enabled	Switched on	Ready to switch on
O		O		M	M	M	O	O	M	M	M	M	M	M	M
X	X	reserved	Ip mode active	X	Target reached	X	X	X	X	X	X	X	X	X	X

		Description
Target reached (bit10)	0	Halt = 0 : Position not (yet) reached Halt = 1 : Motor in deceleration
	1	Halt = 0 : Position reached Halt = 1 : Motor speed zero
Ip mode active (bit12)	0	Interpolated position mode inactive
	1	Interpolated position mode active

Profile Position Mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Manuf. specific		Operation mode specific		Internal limit active	Target reached	Remote	Manuf. specific	Warning	Switch on disabled	Quick stop	Voltage enabled	Fault	Operation enabled	Switched on	Ready to switch on
O		O		M	M	M	O	O	M	M	M	M	M	M	M
X	X	Following error	Set-point ack	X	Target reached	X	X	X	X	X	X	X	X	X	X

Name	Value	Description
Target reached (bit10)	0	Halt = 0 : *Target Position* not reached Halt = 1 : Axle decelerates
	1	Halt = 0 : *Target Position* reached Halt = 1 : Velocity of axle is 0
Set-point acknowledge (bit12)	0	Trajectory generator has not assumed the positioning values (yet)
	1	Trajectory generator has assumed the positioning values
Following error (bit13)	0	No following error
	1	Following error

Homing Mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Manuf. specific		Operation mode specific		Internal limit active	Target reached	Remote	Manuf. specific	Warning	Switch on disabled	Quick stop	Voltage enabled	Fault	Operation enabled	Switched on	Ready to switch on
O		O		M	M	M	O	O	M	M	M	M	M	M	M
X	X	Homing error	Homing attained	X	Target reached	X	X	X	X	X	X	X	X	X	X

Name	Value	Description
Target reached (bit10)	0	Halt = 0 : *Homing position* not reached Halt = 1 : Axle decelerates
	1	Halt = 0 : Homing position reached Halt = 1 : Velocity of axle is 0
Homing attained (bit12)	0	Homing mode not yet completed
	1	Homing mode carried out successfully
Homing error (bit13)	0	No Homing error
	1	Homing error occurred Homing mode carried out not successfully The error cause is found by reading the error code

Profile Velocity Mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Manuf. specific		Operation mode specific		Internal limit active	Target reached	Remote	Manuf. specific	Warning	Switch on disabled	Quick stop	Voltage enabled	Fault	Operation enabled	Switched on	Ready to switch on
O		O		M	M	M	O	O	M	M	M	M	M	M	M
X	X	Max slippage error	Speed	X	Target reached	X	X	X	X	X	X	X	X	X	X

Name	Value	Description
Target reached (bit10)	0	Halt = 0 : *Target Velocity* not (yet) reached Halt = 1 : Axle decelerates
	1	Halt = 0 : *Target Velocity* reached Halt = 1 : Velocity of axle is 0
Speed (bit12)	0	Not supported
	1	
Max slippage error (bit13)	0	Not supported
	1	

Cyclic Synchronous Position Mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Manuf. specific		Operation mode specific		Internal limit active	Target reached	Remote	Manuf. specific	Warning	Switch on disabled	Quick stop	Voltage enabled	Fault	Operation enabled	Switched on	Ready to switch on
O		O		M	M	M	O	O	M	M	M	M	M	M	M
X	X	Following Error	Target Position Ignored	X	Reserv.	X	X	X	X	X	X	X	X	X	X

Name	Value	Description
Reserved (bit10)	0	Reserved
	1	Reserved
Target Position Ignored (bit12)	0	Target position ignored
	1	Target position shall be used as input to position control loop
Following Error (bit13)	0	No following error
	1	Following error

Cyclic Synchronous Velocity Mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Manuf. specific		Operation mode specific		Internal limit active	Target reached	Remote	Manuf. specific	Warning	Switch on disabled	Quick stop	Voltage enabled	Fault	Operation enabled	Switched on	Ready to switch on
O		O		M	M	M	O	O	M	M	M	M	M	M	M
X	X	Reserv.	Target Velocity Ignored	X	Reserv.	X	X	X	X	X	X	X	X	X	X

Name	Value	Description
Reserved (bit10)	0	Reserved
	1	Reserved
Target Velocity Ignored (bit12)	0	Target velocity ignored
	1	Target velocity shall be used as input to velocity control loop
Reserved (bit13)	0	Reserved
	1	Reserved

Velocity Mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Manuf. specific		Operation mode specific		Internal limit active	Target reached	Remote	Manuf. specific	Warning	Switch on disabled	Quick stop	Voltage enabled	Fault	Operation enabled	Switched on	Ready to switch on
O		O		M	M	M	O	O	M	M	M	M	M	M	M
X	X	reserved (0)	reserved (0)	X	reserved (0)	X	X	X	X	X	X	X	X	X	X

Note: The Velocity mode is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior

Name: **vl Target Velocity**
Index.Sub: **6042.0H**
Type: Integer16
Access: rw
Unit: User-defined velocity unit [2]
Range: Integer16
Default Value: 0
Store Supported: No

Description: The 'vl Target Velocity' is the required velocity for the system.

Positive values shall indicate forward direction and negative values shall indicate backward direction.

Notes: [1] This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior.

[2] The User-defined velocity unit is converted to the revolution per minute (rpm) by mean of 'vl dimension factor' :

Velocity [user-defined units] x 'vl dimension factor' = Velocity [rpm]

[3] The 'device internal velocity unit' is 'rpm' [revs/min].

Name: **vl Velocity Demand**
Index.Sub: **6043.0H**
Type: Integer16
Access: ro
Unit: User-defined velocity unit [2]
Range: Integer16
Default Value: No
Store Supported: No

Description: The 'vl Velocity Demand' is the instantaneous velocity generated by the ramp function.

Positive values shall indicate forward direction and negative values shall indicate backward direction.

Notes: [1] This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior.

[2] The User-defined velocity unit is converted to the revolution per minute (rpm) by mean of 'vl dimension factor' :

Velocity [user-defined units] x 'vl dimension factor' = Velocity [rpm]

[3] The 'device internal velocity unit' is 'rpm' [revs/min].

Name: **vl Control Effort**
Index.Sub: **6044.0H**
Type: Integer16
Access: ro
Unit: User-defined velocity unit [(2)]
Range: Integer16
Default Value: No
Store Supported: No

Description: This object is the velocity at the motor spindle or load (without closed loop control this value reads the 'vl Velocity Demand').

Positive values shall indicate forward direction and negative values shall indicate backward direction.

Notes: [(1)] This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior.

[(2)] The User-defined velocity unit is converted to the revolution per minute (rpm) by mean of 'vl dimension factor' :

Velocity [user-defined units] x 'vl dimension factor' = Velocity [rpm]

[(3)] The 'device internal velocity unit' is 'rpm' [revs/min].

Name: **vl Velocity min max amount (number of entries)**
Index.Sub: **6046.0H**
Type: Unsigned8
Access: ro
Unit: ---
Range: 2
Default Value: 2
Store Supported: No

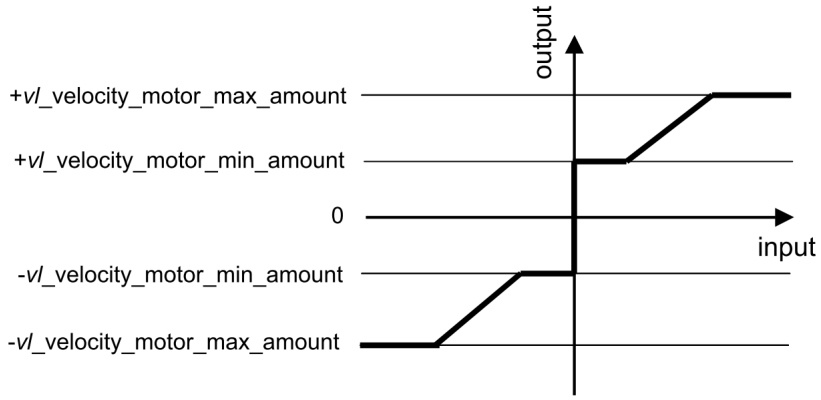
Description: Highest sub-index supported

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior.

Name: **vl Velocity min amount**
Index.Sub: **6046.1H**
Type: Unsigned32
Access: rw
Unit: User-defined velocity unit [2]
Range: Unsigned32
Default Value: 1
Store Supported: No

Description: This object indicates the configured minimum amount of velocity.

The 'vl Velocity min amount' is mapped internally to the 'vl velocity min pos' and 'vl velocity min neg' values. Only the positive values is returned if the 'vl Velocity min amount' is read out.



Notes: [1] This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior.

[2] The User-defined velocity unit is converted to the revolution per minute (rpm) by mean of 'vl dimension factor' :

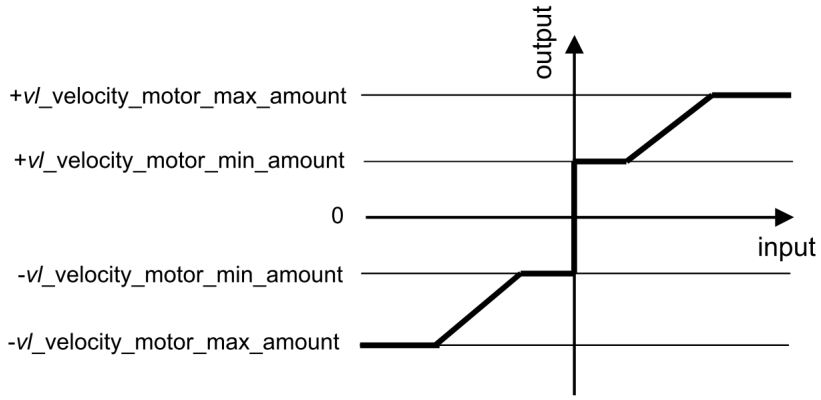
Velocity [user-defined units] x 'vl dimension factor' = Velocity [rpm]

[3] The 'device internal velocity unit' is 'rpm' [revs/min].

Name: **vl Velocity max amount**
Index.Sub: **6046.2H**
Type: Unsigned32
Access: rw
Unit: User-defined velocity unit [2]
Range: Unsigned32
Default Value: 2000
Store Supported: No

Description: This object indicates the configured maximum amount of velocity.

The 'vl Velocity max amount' is mapped internally to the 'vl velocity max pos' and 'vl velocity max neg' values. Only the positive values is returned if the 'vl Velocity max amount' is read out.



Notes: [1] This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior.

[2] The User-defined velocity unit is converted to the revolution per minute (rpm) by mean of 'vl dimension factor' :

Velocity [user-defined units] x 'vl dimension factor' = Velocity [rpm]

[3] The 'device internal velocity unit' is 'rpm' [revs/min].

Name: **vl Velocity acceleration (number of entries)**
Index.Sub: **6048.0H**
Type: Unsigned8
Access: ro
Unit: ---
Range: 2
Default Value: 2
Store Supported: No

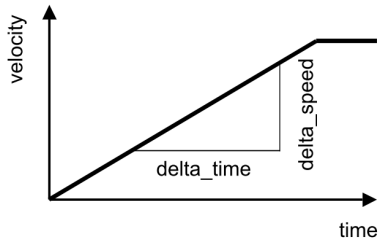
Description: Highest sub-index supported

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior.

Name: **vl Velocity acceleration (delta speed)**
Index.Sub: **6048.1H**
Type: Unsigned32
Access: rw
Unit: User-defined velocity unit [(2)]
Range: Unsigned32
Default Value: 1500
Store Supported: No

Description: This object indicates the configured 'delta speed' of the slope for acceleration ramp :

$$vl\ velocity\ acceleration = \frac{delta\ speed}{delta\ time}$$



Notes: [(1)] This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior.

[(2)] The User-defined velocity unit is converted to the revolution per minute (rpm) by mean of 'vl dimension factor' :

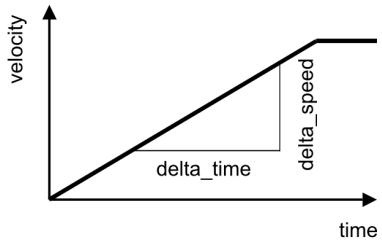
Velocity [user-defined units] x 'vl dimension factor' = Velocity [rpm]

[(3)] The 'device internal velocity unit' is 'rpm' [revs/min].

