

HDT brushless servodrives
DGM series

Canopen manual



Version 6.03

Date rev.	27 march 2012
-----------	---------------

Main Index

1	Introducing CanOpen	8
1.1	Introduction.....	8
2	CanOpen protocol.....	9
2.1	DGM parameters for CanOpen protocol setting.....	9
3	DS-301 protocol.....	10
3.1	SDO.....	10
3.2	PDO.....	10
3.3	SYNC.....	10
3.4	EMERGENCY OBJECT.....	10
3.4.1	Emergency object structure.....	10
3.4.2	Undervoltage Automatic Reset with DC BUS	11
3.4.3	Overvoltage Automatic Reset with DC BUS.....	12
3.5	NETWORK MANAGEMENT OBJECTS	13
3.6	Node Guarding protocol.....	14
3.7	DS-301 Protocol Communication Parameters.....	15
3.8	Communication Parameters.....	16
3.8.1	Description of the communication parameters.....	17
3.8.2	Object 1010h – Store Parameters.....	18
	With vector store parameters it's possible save the drive parameters. According to the type of parameters you want to save you can write on the subindex 1,2,3 or 4 the 32bit data 65766173h (save command). This value corresponds to string “save” in ASCII format.....	18
3.8.3	Error Codes.....	19
3.9	SDO parameters.....	20
3.9.1	Description of the SDO parameters.....	20
3.10	RPDO parameters.....	21
3.10.1	Description of the RPDO parameters.....	22
3.10.2	Description of the RPDO mapping parameters.....	22
3.11	PDO parameters (transmission).....	23
3.11.1	Description of the TPDO parameters.....	24
3.11.2	Description of the TPDO mapping parameters.....	24
4	DS-402 Protocol Parameters.....	25
4.1	DS-402 Standard.....	25
4.2	State Machine.....	26
4.3	PDO default mapping settings	27
4.3.1	RPDO 2 default mapping (index 1600h).....	28
4.3.2	RPDO 2 default mapping (index 1601h).....	28
4.3.3	RPDO 3 default mapping (index 1602h).....	28
4.3.4	RPDO 4 default mapping (index 1603h).....	29
4.3.5	TPDO 1 default mapping (index 1A00h).....	29
4.3.6	TPDO 2 default mapping (index 1A01h).....	29
4.3.7	TPDO 3 default mapping (index 1A02h).....	30
4.3.8	TPDO 4 default mapping (index 1A03h).....	30
4.4	DS-402 Control Variables.....	31
4.4.1	Object 6040h - Controlword.....	32
4.4.1.1	Controlword Bits 0..3 and 7.....	32
4.4.1.2	Controlword Bits 4,5,6 and 8.....	32
4.4.2	Object 6041h - Statusword.....	33
4.4.2.1	Statusword Bits regarding the drive state.....	33
4.4.2.2	Statusword Bits 12 and 13 - operation mode.....	34