Name: **vl Velocity acceleration (delta time)**
Index.Sub: **6048.2H**
Type: Unsigned16
Access: rw
Unit: seconds
Range: Unsigned16
Default Value: 1
Store Supported: No

Description: This object indicates the configured 'delta time' of the slope for acceleration ramp :

$$vl\ velocity\ acceleration = \frac{delta\ speed}{delta\ time}$$



The value of 'delta time' shall be given in seconds.
This function directly follows the set-point if the parameter 0 is defined for the delta time value.

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior.

Name: **vl Velocity deceleration (number of entries)**
Index.Sub: **6049.0H**
Type: Unsigned8
Access: ro
Unit: ---
Range: 2
Default Value: 2
Store Supported: No

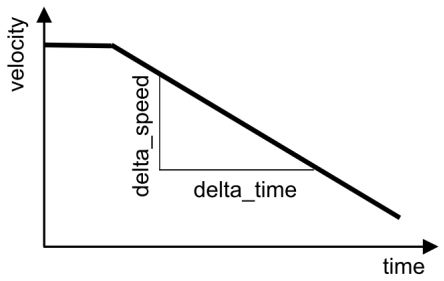
Description: Highest sub-index supported

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior.

Name: **vl Velocity deceleration (delta speed)**
Index.Sub: **6049.1H**
Type: Unsigned32
Access: rw
Unit: User-defined velocity unit [(2)]
Range: Unsigned32
Default Value: 1500
Store Supported: No

Description: This object indicates the configured 'delta speed' of the slope for deceleration ramp :

$$vl\ velocity\ deceleration = \frac{delta\ speed}{delta\ time}$$



Notes: [(1)] This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior.

[(2)] The User-defined velocity unit is converted to the revolution per minute (rpm) by mean of 'vl dimension factor' :

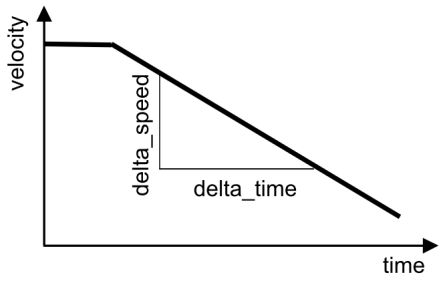
Velocity [user-defined units] x 'vl dimension factor' = Velocity [rpm]

[(3)] The 'device internal velocity unit' is 'rpm' [revs/min].

Name: **vl Velocity deceleration (delta time)**
Index.Sub: **6049.2H**
Type: Unsigned16
Access: rw
Unit: seconds
Range: Unsigned16
Default Value: 1
Store Supported: No

Description: This object indicates the configured 'delta time' of the slope for deceleration ramp :

$$vl\ velocity\ deceleration = \frac{delta\ speed}{delta\ time}$$



Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior.

Name: **vl Dimension Factor (number of entries)**
Index.Sub: **604C.0H**
Type: Unsigned8
Access: ro
Unit: ---
Range: 2
Default Value: 2
Store Supported: No

Description: Highest sub-index supported

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior.

Name: **vl Dimension Factor (numerator)**
Index.Sub: **604C.1H**
Type: Integer32
Access: rw
Unit: --
Range: Integer32 (except the value 0)
Default Value: 1
Store Supported: No

Description: This object indicates the numerator of the 'vl dimension factor'.

The 'vl dimension factor' is used to include gearing in calculation or serves to scale the frequencies or specific units of the user. It influences the 'vl target velocity', 'vl velocity demand', 'vl control effort' as well as the velocity limit function and ramp function.

The purpose of the 'vl dimension factor' is to convert the 'user-defined velocity unit' to the revolution/minute unit (rpm) :

Velocity [user-defined unit] x 'vl Dimension factor' = Velocity [rpm]

The 'vl dimension factor' shall be in the range of -2147483648 to +2147483647, but the value 0 shall be not used.

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior.

The 'vl dimension factor' is used only for objects retaled to the 'Velocity Mode (vl)'.

Name: **vl Dimension Factor (denominator)**
Index.Sub: **604C.2H**
Type: Integer32
Access: rw
Unit: --
Range: Integer32 (except the value 0)
Default Value: 1
Store Supported: No

Description: This object indicates the denominator of the 'vl dimension factor'.

The 'vl dimension factor' is used to include gearing in calculation or serves to scale the frequencies or specific units of the user. It influences the 'vl target velocity', 'vl velocity demand', 'vl control effort' as well as the velocity limit function and ramp function.

The purpose of the 'vl dimension factor' is to convert the user-defined unit to the revolution/minute unit (rpm) :

Velocity [user-defined unit] x 'vl Dimension factor' = Velocity [rpm]

The 'vl dimension factor' shall be in the range of -2147483648 to +2147483647, but the value 0 shall be not used.

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r69 or superior.

The 'vl dimension factor' is used only for objects retaled to the 'Velocity Mode (vl)'.

Name: **Quick_stop_option_code**
Index.Sub: **605A.0H**
Type: Integer16
Access: rw
Unit: --
Range: 0 = Disable drive function, motor is free to rotate
2 = Slow down with quick stop ramp and transit into Switch On Disabled
6 = Slow down with quick stop ramp and stay in Quick Stop Active

Default Value: 0
Store Supported: Yes *(1)*

Description: This parameter determines what action should be taken if the *Quick Stop Function* is executed.

Notes: *(1)* 'Store supported' is available for this object *only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r38 or superior.*

Name: **Halt_option_code**
Index.Sub: **605D.0H**
Type: Integer16
Access: rw
Unit: --
Range: 2 = Slow down with quick stop ramp

Default Value: 2
Store Supported: No

Description: This parameter determines what action should be taken if the bit8 (Halt) in the *Controlword* is active

Notes:

Name: **Fault_reaction_option_code**
Index.Sub: **605E.0H**
Type: Integer16
Access: rw
Unit: --
Range: 0 = Disable drive, motor is free to rotate
1 = Slow down with slow down ramp
2= Slow down with quick stop ramp

Default Value: 0
Store Supported: Yes(1)

Description: This parameter determines what action should be taken if a non-fault occurs in the drive.

Notes:

(1) 'Store supported' is available for this object *only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V02r59 or superior.*

Name: **Modes of operation**
Index.Sub: **6060.0H**
Data Type: Integer8
PDO Mapping: Yes
Access: rw
Unit: **---**
Range: 9
Default Value: No
Store Supported: No

Description: This parameter switches the currently chosen operation mode. A read of **modes of operation** shows only the value of modes of operation. The current mode of the drive is reflected in the object **modes of operation display**.

Value	*Description*
1	Profile Position Mode
2	Velocity Mode
3	Profile Velocity Mode
6	Homing mode
7	Interpolated Position Mode
8	Cyclic Synchronous Position Mode
9	Cyclic Synchronous Velocity Mode

Name: **Modes of operation display**
Index.Sub: **6061.0H**
Data Type: Integer8
PDO Mapping: Yes
Access: ro
Unit: **---**
Range: 9
Default Value: No
Store Supported: No

Description: This parameter shows the current mode of operation. The meaning of the returned value corresponds to that of the **modes of operation** (object 6060h).

Value	*Description*
1	Profile Position Mode
2	Velocity Mode
3	Profile Velocity Mode
6	Homing mode
7	Interpolated Position Mode
8	Cyclic Synchronous Position Mode
9	Cyclic Synchronous Velocity Mode

Name: **Position_demand_value**
Index.Sub: **6062.0H**
Type: Integer32
Access: ro
Unit[1] : Increments
Unit[2] : User-defined position unit [3]
Range: Integer32
Default Value: --
Store Supported: No

Description: It indicates the present position demand value output from the trajectory generator.

Notes: *[1] for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower'*

[2] for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior'

[3] See 'Factor Group' chapter

[4] The 'device internal position unit' is 'Increments'

Name: **Position_Actual_Internal_Value**
Index.Sub: **6063.0H**
Type: Integer32
Access: ro
Unit: Increments
Range: -2147483648 ÷ 2147483647
Default Value: 0
Store Supported: No

Description: This object provides the position actual value of the encoder or motor (by mean of 'Sensor_selection_code' object is possible to select the motor or encoder reference). The value is given in Increments.

Notes:

Name: **Position_Actual_Value**
Index.Sub: **6064.0H**
Type: Integer32
Access: ro
Unit[(1)] : Increments
Unit[(2)] : User-defined position unit [(3)]
Range: -2147483648 ÷ 2147483647
Default Value: 0
Store Supported: No

Description: This object provides the position actual value of the encoder or motor (by mean of 'Sensor_selection_code' object is possible to select the motor or encoder reference).

Notes: *[(1)] for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower'*

[(2)] for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior'

[(3)] See 'Factor Group' chapter

[(4)] The 'device internal position unit' is 'Increments'

Name: **Following_error_window**
Index.Sub: **6065.0H**
Type: Unsigned32
Access: rw
Unit[(1)] : Increments
Unit[(2)] : User-defined position unit [(3)]
Range: Unsigned32
Default Value: FFFFFFFFH
Store Supported: No

Description: It defines a range of tolerated position values symmetrically to the *Position_demand* value. If the *Position_actual_value* is out of the Following_error_window longer than the *Following_error_time_out* time, a following error occurs and the corresponding bit13 (following error) in the *Statusword.*

To reset following error state, is possible to use the Homing_method 35 after that the emergency condition is reset by mean of the Fault_reset command in the Controlword.

If the value of *Following_error_window* is FFFFFFFFH, the following control will be switched off.

Notes: *[(1)] for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower'*

[(2)] for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior'

[(3)] See 'Factor Group' chapter

[(4)] The 'device internal position unit' is 'Increments'

Name: **Following_error_time_out**
Index.Sub: **6066.0H**
Type: Unsigned16
Access: rw
Unit: ms
Range: Unsigned16
Default Value: 0
Store Supported: No

Description: When the following error occurs longer than the defined value of the time-out, the corresponding bit 13 (following error) in the Statusword will be set to one.

Notes:

Name: **Position_window**
Index.Sub: **6067.0H**
Type: Integer32
Access: rw
Unit[1] : Increments
Unit[2] : User-defined position unit [3]
Range: Integer32
Default Value: FFFF FFFFh
Store Supported: No

Description: This object defines a symmetrical range of accepted positions relatively to the target position: (target_position – position_window ; target_position + position_window). If the position is within the position window, this target position is regarded as reached.

Notes: *[1] for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower'*

[2] for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior'

[3] See 'Factor Group' chapter

[4] The 'device internal position unit' is 'Increments'

[5] If the value of the position window is 2^{32} -1, the position window control is switched off.

Name: **Position_window_time**
Index.Sub: **6068.0H**
Type: Unsigned16
Access: rw
Unit: ms
Range: Unsigned16
Default Value: 0
Store Supported: No

Description: When the actual position is within the *position window* during the defined *position window time*, the corresponding bit10 *target reached* in the *Statusword* will be set to one.

Notes:

Name: **Sensor_selection_code**
Index.Sub: **606A.0H**
Type: Integer16
Access: rw
Unit: --
Range: --
Default Value: -1
Store Supported: Yes [(1)]

Description: It selects whether an encoder is to be user or not :

0 = Encoder used (#0) [Incremental Encoder]
-1 = No encoder
-2 = Encoder used (#1) [Incremental Encoder]
-3 = Encoder used (#3) [Absolute Encoder multi turn BiSS][(2)]

Notes:

[(1)] 'Store supported' is available for this object *only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r36 or superior.*

[(2)] This object is available only on drive models (SB4,SM4,SW4,SW5) fitted with BiSS Encoder input and only *with firmware V02r76 or superior. The Biss Encoder resolution is fixed to 65536 pulses/rev.*

Name: **Velocity_demand_value**
Index.Sub: **606B.0H**
Type: Integer32
Access: ro
Unit[(1)] : *Inc/s*
Unit[(2)] : User-defined velocity unit [(3)]
Range: Integer32
Default Value: --
Store Supported: No

Description: It indicates the output of the trajectory generator.

Notes: [(1)] *for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower'*

[(2)] *for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior '*

[(3)] *See 'Factor Group' chapter*

[(4)] The 'device internal velocity unit' is 'Increments/sec'

Name: **Velocity_Actual_Value**
Index.Sub: **606C.0H**
Type: Integer32
Access: ro
Unit[(1)] : *Inc/s*
Unit[(2)] : User-defined velocity unit [(3)]
Range: Integer32
Default Value: 0
Store Supported: No

Description: This object contains the current velocity of the motor.

Notes: *[(1)] for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower '*

[(2)] for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior '

[(3)] See 'Factor Group' chapter

[(4)] The 'device internal velocity unit' is 'Increments/sec'

Name: **Velocity_window**
Index.Sub: **606D.0H**
Type: Unsigned16
Access: rw
Unit[(1)] : *Inc/s*
Unit[(2)] : User-defined velocity unit [(3)]
Range: Unsigned16
Default Value: 0
Store Supported: No

Description: This object is used to to handle the bit10 (target reached) in the Statusword. When the difference between the Target_velocity and the Velocity_actual_value is within the Velocity_window longer than the Velocity_window_time, the bit10 is set.

Notes: *[(1)] for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower '*

[(2)] for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior '

[(3)] See 'Factor Group' chapter

[(4)] The 'device internal velocity unit' is 'Increments/sec'

Name: **Velocity_window_time**
Index.Sub: **606E.0H**
Type: Unsigned16
Access: rw
Unit: ms
Range: Unsigned16
Default Value: 0
Store Supported: No

Description: This object is used to to handle the bit10 (target reached) in the Statusword. When the difference between the Target_velocity and the Velocity_actual_value is within the Velocity_window longer than the Velocity_window_time, the bit10 is set.

Notes:

Name: **Current_Actual_Value**
Index.Sub: **6078.0H**
Type: Unsigned16
Access: ro
Unit: mA
Range: 0 ÷ (max drive current)
Default Value: 0
Store Supported: No

Description: This object contains the current value of the current supplied to the motor (Irms).

Notes:

Name: **Target_Position**
Index.Sub: **607A.0H**
Type: Integer32
Access: rw
Unit[1] : Increments
Unit[2] : User-defined position unit [3]
Range: Integer32
Default Value: --
Store Supported: No

Description: It indicates the position that the drive should move to in position profile mode using the current settings of motion control parameters such as velocity, acceleration, deceleration, motion profile type, etc. The target position will be interpreted as absolute or relative depending on the 'abs/rel' flag in the Controlword.

Notes: *[1] for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower '*

[2] only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior

[3] See 'Factor Group' chapter

[4] The 'device internal position unit' is 'Increments'

Name: **Home_Offset**
Index.Sub: **607C.0H**
Type: Integer32
Access: rw
Unit[1] : Increments
Unit[2] : User-defined position unit [3]
Range: Integer32
Default Value: 0
Store Supported: Yes [5]

Description: The *home offset* is the difference between the zero position for the application and the machine home position (found during homing). During homing the machine home position is found and once the homing is completed, the zero position is offset from the home position by adding the *home offset* to the home position.

Notes: *[1] for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower '*

[2] for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior

[3] See 'Factor Group' chapter

[4] The 'device internal position unit' is 'Increments'

[5] 'Store supported' is available for this object *only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r36 or superior.*

Name: **Max_Profile_Velocity**
Index.Sub: **607F.0H**
Type: Unsigned32
Access: rw
Unit[(1)] : *Inc/s*
Unit[(2)] : User-defined velocity unit [(3)]
Range[(1)]: Min = 250 Inc/s ÷ 'Max' [(9)]
Range[(2)]: Min = 250 ÷ 'Max' [(10)]
Default Value [(1)]: 'Max' [(9)]
Default Value [(2)]: 'Max' [(10)]

Store Supported: No

Description: The Max_Profile_Velocity is the maximum allowed speed during a profiled move.

Notes: *[(1)] for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower '*

[(2)] for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior

[(3)] See 'Factor Group' chapter

[(4)] only *for SW1,SM1A,DMD105 Drives*

[(5)] for CSMD1,SB4,SW4,SM4,SW5 Drives

[(6)] This object can only be set with the motor at a standstill;

[(7)] This object must be higher than the object *Min_Profile_Velocity*.

[(8)] If the value to be set is lower than the minimum value range, the minimum value is stored.

[(9)] If 'Motor_Step_Angle' value is different from 65535 :

*'Max' [(4)] = 2100rpm * [(Motor_Poles*4) * Motor_Step_Angle / 60]*

*'Max' [(5)] = 3000rpm * [(Motor_Poles*4) * Motor_Step_Angle / 60]*

If 'Motor_Step_Angle' value is equal to 65535 [(5)] :

*'Max' = 3000rpm * [65536 / 60]*

[(10)] '*Max*' *[(2)]* value is calculated according to this formula :

*Max(User-defined velocity unit) = 3276800 * (2013.2h / 2013.1h) * (60EF.0h / 65536)*

See 'Factor Group' chapter for more details.

[(11)] The 'device internal velocity unit' is 'Increments/sec'

Name: **Profile_Velocity**
Index.Sub: **6081.0H**
Type: Unsigned32
Access: rw
Unit[1] : *Inc/s*
Unit[2] : User-defined velocity unit [3]
Range: 1 ÷ 'Max_Profile_Velocity' (607F.0h)
Default Value: --
Store Supported: Yes

Description: It sets the velocity normally attained at the end of the acceleration ramp during a profile move and is valid both directions of motion.

Notes: *[1] for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower '*

[2] for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior '

[3] See 'Factor Group' chapter

[4] This object is used in 'Profile Position mode'

[5] The 'device internal velocity unit' is 'Increments/sec'

Name: **Profile_Acceleration**
Index.Sub: **6083.0H**
Type: Unsigned32
Access: rw
Unit[(1)] : *Inc/s²*
Unit[(2)] : User-defined acceleration unit [(3)]
Range: unsigned32
Default Value: 20000
Store Supported: No

Description: It indicates the configured acceleration ramp.

Notes: *[(1)] for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower '*

[(2)] for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior '

[(3)] See 'Factor Group' chapter

[(4)] The 'device internal acceleration unit' is 'Increments/sec²'

Name: **Profile_Deceleration**
Index.Sub: **6084.0H**
Type: Unsigned32
Access: rw
Unit[(1)] : *Inc/s²*
Unit[(2)] : User-defined acceleration unit [(3)]
Range: unsigned32
Default Value: 20000
Store Supported: No

Description: It indicates the configured deceleration ramp.

Notes: *[(1)] for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower '*

[(2)] for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior '

[(3)] See 'Factor Group' chapter

[(4)] The 'device internal acceleration unit' is 'Increments/sec²'

Name: **Quick_Stop_deceleration**
Index.Sub: **6085.0H**
Type: Unsigned32
Access: r/w
Unit[(1)] : *Inc/s²*
Unit[(2)] : User-defined acceleration unit [(3)]
Range: Unsigned32
Default Value: 20000
Store Supported: No

Description: It sets the deceleration ramp used to stop the motor if the 'Quick Stop' command is given and the *Quick_Stop_Option_code* is set to 2.

Notes: *[(1)] for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower '*

[(2)] for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior '

[(3)] See 'Factor Group' chapter

[(4)] The 'device internal acceleration unit' is 'Increments/sec²'

Name: **Motion_profile_type**
Index.Sub: **6086.0H**
Type: Integer16
Access: rw
Unit: --
Range: 0
Default Value: 0
Store Supported: No

Description: This object indicates the configured type of motion profile used to perform a profiled motion. The 'Slim Line Series Drives' supports only the linear ramp (trapezoidal profile) :

0 = Linear ramp, trapezoidal profile

Notes:

Name: **Position notation index**
Index.Sub: **6089.0H**
Data Type: Integer8
Access: rw
Unit: **---**
Supported value: 0
Default Value: 0
Store Supported: No

Description: This parameter defines the decimal exponent of position p.u.(10^n)

Notes: (1) This object is available only *for SW1,SM1A,DMD105 Drives*

(2) This object is not available in EtherCAT fieldbus.

Name: **Position dimension index**
Index.Sub: **608A.0H**
Data Type: Integer8
Access: rw
Unit: **---**
Supported value: AC_h (steps)
Default Value: Ac_h (steps)
Store Supported: No

Description: This parameter defines the unit of position p.u.

Notes: (1) This object is available only *for SW1,SM1A,DMD105 Drives*

(2) This object is not available in EtherCAT fieldbus.

Name: **Velocity notation index**
Index.Sub: **608B.0H**
Data Type: Integer8
Access: rw
Unit: **---**
Supported value: 0
Default Value: 0
Store Supported: No

Description: This parameter defines the decimal exponent of velocity p.u. (10^n)

Notes: *(1) This object is available only for SW1,SM1A,DMD105 Drives*

(2) This object is not available in EtherCAT fieldbus.

Name: **Velocity dimension index**
Index.Sub: **608C.0H**
Data Type: Integer8
Access: rw
Unit: **---**
Supported value: 20_h (Hz)
Default Value: 20_h (Hz)
Store Supported: No

Description: This parameter defines the unit of velocity p.u.

Notes: *(1)* This object is available only *for SW1,SM1A,DMD105 Drives*

(2) This object is not available in EtherCAT fieldbus.

Name: **Acceleration notation index**
Index.Sub: **608D.0H**
Data Type: Integer8
Access: rw
Unit: **---**
Supported value: 0
Default Value: 0
Store Supported: No

Description: This parameter defines the decimal exponent of acceleration and deceleration p.u. (10^n)

Notes: *(1) This object is available only for SW1,SM1A,DMD105 Drives*

(2) This object is not available in EtherCAT fieldbus.

Name: **Acceleration dimension index**
Index.Sub: **608E.0H**
Data Type: Integer8
Access: rw
Unit: **---**
Supported value: $B2_h$ (steps / sec^2)
Default Value: $B2_h$ (steps / sec^2)
Store Supported: No

Description: This parameter defines the unit of acceleration and deceleration p.u.

Notes: *(1)* This object is available only *for SW1,SM1A,DMD105 Drives*

(2) This object is not available in EtherCAT fieldbus.

Name: **Position Encoder Resolution (number of entries)**
Index.Sub: **608F.0H**
Data Type: Unsigned8
Access: ro
Unit: **---**
Range: 2
Default Value: 2
Store Supported: No

Description: Highest sub-index supported

Notes: See 'Factor Group' chapter for more details.

Name: **Position Encoder Resolution (Encoder Increments)**
Index.Sub: **608F.1H**
Data Type: Unsigned32
Access: rw
Unit: Increments
Range: **---**
Default Value: 1600
Store Supported: Yes

Description: The *Position Encoder Resolution* defines the ratio of encoder increments per motor revolutions :

$$Position\ Encoder\ Resolution = \frac{Encoder\ Increments\,(608\,F.1\,h)}{Motor\ Revolution\,(608\,F.2\,h)}$$

Notes: *See 'Factor Group' chapter for more details.*

Name: **Position Encoder Resolution (Motor Revolution)**
Index.Sub: **608F.2H**
Data Type: Unsigned32
Access: rw
Unit: **---**
Range: 1 - Unsigned32
Default Value: 1
Store Supported: Yes

Description: The *Position Encoder Resolution* defines the ratio of encoder increments per motor revolutions :

$$Position\ Encoder\ Resolution = \frac{Encoder\ Increments\,(608\,F.1\,h)}{Motor\ Revolution\,(608\,F.2\,h)}$$

Notes: See 'Factor Group' chapter for more details.

Name: **Gear_Ratio (Highest sub-index supported)**
Index.Sub: **6091.0H**
Type: Unsigned8
Access: ro
Unit: --
Range: 2
Default Value: 2
Store Supported: No

Description: Highest sub-index supported

Notes:

Name: **Gear_Ratio (Motor_Shaft_Revolutions)**
Index.Sub: **6091.1H**
Type: Unsigned32
Access: rw
Unit: --
Range: 1÷4294967295
Default Value: 1
Store Supported: Yes

Description: This object defines the 'Motor shaft revolutions' used to calculate the Gear ratio between Motor Shaft Revolutions and Driving Shaft revolutions :

$$Gear\ Ratio = \frac{Motor\ Shaft\ Revolutions\,(6091.1\,h)}{Driving\ Shaft\ Revolutions\,(6091.2\,h)}$$

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior.