4.4.3 Object 6007h - Abort connection option code.....	35
4.4.4 Object 603Fh - Error code.....	35
4.4.5 Object 604Eh – vl velocity reference.....	36
4.4.6 Object 605Ah - Quick stop option code.....	36
4.4.7 Object 605Dh - Halt option code.....	36
4.4.8 Object 6050h - vl slow down time.....	37
4.4.9 Object 6051h - vl quick stop time.....	37
4.4.10 Object 6060h - Modes of operation.....	37
4.4.11 Object 6061h - Modes of operation display.....	38
4.4.12 Object 60FDh - Digital inputs.....	39
4.5 Conversion Factors.....	41
4.5.1 Object 604Ch – vl dimension factor.....	42
(*) Excluded zero value.....	42
4.5.2 Object 6089h - Position notation index.....	42
4.5.3 Object 608Ah - Position dimension index.....	42
4.5.4 Object 608Bh - Velocity notation index.....	43
4.5.5 Object 608Ch - Velocity dimension index.....	43
4.5.6 Object 608Dh - Acceleration notation index.....	43
4.5.7 Object 608Eh - Acceleration dimension index.....	43
4.5.8 Object 6093h - Position factor.....	44
4.5.9 Object 6094h - Velocity encoder factor.....	44
4.5.10 Object 6097h - Acceleration factor.....	45
4.5.11 Object 607Eh - Polarity.....	45
4.6 Position Control Parameters.....	46
4.6.1 Object 6062h - Position demand value.....	47
4.6.2 Object 6063h - Position actual value*.....	47
4.6.3 Object 6064h - Position actual value.....	47
4.6.4 Object 6065h - Following error window.....	48
4.6.5 Object 6066h - Following error time out.....	48
4.6.6 Object 607Ch - Home offset.....	48
4.6.7 Oggetto 607Dh - Software position limit.....	49
4.6.8 Object 60F4h – Following error actual value.....	49
4.6.9 Object 60FCh - Position demand value*.....	50
4.7 Positioner (Profile Position Mode).....	51
4.7.1 Controlword in profile position mode (bit setting).....	52
4.7.2 Statusword in profile position mode bit value.....	53
4.7.3 Manage the Positioner through bit 5 (Change set immediately) Controlword	54
4.7.4 Object 607Ah - Target position.....	55
4.7.5 Object 6081h - Profile velocity.....	56
4.7.6 Object 6083h - Profile acceleration.....	56
4.7.7 Object 6084h - Profile deceleration.....	56
4.7.8 Object 6086h - Motion profile type.....	57
4.8 Homing (Homing Mode).....	59
4.8.1 Controlword in Homing Mode (bit setting).....	60
4.8.2 Statusword in Homing Mode (bit value).....	61
4.8.3 Object 6098h - Homing method.....	62
4.8.3.1 Method 0 - No homing operation required.....	62
4.8.3.2 Method 1 - Homing on the negative limit switch and index pulse.....	62
4.8.3.3 Method 2 - Homing on the positive limit switch and index pulse.....	62
4.8.3.4 Method 3 - Homing on the positive home mark and index pulse.....	63
4.8.3.5 Method 4 - Homing on the positive home mark and index pulse.....	63

4.8.3.6 Method 5 - Homing on the negative home mark and index pulse.....	64
4.8.3.7 Method 6 - Homing on the negative home mark and index pulse.....	64
4.8.3.8 Method 7 - Homing on the home mark and index pulse.....	65
4.8.3.9 Method 8 - Homing on the home mark and index pulse.....	65
4.8.3.10 Method 9 - Homing on the home mark and index pulse.....	65
4.8.3.11 Method 10 - Homing on the home mark and index pulse.....	65
4.8.3.12 Method 11 - Homing on the home mark and index pulse.....	66
4.8.3.13 Method 12 - Homing on the home mark and index pulse.....	66
4.8.3.14 Method 13 - Homing on the home mark and index pulse.....	66
4.8.3.15 Method 14 - Homing on the home mark and index pulse.....	66
4.8.3.16 Methods from 17 to 30.....	67
4.8.3.17 Method 33 - Homing on index pulse (negative direction).....	68
4.8.3.18 Method 34 - Homing on index pulse (positive direction).....	68
4.8.3.19 Method 35 - Homing on the current position.....	68
4.8.4 Object 609Ah – Homing acceleration.....	69
4.9 Specific limit switch settings	69
4.10 Interpolated Position Mode.....	71
4.10.1 Controlword in Interpolated Position Mode bit setting.....	72
4.10.2 Statusword in Interpolated Position Mode bit value.....	73
4.10.3 Object 60C0h - Interpolation sub mode select.....	74
4.10.4 Object 60C1h - Interpolation data record.....	75
4.10.5 Object 60C2h - Interpolation time period.....	76
4.11 Profile Velocity Mode.....	77
4.11.1 Controlword in Profile Velocity Mode (bits setting).....	78
4.11.2 Statusword in Profile Velocity Mode (bits value).....	79
4.11.3 Object 6069h - Velocity sensor actual value.....	80
4.11.4 Object 606Bh - Velocity demand value.....	80
4.11.5 Object 606Ch - Velocity actual value.....	80
4.11.6 Object 606Dh - Velocity window.....	80
4.11.7 Object 606Eh - Velocity window time.....	81
4.11.8 Object 606Fh - Velocity threshold.....	81
4.11.9 Object 6070h - Velocity threshold time.....	81
4.11.10 Object 60FFh – Target Velocity.....	82
4.12 Velocity Mode.....	83
4.12.1 Controlword in Velocity Mode (bits setting).....	84
4.12.2 Statusword in profile position mode (bits value).....	85
4.12.3 Object 6042h - vl target velocity.....	86
4.12.4 Object 6043h - vl velocity demand.....	86
4.12.5 Object 6044h - vl control effort.....	86
4.12.6 Object 6046h - vl velocity min max amount.....	86
4.12.7 Object 6048h - vl velocity acceleration.....	87
4.12.8 Object 6049h - vl velocity acceleration.....	88
4.13 Profile Torque Mode.....	88
4.13.1 Setting bit Controlword in Profile Torque Mode.....	90
4.13.2 Valore bit Statusword in Profile Torque Mode.....	90
4.13.3 Object 6071 - Target torque.....	91
4.13.4 Object 6077 - Torque actual value.....	91
4.13.5 Object 6078 - Torque slope.....	91
4.13.6 Object 6088 - Torque profile type.....	91
4.13.7 Torque current limitations.....	93
4.14 Electric Axis.....	96

4.14.1 Set bit Controlword in Electric Axis.....	98
4.14.2 Value bit Statusword in Electric Axis.....	99
4.14.3 Object 2200 – Numerator Ratio.....	100
4.14.4 Object 2201 - Denominator Ratio.....	100
4.14.5 Object 2202 – Lock mode.....	100
4.14.6 Object 2203 – Mode Unlock.....	101
4.14.7 Object 2204 – Encoder Mode.....	102
4.14.8 Object 2205 – Unlock position absolute/relative.....	104
4.14.9 Object 2206 – Lock in Speed-acceleration.....	104
4.14.10 Object 2207 – Lock in speed-Jerk	104
4.14.11 Object 2208 – Lock in position-quota.....	104
4.14.12 Object 2209 – Unlock in speed-deceleration.....	105
4.14.13 Object 220A – Unlock in speed-Jerk	105
4.14.14 Object 220B - Unlock in position-speed.....	105
4.14.15 Object 220C - Unlock in position-deceleration.....	105
4.14.16 Object 220D - Unlock in position-Quota.....	106
4.14.17 Object 220E - Encoder pulses	106
4.14.18 Object 220F – Pulse Encoder master.....	106
5 Specific parameters of the Drive.....	107
5.1.1 Object 2000h – Motor Rated Speed	109
5.1.2 Object 2001h – Motor Rated Current	109
5.1.3 Object 2002h – Motor Peak Current.....	109
5.1.4 Object 2003h – Pole pairs of the motor.....	109
5.1.5 Object 2004h – Offset resolver.....	110
5.1.6 Object 2005h – Speed Limit.....	110
5.1.7 Object 2006h – Current Limit.....	110
5.1.8 Object 2007h – Kp position regulator.....	110
5.1.9 Object 2008h – Kp speed regulator.....	111
5.1.10 Object 2009h – Ki speed regulator.....	111
5.1.11 Object 200Ah – Kp current regulator.....	111
5.1.12 Object 200Bh – Ki current regulator.....	111
5.1.13 Object 200Ch – Eanble Notch Filter.....	111
5.1.14 Object 200Dh – Notch filter frequency.....	112
5.1.15 Object 200Eh – Band with of notch filter.....	112
5.1.16 Object 200Fh – I2t management Mode.....	112
5.1.17 Object 2010h – “Enable” Input management.....	112
5.1.18 Object 2011h – Jerk position.....	113
5.1.19 Object 2012h – Measured current.....	113
5.1.20 Object 2013h – Motor brake time coupling.....	114
5.1.21 Object 2014h – Motor Brake Release time.....	114
5.1.22 Object 2015h – Time of Brake motor deceleration.....	114
5.1.23 Object 2016h – Brake motor speed intervention.....	114
5.1.24 Object 2017h – Enable Brake motor.....	114
5.1.25 Object 201Ah – Enable Position Limit.....	115
5.1.26 Object 2020h – ADC External.....	115
5.2 Function of relative shift related to input 7.....	115
5.2.1 Object 2018h – Enable input 7.....	115
5.2.2 Object 2019h – Relative shift offset input 7	115
5.2.3 Bit 14 status word fordetection of positioning input 7.....	116
5.2.4 Relative shiftment function management from input 7.....	116