See 'Factor Group' chapter for more details.

Name: **Gear_Ratio (Driving_Shaft_Revolutions)**
Index.Sub: **6091.2H**
Type: Unsigned32
Access: rw
Unit: **--**
Range: 1÷4294967295
Default Value: 1
Store Supported: Yes

Description: This object defines the 'Driving shaft revolutions' used to calculate the Gear ratio between Motor Shaft Revolutions and Driving Shaft revolutions :

$$Gear\ Ratio = \frac{Motor\ Shaft\ Revolutions\,(6091.1\,h)}{Driving\ Shaft\ Revolutions\,(6091.2\,h)}$$

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior.

See 'Factor Group' chapter for more details.

Name: **Feed_Constant (Highest sub-index supported)**
Index.Sub: **6092.0H**
Type: Unsigned8
Access: ro
Unit: --
Range: 2
Default Value: 2
Store Supported: No

Description: Highest sub-index supported

Notes:

Name: **Feed_Constant (Feed)**
Index.Sub: **6092.1H**
Type: Unsigned32
Access: rw
Unit: User-defined position unit
Range: 1÷4294967295
Default Value: 200
Store Supported: Yes

Description: This object is used to calculate the ratio of Feed (given in user-defined position units) per Driving Shaft revolutions :

$$Feed\,Constant = \frac{Feed\,(6092.1\,h)}{Driving\,Shaft\;Revolutions\,(6092.2\,h)}$$

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior.

See ‘Factor Group’ chapter for more details.

Name: **Feed_Constant (Driving_Shaft_Revolutions)**
Index.Sub: **6092.2H**
Type: Unsigned32
Access: rw
Unit: **--**
Range: 1÷4294967295
Default Value: 1
Store Supported: Yes

Description: This object is used to calculate the ratio of Feed (given in user-defined position units) per Driving Shaft revolutions :

$$Feed\,Constant = \frac{Feed\,(6092.1\,h)}{Driving\,Shaft\;Revolutions\,(6092.2\,h)}$$

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior.

See ‘Factor Group’ chapter for more details.

Name: **Homing_method**
Index.Sub: **6098.0H**
Type: Integer8
Access: rw
Unit: --
Range: 1,2,17,18,19,20,21,22,35
Default Value: 19
Store Supported: Yes (1)

Description: It indicates the Homing method type used during the homing procedure. The Homing methods supported are :

Notes: (1) 'Store supported' is available for this object *only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r36 or superior.*

Name: **Homing speeds (number of entries)**
Index.Sub: **6099.0H**
Data Type: Unsigned8
Access: ro
Unit: **---**
Range: 2
Default Value: 2
Store Supported: No

Description: Highest sub-index supported

Notes:

Name: **Homing speeds (Speed during search for switch)**
Index.Sub: **6099.1H**
Data Type: Unsigned32
Access: rw
Unit[(1)] : *Inc/s*
Unit[(2)] : User-defined velocity unit [(3)]
Range: 1 ÷ 'Max_Profile_Velocity' (607F.0h)
Default Value: 10000
Store Supported: Yes [(5)]

Description: Speed during search for switch

Notes: *[(1)] for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower '*

[(2)] for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior '

[(3)] See 'Factor Group' chapter

[(4)] The 'device internal velocity unit' is 'Increments/sec'

[(5)] 'Store supported' is available for this object *only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r36 or superior.*

Name: **Homing speeds (Speed during search for zero)**
Index.Sub: **6099.2H**
Data Type: Unsigned32
Access: rw
Unit[(1)] : *Inc/s*
Unit[(2)] : User-defined velocity unit [(3)]
Range: 1 ÷ 'Max_Profile_Velocity' (607F.0h)
Default Value: 250
Store Supported: Yes [(5)]

Description: Speed during search for zero

Notes: *[(1)] for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower '*

[(2)] for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior '

[(3)] See 'Factor Group' chapter

[(4)] The 'device internal velocity unit' is 'Increments/sec'

[(5)] 'Store supported' is available for this object only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r36 or superior.

Name: **Homing acceleration**
Index.Sub: **609A.0H**
Type: Unsigned32
Access: rw
Unit[(1)] : *Inc/s²*
Unit[(2)] : User-defined acceleration unit [(3)]
Range: unsigned32
Default Value: 20000
Store Supported: Yes [(5)]

Description: It sets the acceleration and deceleration for homing movements.

Notes: *[(1)] for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower '*

[(2)] for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior '

[(3)] See 'Factor Group' chapter

[(4)] The 'device internal acceleration unit' is 'Increments/sec²'

[(5)] 'Store supported' is available for this object only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r36 or superior.

Name: **Touch_Probe_Function**
Index.Sub: **60B8.0H**
Type: Unsigned16
Access: rw
Unit: --
Range: –
Default Value: 0000H
Store Supported: No

Description: This object indicates the configured function of the Touch Probe

Bit	Value	Description
0	0	Switch off Touch Probe1
	1	Enable Touch Probe1
1	0	Trigger first event
	1	Continuos
3,2	00b	Trigger with Touch Probe 1 Input (B0_IN0)
	01b	Trigger with zero impulse signal of position encoder
	10b	Touch Probe Source defined by object 60D0.1h
	11b	
4	0	Switch off sampling at positive edge of Touch Probe 1
	1	Enable sampling at positive edge of Touch Probe 1
5	0	Switch off sampling at negative edge of Touch Probe 1
	1	Enable sampling at negative edge of Touch Probe 1
6,7	--	User defined
8	0	Switch off Touch Probe2
	1	Enable Touch Probe2
9	0	Trigger first event
	1	Continuos
11,10	00b	Trigger with Touch Probe 2 Input (B0_IN1)
	01b	Trigger with zero impulse signal of position encoder
	10b	Touch Probe Source defined by object 60D0.2h
	11b	
12	0	Switch off sampling at positive edge of Touch Probe 2
	1	Enable sampling at positive edge of Touch Probe 2
13	0	Switch off sampling at negative edge of Touch Probe 2
	1	Enable sampling at negative edge of Touch Probe 2
14,15	--	User defined

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r11 or superior.

Name: **Touch_Probe_Status**
Index.Sub: **60B9.0H**
Type: Unsigned16
Access: ro
Unit: --
Range: –
Default Value: 0000H
Store Supported: No

Description: This object provides the status of the Touch Probe.

Bit	Value	Description
0	0	Touch Probe1 is switched off
	1	Touch Probe1 is enabled
1	0	Touch Probe1 no positive edge value stored
	1	Touch Probe1 positive edge position stored
2	0	Touch Probe1 no negative edge value stored
	1	Touch Probe1 negative edge position stored
3-5	0	Reserved
6	0	User defined
7	0	Touch Probe1 Input = low level
	1	Touch Probe1 Input = high level
8	0	Touch Probe2 is switched off
	1	Touch Probe2 is enabled
9	0	Touch Probe2 no positive edge value stored
	1	Touch Probe2 positive edge position stored
10	0	Touch Probe2 no negative edge value stored
	1	Touch Probe2 negative edge position stored
11-13	0	Reserved
14	0	User defined
15	0	Touch Probe2 Input = low level
	1	Touch Probe2 Input = high level

- Bit1 and bit2 are set to 0 when touch probe1 is switched off (object 60B8 bit0 is 0).
- Bit9 and bit10 are set to 0 when touch probe2 is switched off (object 60B8 bit8 is 0).

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r11 or superior.

Name: **Touch_Probe1_Positive_Edge_Position**
Index.Sub: **60BA.0H**
Type: Integer32
Access: ro
Unit: User-defined position unit [2]
Range: Integer32
Default Value: 0000H
Store Supported: No

Description: This object provides the position value of the Touch Probe1 at positive edge.

Notes: [1] This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r11 or superior.

[2] *See 'Factor Group' chapter*

[3] The 'device internal position unit' is 'Increments'

Name: **Touch_Probe1_Negative_Edge_Position**
Index.Sub: **60BB.0H**
Type: Integer32
Access: ro
Unit: User-defined position unit [2]
Range: Integer32
Default Value: 0000H
Store Supported: No

Description: This object provides the position value of the Touch Probe1 at negative edge.

Notes: [1] This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r11 or superior.

[2] *See 'Factor Group' chapter*

[3] The 'device internal position unit' is 'Increments'

Name: **Touch_Probe2_Positive_Edge_Position**
Index.Sub: **60BC.0H**
Type: Integer32
Access: ro
Unit: User-defined position unit [2]
Range: Integer32
Default Value: 0000H
Store Supported: No

Description: This object provides the position value of the Touch Probe2 at positive edge.

Notes: [1] This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r11 or superior.

[2] *See 'Factor Group' chapter*

[3] The 'device internal position unit' is 'Increments'

Name: **Touch_Probe2_Negative_Edge_Position**
Index.Sub: **60BD.0H**
Type: Integer32
Access: ro
Unit: User-defined position unit [2]
Range: Integer32
Default Value: 0000H
Store Supported: No

Description: This object provides the position value of the Touch Probe2 at negative edge.

Notes: [1] This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r11 or superior.

[2] *See 'Factor Group' chapter*

[3] The 'device internal position unit' is 'Increments'

Name: **Touch_Probe_Source**
Index.Sub: **60D0.0H**
Type: Unsigned8
Access: ro
Unit: --
Range: 1÷254
Default Value: 2
Store Supported: No

Description: This object provides the number of Touch Probe Sources supported

Notes: [1] This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r11 or superior.

Name: **Touch_Probe1_Source**
Index.Sub: **60D0.1H**
Type: Integer16
Access: rw
Unit: --
Range: see table below
Default Value: -1
Store Supported: Yes [2]

Description: This object provides the source of the Touch Probe 1 functions.

Value	Description
-8	B0_IN7 (Digital Input 7 Bank0)
-7	B0_IN6 (Digital Input 6 Bank0)
-6	B0_IN5 (Digital Input 5 Bank0)
-4	B0_IN3 (Digital Input 3 Bank0)
-3	B0_IN2 (Digital Input 2 Bank0)
-2	B0_IN1 (Digital Input 1 Bank0)
-1	B0_IN0 (Digital Input 0 Bank0)
+5	Hardware Zero impulse signal of position encoder

Notes: [1] This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r11 or superior.

[2] 'Store supported' is available for this object *only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r36 or superior.*

Name: **Touch_Probe2_Source**
Index.Sub: **60D0.2H**
Type: Integer16
Access: rw
Unit: --
Range: see table below
Default Value: -2
Store Supported: Yes [2]

Description: This object provides the source of the Touch Probe 2 functions.

Value	Description
-8	B0_IN7 (Digital Input 7 Bank0)
-7	B0_IN6 (Digital Input 6 Bank0)
-6	B0_IN5 (Digital Input 5 Bank0)
-4	B0_IN3 (Digital Input 3 Bank0)
-3	B0_IN2 (Digital Input 2 Bank0)
-2	B0_IN1 (Digital Input 1 Bank0)
-1	B0_IN0 (Digital Input 0 Bank0)
+5	Hardware Zero impulse signal of position encoder

Notes: [1] This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r11 or superior.

[2] 'Store supported' is available for this object *only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r36 or superior.*

Name: **Touch_Probe1_Positive_Edge_Counter**
Index.Sub: **60D5.0H**
Type: Unsigned16
Access: ro
Unit: --
Range: Unsigned16
Default Value: 0
Store Supported: No

Description: This object provides a continuous counter that is incremented with each positive edge at touch probe 1. The counter is only valid if Touch probe input is enabled (60B8.0h , Bit0=1) and continuous touch probe mode is enabled (60B8.0h, Bit1 =1).

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r11 or superior.

Name: **Touch_Probe1_Negative_Edge_Counter**
Index.Sub: **60D6.0H**
Type: Unsigned16
Access: ro
Unit: --
Range: Unsigned16
Default Value: 0
Store Supported: No

Description: This object provides a continuous counter that is incremented with each negative edge at touch probe 1. The counter is only valid if Touch probe input is enabled (60B8.0h , Bit0=1) and continuous touch probe mode is enabled (60B8.0h, Bit1 =1).

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r11 or superior.

Name: **Touch_Probe2_Positive_Edge_Counter**
Index.Sub: **60D7.0H**
Type: Unsigned16
Access: ro
Unit: --
Range: Unsigned16
Default Value: 0
Store Supported: No

Description: This object provides a continuous counter that is incremented with each positive edge at touch probe 2. The counter is only valid if Touch probe input is enabled (60B8.0h , Bit8=1) and continuous touch probe mode is enabled (60B8.0h, Bit9 =1).

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r11 or superior.

Name: **Touch_Probe2_Negative_Edge_Counter**
Index.Sub: **60D8.0H**
Type: Unsigned16
Access: ro
Unit: --
Range: Unsigned16
Default Value: 0
Store Supported: No

Description: This object provides a continuous counter that is incremented with each negative edge at touch probe 2. The counter is only valid if Touch probe input is enabled (60B8.0h , Bit8=1) and continuous touch probe mode is enabled (60B8.0h, Bit9 =1).

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r11 or superior.

Name: **Interpolation data record (number of entries)**
Index.Sub: **60C1.0H**
Data Type: Unsigned8
Access: ro
Unit: **---**
Range: 1
Default Value: 1
Store Supported: No

Description: Number of entries

Notes: See 60C1.1h

Name: **Interpolation data record (position set-point)**
Index.Sub: **60C1.1H**
Data Type: Integer32
Access: rw
Unit[(1)] : Increments
Unit[(2)] : User-defined position unit [(3)]
Range: -2147483648 ÷ 2147483647
Default Value: ---
Store Supported: No

Description: Defines the position set-point used for linear interpolation.

Notes: *[(1)] for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower '*

[(2)] for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior '

[(3)] See 'Factor Group' chapter

[(4)] See 60C1.0h

[(5)] The 'device internal position unit' is 'Increments'

Name: **Interpolation time period (number of entries)**
Index.Sub: **60C2.0H**
Data Type: Unsigned8
Access: ro
Unit: ---
Range: 2
Default Value: 2
Store Supported: No

Description: Number of entries

Notes: (1) See 60C2.1h, 60C2.2h

Name: **Interpolation time period (interpolation time units)**
Index.Sub: **60C2.1H**
Data Type: Unsigned8
Access: rw
Unit: ---
Range: ---
Default Value: 1
Store Supported: Yes (2)

Description: Defines a parameter used for interpolation period computation :
Interpolation time period = Interpolation_time_units * 10 $^{Interpolation\_time\_index}$ seconds

Notes: (1) See 60C2.2h

(2) 'Store supported' is available for this object *only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r36 or superior.*

Name: **Interpolation time period (interpolation time index)**
Index.Sub: **60C2.2H**
Data Type: Signed8
Access: rw
Unit: ---
Range: only -3
Default Value: -3 (see Appendix B)
Store Supported: No

Description: Defines a parameter used for interpolation period computation :
Interpolation time period = Interpolation_time_units * 10 $^{Interpolation\_time\_index}$ seconds

Notes: See 60C2.1h

Name: **Interpolation sync definition (number of entries)**
Index.Sub: **60C3.0H**
Data Type: Unsigned8
Access: ro
Unit: ---
Range: 2
Default Value: 2
Store Supported: No

Description: Number of entries

Notes: This object is not available in EtherCAT fieldbus.

Name: **Interpolation sync definition (synchronize on group)**
Index.Sub: **60C3.1H**
Data Type: Unsigned8
Access: rw
Unit: ---
Range: 0 ÷ 255
Default Value: 0
Store Supported: Yes [(2)]

Description: Defines the interpolation method.

0 = SYNC
1...255 = reserved (not implemented)

Notes: [(1)] This object is not available in EtherCAT fieldbus.

[(2)] 'Store supported' is available for this object *only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r36 or superior.*

Name: **Interpolation sync definition (ip sync every n event)**
Index.Sub: **60C3.2H**
Data Type: Unsigned8
Access: rw
Unit: ---
Range: 0 ÷ 255
Default Value: 1
Store Supported: Yes [(2)]

Description: Defines the SYNC number after that the position set-point used for interpolation is carried out.

Notes: [(1)] This object is not available in EtherCAT fieldbus.

[(2)] 'Store supported' is available for this object *only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r36 or superior.*

Name: **Motor_Resolution**
Index.Sub: **60EF.0H**
Data Type: Unsigned32
Access: ro
Unit: Increments
Range: 200 ÷ 65536
Default Value: 200
Store Supported: No

Description: This objects returns the Motor Resolution in Inc/revolution :

- If 'Motor_Step_Angle' value is different from 65535, the number of motor Increments per revolution are computed as follow :

  *Motor_Resolution = (Motor_Poles*4) * Motor_Step_Angle;*

  If 'Motor_Poles' = 50 (standard typical value) :

Motor_Step_Angle = 1	→	Motor_Resolution = 200	Inc/rev
Motor_Step_Angle = 2	→	Motor_Resolution = 400	Inc/rev
Motor_Step_Angle = 4	→	Motor_Resolution = 800	Inc/rev
Motor_Step_Angle = 8	→	Motor_Resolution = 1600	Inc/rev
Motor_Step_Angle = 16	→	Motor_Resolution = 3200	Inc/rev
Motor_Step_Angle = 32	→	Motor_Resolution = 6400	Inc/rev
Motor_Step_Angle = 64	→	Motor_Resolution = 12800	Inc/rev
Motor_Step_Angle = 128	→	Motor_Resolution = 25600	Inc/rev

- If 'Motor_Step_Angle' value is equal to -1 , the number of motor Increments per revolution is :

Motor_Step_Angle = -1	→	Motor_Resolution = 65536	Inc/rev

Notes: This object is available only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior.

See 'Factor Group' chapter for more details.

Name: **Following_error_actual_value**
Index.Sub: **60F4.0H**
Type: Integer32
Access: ro
Unit[1] : Increments
Unit[2] : User-defined position unit [3]
Range: Integer32
Default Value: --
Store Supported: No

Description: It represents the actual value of the following error.

For 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r17 or superior ', when Feedback Feature is enabled and Sensor_Selectio_Code different than -2, it returns the value of object Feedback_Actual_Displacement converted in user defined position unit.

Notes: *[1] for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower '*

[2] for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior '

[3] See 'Factor Group' chapter

[4] The 'device internal position unit' is 'Increments'

Name: **Digital_Inputs**
Index.Sub: **60FD.0H**
Type: Unsigned32
Access: ro
Unit: **--**
Range: Unsigned32
Default Value: --
Store Supported: No

Description: It contains the status of Digital inputs of the drive :

Bit#	*Description*	
0	***Negative limit switch***	Negative Limit switch
1	***Positive limit switch***	Positive Limit switch
2	***Home switch***	Home switch
3	Not implemented	Interlock
4		reserved
5		reserved
6		reserved
7		reserved
8		reserved
9		reserved
10		reserved
11		reserved
12		reserved
13		reserved
14		reserved
15		reserved
16	***B0_IN0*** [(3)]	Manufacturer specific
17	***B0_IN1*** [(3)]	Manufacturer specific
18	***B0_IN2*** [(3)]	Manufacturer specific
19	***B0_IN3*** [(3)]	Manufacturer specific
20	***B0_IN4*** [(3)]	Manufacturer specific
21	***B0_IN5*** [(3)]	Manufacturer specific
22	***B0_IN6*** [(3)]	Manufacturer specific
23	***B0_IN7*** [(3)]	Manufacturer specific
24		Manufacturer specific
25		Manufacturer specific
26		Manufacturer specific
27		Manufacturer specific
28		Manufacturer specific
29		Manufacturer specific
30		Manufacturer specific
31		Manufacturer specific

Notes:
[(1)] See D Appendix.
[(2)] See 2081.0h object.
[(3)] Available *only for CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V01r36 or superior.*

Name: **Target_Velocity**
Index.Sub: **60FF.0H**
Type: Integer32
Access: rw
Unit[(1)] : *Inc/s*
Unit[(2)] : User-defined velocity unit *(3)*
Range: 0 ÷ 'Max_Profile_Velocity' (607F.0h)
Default Value: 0
Store Supported: No

Description: This object indicates the configured target velocity.

Notes:

(1) for 'SW1,SM1A,DMD105 Drives' and for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r79 or lower '

(2) for 'CSMD1,SB4,SW4,SM4,SW5 Drives with firmware V00r80 or superior '

(3) See 'Factor Group' chapter

(4) This object is used in 'Profile Velocity mode (pv)' and 'Cyclic Synchronous Velocity mode (csv)'

(5) The 'device internal velocity unit' is 'Increments/sec'

Name: **B1_Digital_Inputs**
Index.Sub: **6100.2H**
Type: Unsigned8
Access: ro
Unit: **--**
Range: 0 (all inputs are open) ÷ 2^n-1 (all inputs are closed, where n is the number of digital inputs available)
Default Value: --
Store Supported: No

Description: It contains the status of all the inputs on bank 1 of the drive.

Notes: See D Appendix.
A numeric value is associated to the input, following the procedure laid out hereafter:

INPUTS	VALUE
Input 0	1
Input 1	2
Input 2	4
Input 3	8
Input 4	16
Input 5	32
Input n	2^n

Name: **B0_Digital_Outputs**
Index.Sub: **6200.1H**
Type: Unsigned8
Access: rw
Unit: **--**
Range: 0 (all outputs are open) ÷ 2^n-1 (all outputs are closed, where n is the number of digital outputs available)
Default Value: --
Store Supported: No

Description: It contains the status of all the outputs on bank 0 of the drive.

Notes: A numeric value is associated to the outputs, following the procedure laid out hereafter:

OUTPUTS	VALUE
Output 0	1
Output 1	2
Output 2	4
Output 3	8
Output 4	16
Output 5	32
Output n	2^n

Name: **B1_Digital_Outputs**
Index.Sub: **6300.2H**
Type: Unsigned8
Access: rw
Unit: **--**
Range: 0 (all outputs are open) ÷ 2^n-1 (all outputs are closed, where n is the number of digital outputs available)
Default Value: --
Store Supported: No

Description: It contains the status of all the outputs on bank 1 of the drive.

Notes: A numeric value is associated to the outputs, following the procedure laid out hereafter:

OUTPUTS	VALUE
Output 0	1
Output 1	2
Output 2	4
Output 3	8
Output 4	16
Output 5	32
Output n	2^n

Name: **Analog_In[0÷1]**
Index.Sub: **6404.1H,6404.2H**
Type: Integer16
Access: ro
Unit: mV
Range: -10000 ÷ 10000
Default Value: --
Store Supported: No

Description: It contains the value of the drive's analog input 0 and 1.

Notes: The number of available analog inputs depends on the version of the drive currently in use.

Name: **Analog_Out[0÷1]**
Index.Sub: **6414.1H,6414.2H**
Type: Unsigned16
Access: rw
Unit: mV
Range: 0 ÷ 10000
Default Value: 0
Store Supported: No

Description: This object is used to set/read the value of the analog outputs.

Notes: The number of available analog outputs depends on the version of the drive currently in use. The build-in converter (12 bit) has a resolution of 2.5mV.

Name: **Supported_drive_modes**
Index.Sub: **6502.0H**
Data Type: Unsigned32
Access: ro
Unit: ---
Range: Unsigned32
Default Value: 485 (1E5H)
Store Supported: No

Description: It provides information on the supported drive modes :

31		16	15		9	8	7	6	5	4	3	2	1	0
Manufacturer specific			reserved			**csv**	**csp**	**ip**	**hm**	reserved	tq	**pv**	**vl**	**pp**

Notes:

5.0 EtherCAT

The drives equipped with EtherCAT fieldbus that support CiA 402 Device Profile have a configuration code of C680. They do not need to be configured as NodeId and BaudRate (§2.2 & §2.3). The protocols supported are: CoE (CANopen over EtherCAT) and FoE (File Access over EtherCAT). The EVER drives supports different types of synchronization: Free Run, Synchronous with SM Event, Distributed Clocks. The services EMCY (§2.8) and Diagnostics are supported too.

5.1 STATE LED

Near the EtherCAT Connector A there is a STATE LED that can be in one of the following situation:

State	Slave Condition	Communication
Off	Init	After switch-on the EtherCAT slave is in the *Init* state. No SDO or PDO communication is possible.
Blinking	Pre-Operational	In *Pre-Operational* state SDO communication is possible, but not PDO communication.
Single Flash	Safe-Operational	In *Safe-Operational* state SDO and PDO communication is possible, although the slave keeps its outputs (RPDO) in a safe state, while the input (TPDO) data are updated cyclically.
On	Operational	In the *Operational* state the slave copies the output data of the masters to its outputs (RPDO). PDO and SDO communication is possible.

5.2 PDO Mapping

The PDO mapping for EtherCAT devices is fixed and contains the whole objects necessary to all supported modes:

SlimLine Drives:

Direction	Description
RPDO1	Controlword (0x6040), Modes_of_Operation (0x6060), B0_Digital_Outputs (0x6200), Target_Position (0x607A)
RPDO2	Interpolation_Data_x1 (0x60C1.1), Target_Velocity (0x60FF)
TPDO1	Statusword (0x6041), Modes_of_Operation_Display (0x6061), B0_Digital_Inputs (0x6000.1), Position_Actual_Value (0x6064)
TPDO2	Velocity_Actual_Value (0x606C), Following_Error_Actual_Value (0x60F4)

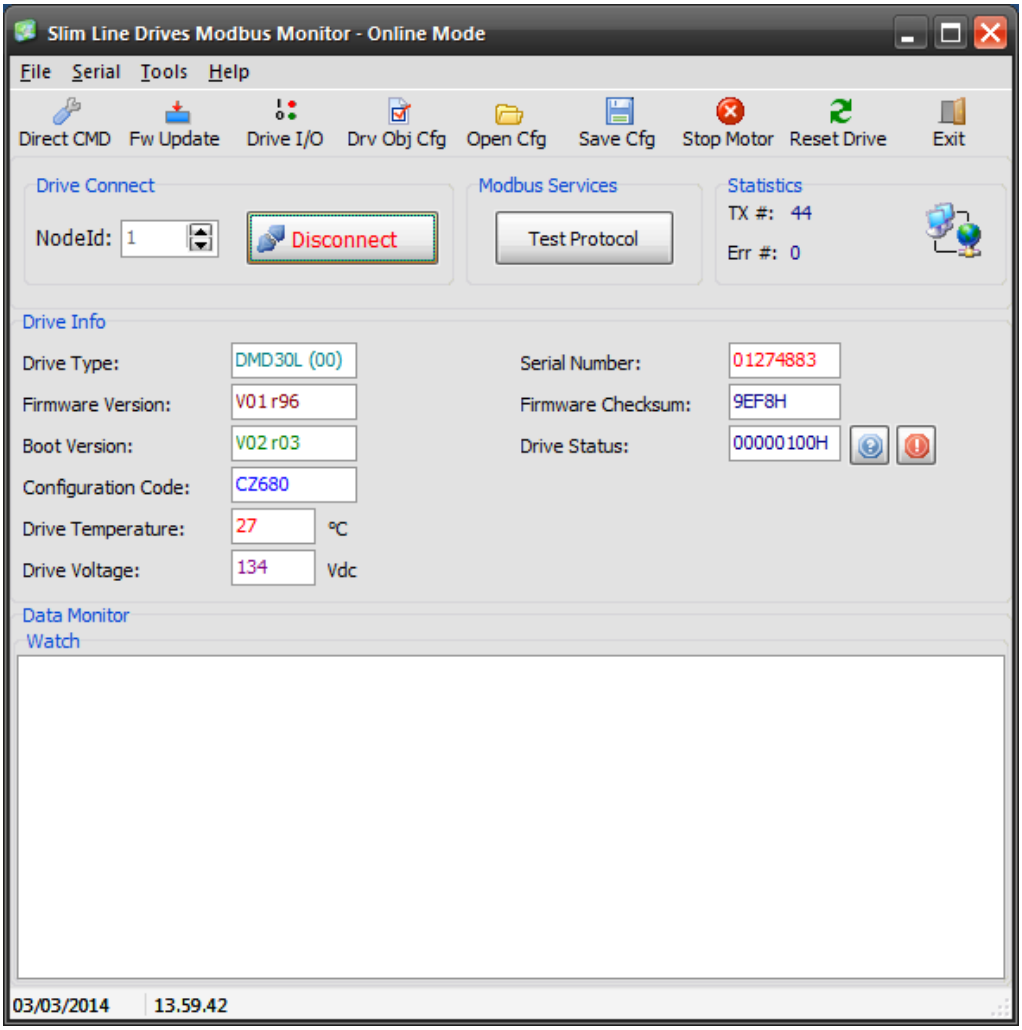
Titanio Drives:

Direction	Description
Mapping 1 RPDO (0x1600)	Controlword (0x6040), Modes_of_Operation (0x6060), B0_Digital_Outputs (0x6200), Target_Position (0x607A), Interpolation_Data_x1 (0x60C1.1), Target_Velocity (0x60FF)
Mapping 2 RPDO (0x1601)	Controlword (0x6040), Modes_of_Operation (0x6060), B0_Digital_Outputs (0x6200), Target_Position (0x607A), Interpolation_Data_x1 (0x60C1.1), Target_Velocity (0x60FF), Touch probe function (0x60B8)
Mapping 1 TPDO (0x1A00)	Statusword (0x6041), Modes_of_Operation_Display (0x6061), B0_Digital_Inputs (0x6000.1), Position_Actual_Value (0x6064),Velocity_Actual_Value (0x606C), Following_Error_Actual_Value (0x60F4)
Mapping 2 TPDO (0x1A01)	Statusword (0x6041), Modes_of_Operation_Display (0x6061), B0_Digital_Inputs (0x6000.1), Position_Actual_Value (0x6064),Velocity_Actual_Value (0x606C), Following_Error_Actual_Value (0x60F4), Touch probe 1 positive edge (0x60BA), Touch probe 2 positive edge (0x60BC), Touch probe status (0x60B9), Error Code (0x603F)

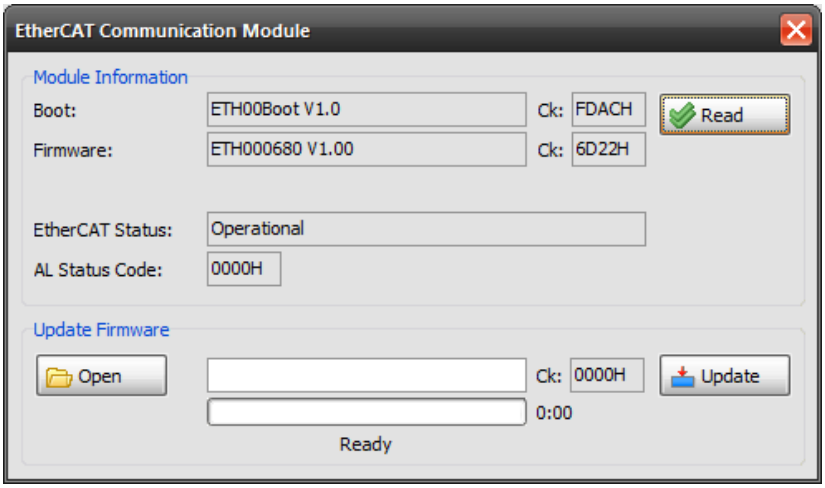
To change the Mapping it is necessary to set the objects 1C12.1 and 1C13.1 while in Pre-Operational condition.

5.3 PC Support Tool

It is possible to check/update the firmware of the EtherCAT Board using the *'Slim Line Modbus Monitor'* and connecting to the service RS232 (or Service Serial Interface for Titanio Drives with a baud rate of 115200) provided by the drive. From the main window it is possible to access to settings of the power board of the drive and update the firmware if necessary.



Under menu *'Tools/EtherCAT Comm. Module'* it is possible to see information of the EtherCAT communication board of the drive and update the firmware if necessary.



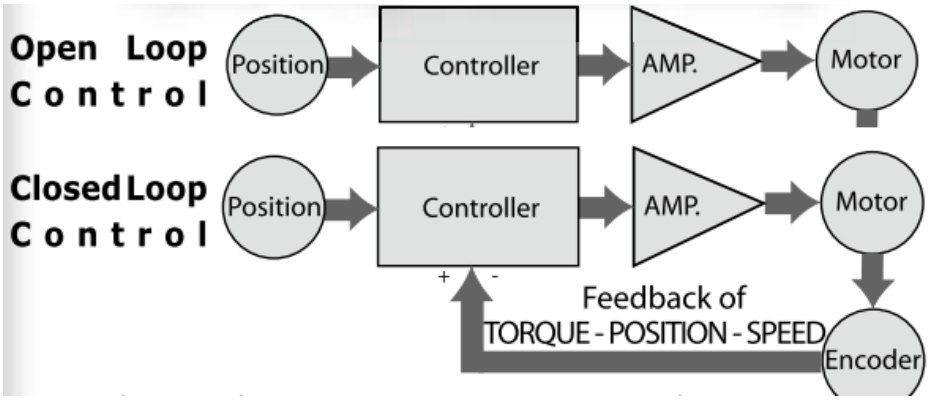
5.4 EtherCAT Slave Information (ESI)

The EtherCAT Slave Information file (XML file) is available for configuring the EtherCAT master. It can be found on the CD-ROM supplied with the drive software kit under the \XML directory.

6.0 Feedback Feature (Torque-Position-Speed Closed loop)

The motor feedback feature permits to grant the perfect synchronization between motor rotor and stator (the motor doesn't lose steps) and optimize the motor efficiency since only the really needed current is supplied. For the motor feedback feature is necessary to have an incremental encoder directly mounted on the rear shaft of a double shaft motor and connected to (ENC#0) on the drive digital inputs.

This feature is available only for CSMD1,SB4,SW4,SM4,SW5 Drives.



With regard to an 'Open Loop Stepper solution' :

- Reliable positioning without synchronization loss;
- Keeps the original position stable and recovers it automatically in case of positioning errors caused by external factors such as mechanical vibrations;
- 100% use of the motor torque;
- Capacity to operate at high velocity related to the current control, which is adjusted depending on the load variations, where the normal systems in open loop use a constant current control at all velocities without considering the load variations.

The Feedback feature is enabled and disabled by 'Feedback_Motor_Check' bit of 'Drive_Working_Settings' object (2200.2h). Before enabling the Feedback feature the Feedback objects have to be defined :

- (2230.1h) 'Feedback_Max_Displacement ' → Maximum displacement allowed
- (2230.3h) 'Feedback_Source_PPR' → Resolution of the incremental encoder
- (2230.12h) 'Feedback_Boost_Current' → Maximum current supplied to the motor
- (2230.0Fh) 'Feedback_Iq_Min' → Minimum current of the feedback regulator
- (2230.15h) 'Feedback_Kfbw_Acc' → Acceleration constant for backward movements
- (2230.16h) 'Feedback_Kfbw_Dec' → Deceleration constant for backward movements
- (2230.06h) 'Feedback_Kffw_Acc' → Acceleration constant for forward movements
- (2230.07h) 'Feedback_Kffw_Dec' → Deceleration constant for forward movements
- (2230.10h) 'Feedback_Ka' → Acceleration proportional constant
- (2230.0Dh) 'Feedback_Kalfas' → Phase advanced constant
- (2230.0Bh) 'Feedback_Ki' → Integral constant
- (2230.0Ch) 'Feedback_Ki_Limit' → Limiter of integral correction
- (2230.04h) 'Feedback_Kp' → Position proportional constant
- (2230.05h) 'Feedback_Kv' → Velocity proportional constant

The 'Feedback_actual_displacement' object returns the actual displacement between motor and encoder.

When the feedback feature is enabled (bit ‘Feedback_Motor_Check’ of ‘Drive_Working_Settings’ object equal to 1), the position and velocity actual values are referred to the incremental encoder (ENC#0), even if the object ‘Sensor_Selection_Code’ (606A.0h) is equal to -1.

7.0 Motor R,L detection

The ‘CSMD1,SB4,SW4,SM4,SW5’ Drives implement the procedure to detect the Motor R and Motor L of the motor. For this procedure are required the ‘Nominal_Current’ (2005.4h) of the motor and bit9 of ‘Drive_Working_Settings_Extend’ object sets to 1. At the end of procedure, the Motor R,L detected are stored into object 2005.7h (Motor_L_detected) and 2005.8h (Motor_R_detected).
If bit9 of ‘Drive_Working_Settings_Extend’ object is equal to 0 then Motor R,L are not automatically detected and their values must be set by mean of object 2005.6h (Motor_R) and object 2005.7h (Motor_L).

The Motor R,L detecting procedure and related objects are not implemented on SW1,SM1A,DMD105 Drives because these drives do not need of these information for motor current regulation.

A Appendix: Dimension Index Table

Physical dimension	Unit	Dimension Index
None	-	00h
Way / length	m	01h
Area	m^2	A0h
Volume	m^3	A1h
Time	s	A2h
	min	47h
	h	48h
	d	49h
	y	4Ah
Power	W	24h
Revolutions / time	rev / s	A3h
	rev / min	A4h
	rev / h	A5h
Angle	rad	10h
	s	43h
	m	42h
	°	41h
Velocity	m / s	A6h
	m / min	A7h
	m / h	A8h
Torque	N / m	A9h
Temperature	K	05h
	°C	2Dh
	F	AAh
Voltage	V	26h
Current	A	04h
Ratio	%	ABh
Frequency	Hz	20h
Steps	steps	ACh
Steps / revolution	Steps / rev	ADh

B Appendix: Notation Index Table

Prefix	Factor	Symbol	Notation Index
reserved	-	-	13h ... 7Fh
exa	10^{18}	E	12h
-	10^{17}	-	11h
-	10^{16}	-	10h
peta	10^{15}	P	0Fh
-	10^{14}	-	0Eh
-	10^{13}	-	0Dh
tera	10^{12}	T	0Ch
-	10^{11}	-	0Bh
-	10^{10}	-	0Ah
giga	10^{9}	G	09h
-	10^{8}	-	08h
-	10^{7}	-	07h
mega	10^{6}	M	06h
-	10^{5}	-	05h
-	10^{4}	-	04h
kilo	10^{3}	k	03h
hecto	10^{2}	h	02h
deca	10^{1}	da	01h
-	10^{0}	-	00h
deci	10^{-1}	d	FFh
centi	10^{-2}	c	FEh
milli	10^{-3}	m	FDh
-	10^{-4}	-	FCh
-	10^{-5}	-	FBh
micro	10^{-6}	µ	FAh
-	10^{-7}	-	F9h
-	10^{-8}	-	F8h
nano	10^{-9}	n	F7h
-	10^{-10}	-	F6h
-	10^{-11}	-	F5h
pico	10^{-12}	p	F4h
-	10^{-13}	-	F3h
-	10^{-14}	-	F2h
femto	10^{-15}	f	F1h
-	10^{-16}	-	F0h
-	10^{-17}	-	EFh
atto	10^{-18}	a	EEh
reserved	-	-	EDh … 80h

C Appendix: Slim Line Drives CAN Monitor Software

The Slim Line Drives CAN Monitor software is a diagnostic tool for Slim Line and Titanio family drives.
It is possible to read/change drive's parameters, start / stop the motor, update the firmware.

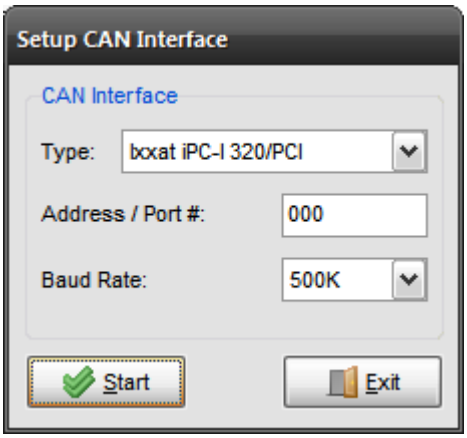
The Slim Line Drives CAN Monitor has the following hardware/software system requirements:

- ➔ **CPU:** Pentium® Class, minimum 1Ghz system clock.
- ➔ **RAM:** 1GB or more depending on OS requirements.
- ➔ **HD:** 20MB free
- ➔ **VIDEO:** minimum 1024x768 64k colors with small fonts.
- ➔ **CAN:** One of the following interfaces:
  - EVER CANUSB-01
  - IXXAT CANdy
  - IXXAT CANdy Lite
  - IXXAT TinCAN
  - IXXAT iPC-I 320/PCI
  - IXXAT USB2CAN compact
  - SiE CAN LPT
  - SiE CAN As
  - SiE CAN USB
  - GC USBCAN-I
- ➔ **OS:** Windows XP/Vista/7/8/10.

For reading the documentation the Adobe Acrobat® Reader 7.0 or superior is required.
To use IXXAT CAN interfaces, IXAAT VCI drivers 2.18 or superior installation is required.
The driver to use the EVER CANSUB-01 interface is located on the CD under EVER_CAN_DRIVER directory. The driver to use the GC USBCAN-I is located on the CD under GC_CAN directory.

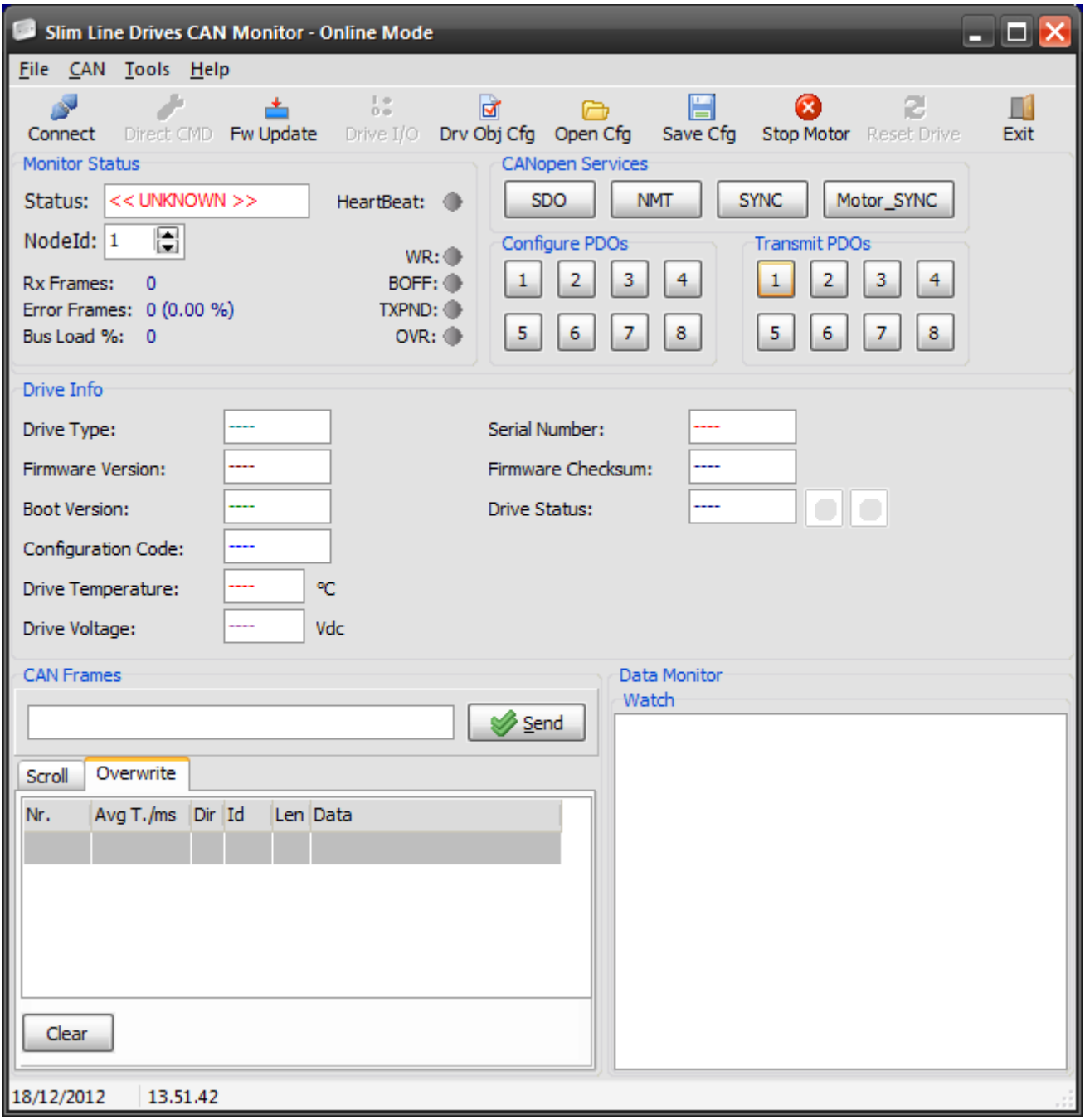
After had installed the Slim Line Drives Monitor tool, to start it simply double click on its icon:

After few seconds appears a communication setup window:



After having selected the right CAN interface, Address and Baud Rate, click on Start button.
PS: The Address/Port # can be either 1 , 2 (LPT1, LPT2, PCMCIA Slot1, Slot2) or the real I/O address (378 for LPT1, 278 for LPT2) for the USB adapters this parameter is not used.

Then the main window will appear:



In the *'Monitor Status'* frame it is possible to change the Node Id of the drive we want to address.
If the drive is switched on, pressing **Connect** button starts the polling of drive information showed on *'Drive Info'* frame. If there are drive's internal messages they will be printed on *'Message Monitor'* frame.

P.S. : Since the Slim Line Drives Monitor uses the server SDO #1 of Slim Line drive to establish the communication, it is important that no other system in the network (CAN Masters, PCs, PLCs…) is using that communication peer.

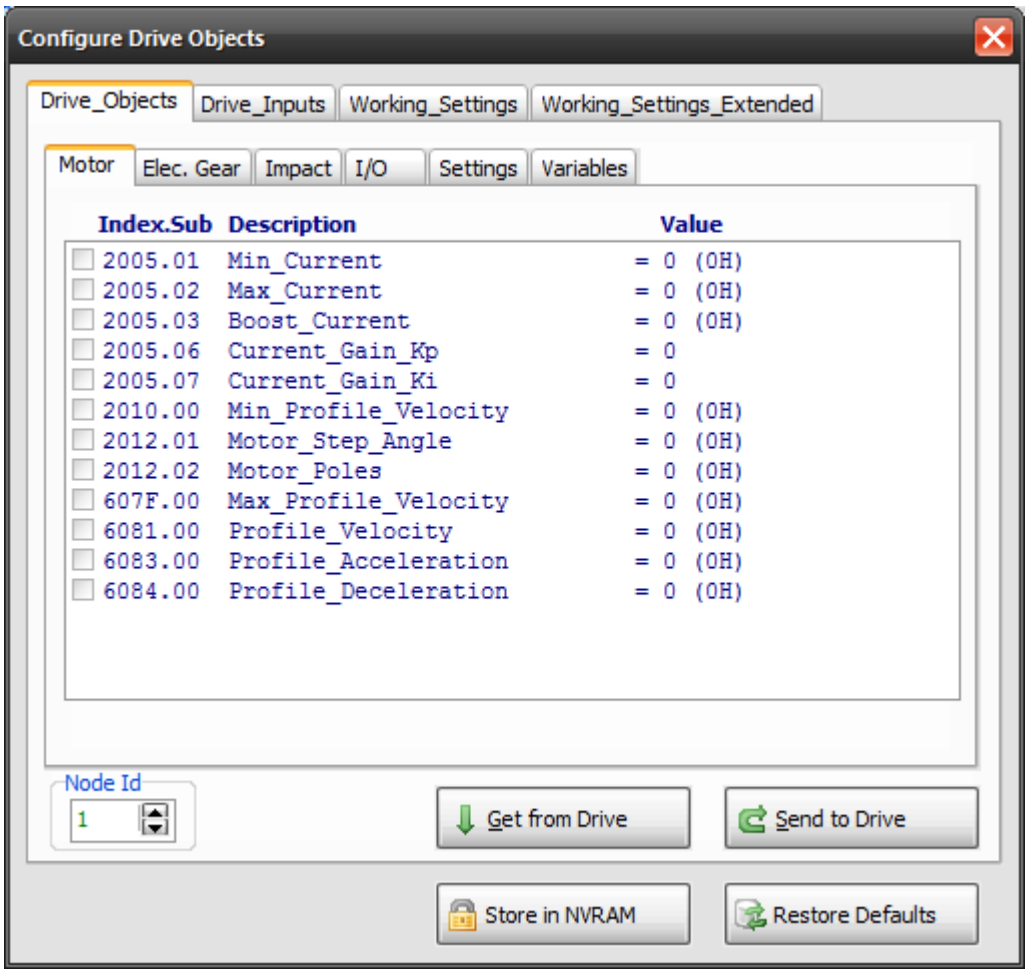
The **?** button open a window with the description of the bits of the *Drive_Status* object.
The **I/O** button open a window with the status of the drive input/outputs: to change the status of an output simply double click on its led.
The **Errors** button open a window with the last 8 errors/protections occurred to the drive.
The **Stop Motor** button stops the motor immediately.
The **Direct CMD** button opens the *'Direct Commands'* window.

The **Fw Update** button opens the *'Download Firmware'* window.
The **Save Cfg** button opens a save to file dialog window to save the current configuration of the *'Data Monitor'* frame and *'Drive Object Config'*.
The **Open Cfg** button opens a load from file dialog window to load a previous saved configuration of the *'Data Monitor'* frame and *'Drive Object Config'*.
The **Exit** button closes the Slim Line Drives CAN program.
Pressing the left mouse button inside the *'Data Monitor'* frame a pop up menu will appear giving the possibility to :

- add a drive's internal object to the list of inspected objects.
- delete an object from the list of inspected objects.
- change the value of the selected object (same as double clicking on one object).
- clear all list of inspected objects.

Pressing the left mouse button inside the *'Message Monitor'* frame a pop up menu will appear giving the possibility to clear the text frame.

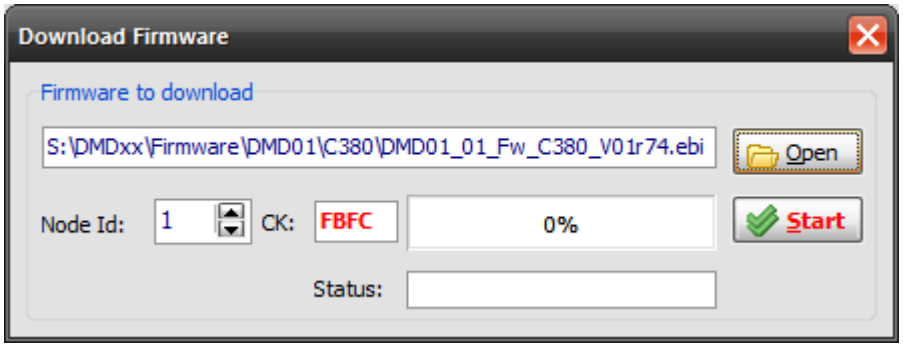
The *'Configure Drive Objects'* window (opened by selecting the Tools/Drive Object Config from the main window) enable to change all drive's parameters at the same time.



To change an object value, simply double click on it.
If the object left check box is checked, then when the **'Send to drive'** button is pressed it will be sent to the drive.
These information are stored in the .cfg configuration file (see main window Save/Open Cfg).
When the **'Get from drive'** button is pressed, whole objects value will be read from the drive.
Pressing **'Store in NVRAM'** button, will be sent to the drive the command to store in NVRAM the object values actually stored in RAM. The next time the drive will switch on it will retrieve objects starting values from NVRAM. See 'Store_Parameters' object for more details.

Pressing **'Restore Default'** button, will be sent to the drive the command to store in NVRAM the object factory default values. See 'Restore_Parameters' object for more details.

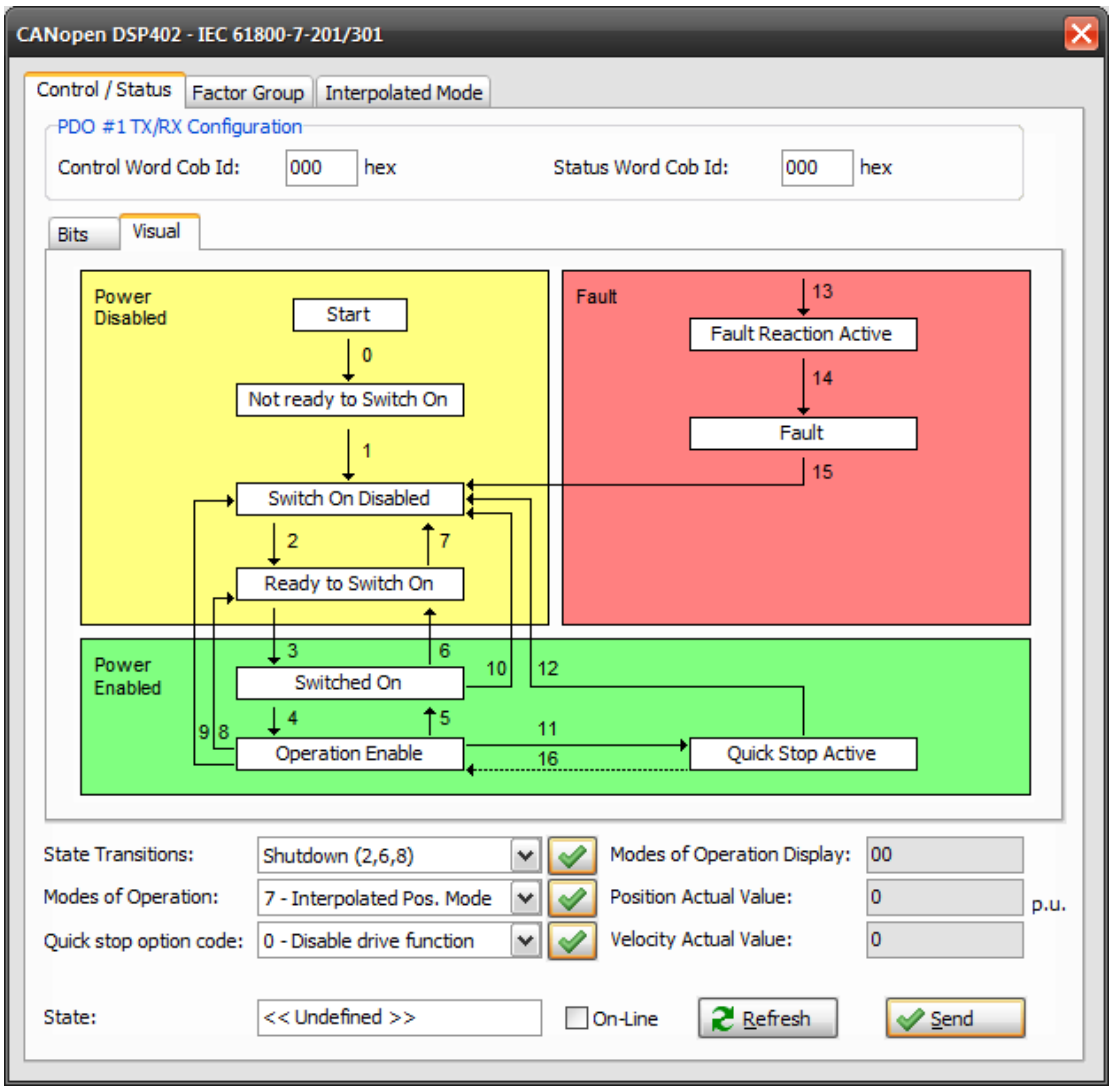
With the Slim Line Drives CAN Monitor it is possible to update the Slim Line drive internal firmware: Select *'Firmware Update'* from the Tools menu.
The window that will pop up is the following:



Open a .ebi firmware file pressing on *'Open'* button and then press on *'Start'* button.

Note: !! Do not switch to other tasks during firmware update procedure !!

It is possible to view a DSP402 specific windows. Select *'CANopen DSP402'* from Tools menu.



D Appendix : Multiplexed IO allocations

The following table shows the functions assigned to each digital input.

SW1 (Slim Line Series Drives)								
		Homing mode						
Input	Encoder Quadrature	Home switch	Index Pulse	Negative Limit Switch	Positive Limit Switch	Sensor selection code	Forward Limit switch	Backward Limit switch
B0_In0	Phase A Encoder #0	√	√	√	√	0	√	√
B0_In1	Phase B Encoder #0	√	√	√	√		√	√
B0_In2	Phase A Encoder #1					-2	√	√
B0_In3	Phase B Encoder #1						√	√
B0_In4 (1)		√	√	√	√		√	√
B0_In5 (1)		√	√	√	√		√	√
B0_In6 (1)							√	√
B0_In7 (1)							√	√
B1_In0 (1)								
B1_In1 (1)								
B1_In2 (1)								
B1_In3 (1)								
B1_In4 (1)								
B1_In5 (1)								
B1_In6 (1)								
B1_In7 (1)								

SM1A (Slim Line Series Drives) / DMD105								
		Homing mode						
Input	Encoder Quadrature	Home switch	Index Pulse	Negative Limit Switch	Positive Limit Switch	Sensor selection code	Forward Limit switch	Backward Limit switch
B0_In0		√	√	√	√		√	√
B0_In1		√	√	√	√		√	√
B0_In2	Phase A Encoder #1					-2	√	√
B0_In3	Phase B Encoder #1						√	√
B0_In4	Not Available							
B0_In5 (2)	Zero Encoder #0	√	√	√	√		√	√
B0_In6 (2)	Phase A Encoder #0					0	√	√
B0_In7 (2)	Phase B Encoder #0						√	√

SB4D								
Input	Encoder Quadrature	Homing mode				Sensor selection code	Forward Limit switch	Backward Limit switch
		Home switch	Index Pulse	Negative Limit Switch	Positive Limit Switch			
B0_In0		√	√	√	√		√	√
B0_In1		√	√	√	√		√	√
B0_In2	Phase A Encoder #1	√	√	√	√	-2	√	√
B0_In3	Phase B Encoder #1	√	√	√	√		√	√
B0_In4								
B0_In5	Zero Encoder #0	√	√	√	√		√	√
B0_In6	Phase A Encoder #0	√	√	√	√	0	√	√
B0_In7	Phase B Encoder #0	√	√	√	√		√	√

Follows the hardware functions assigned to each digital output.

	SW1 (Slim Line Series Drives)
Output	**Function**
B0_Out0	//
B0_Out1	//
B1_Out0 ([1])	//
B1_Out1 ([1])	//
B1_Out2 ([1])	//
B1_Out3 ([1])	//
B1_Out4 ([1])	//
B1_Out5 ([1])	//
B1_Out6 ([1])	//
B1_Out7 ([1])	//

([1]) Available only on SW1 with expansion.
([2]) Internal Inputs not available to user. Used on SM1A versions with internal incremental encoder.

E Appendix: Display Status

The 7 segments drive display can have the following status:

Display Status	Description
L	Drive in boot mode. A new firmware should be downloaded to drive.
U	Firmware update in progress. Do not power off the drive until the update process is completed!
I	Initialization phase. Should last few seconds. While in this condition the drive is not fully operational.
S	Fixed Character = Drive ok and operational Blinking Character = (Master Enable off)
S+1	Warning : Power supply near limit
S+3	Warning : Temperature near limit
S+7	Warning : EEprom near Write Overrun
S+8	Warning : EEprom near End of Life
F+0	Error : Watchdog
F+1	Error : Internal software error
F+2	Error : Missing Calibration
F+4	Error : EEprom fail
F+7	Error : EEprom Write Overrun
P+0	Protection: Motor is in open phase condition
P+1	Voltage protection
P+2	Current protection
P+3	Thermal protection
P+6	Motor Current Regulation is out of range (*)

(*) Verify Motor Currents correctness and Current_Gain_Kp and Current_Gain_Ki objects (for SW1_9x60 drive)

Objects Index (by name)

Objects Index (by index)