1.Tables Index

Table 1: BaudRate.....	9
Table 2: Emergency message structure.....	10
Table 3: DGM alarm bits on Manufacturer specific Error Field.....	11
Table 4: Communication States and Objects.....	13
Table 5: NMT commands.....	14
Table 6: Variable Type Abbreviations.....	15
Table 7: Communication Parameters.....	16
Table 8: Description of the communication parameters.....	17
Table 9: Error Codes.....	19
Table 10: PDO parameters.....	20
Table 11: PDO Parameters (Reception).....	21
Table 12: RPDO configuration parameters (reception).....	22
Table 13: Transmission types in the reception PDOs.....	22
Table 14: PDO mapping parameters (reception).....	22
Table 15: TPDO parameters	23
Table 16: TPDO configuration parameters (transmission).....	24
Table 17: Types of transmission in the transmission PDOs.....	24
Table 18: mapping parameters (transmission).....	24
Table 19: State description (State Machine).....	26
Table 20: Control parameters.....	31
Table 21: controlword – operation mode bit.....	32
Table 22: status word - bit description.....	33
Table 23: status word, bits relative to the state.....	33
Table 24: status word, bits relative to the operation mode	34
Table 25: DGM Drive input.....	39
Table 26: Conversion Factors.....	41
Table 27: Bit Polarity.....	45
Table 28: Position control parameters.....	46
Table 29: Positioner parameters.....	51
Table 30: Homing parameters.....	59
Table 31: Interpolator parameters.....	71
Table 32: Profile velocity mode parameters.....	77
Table 33: Velocity mode parameters.....	83
Table 34: Parameters of Profile Torque Mode.....	88
Tabella 35: Electric Axis Parameters.....	96
Tabella 36: Specific parameters of the Drive.....	107
Tabella 37: I2t management mode.....	112
Tabella 38: “Enable” input management.....	113

Drawings Index

Drawing 1: NMT telegram.....	13
Drawing 2: Node Garding telegram.....	14
Drawing 3: Communication architecture	25
Drawing 4: State Machine DS-402.....	27
Drawing 5: Mode Single Setpoint.....	54

Drawing 6: Mode Change Set Immediately.....	55
Drawing 7: Method 1 - Homing on CCW limit switch and zero resolver impulse.....	62
Drawing 8: Method 2 - Homing on CW limit switch and zero resolver impulse.....	63
Drawing 9: Method 3 - Homing in home sensor.....	63
Drawing 10: Method 4 - Homing on home sensor.....	64
Drawing 11: Method 5 - Homing on home sensor.....	64
Drawing 12: Method 6 - Homing on home sensor.....	65
Drawing 13: Method 7,8,9,10 - Homing on home sensor.....	66
Drawing 14: Method 11,12, 13, 14 - Homing on home sensor.....	67
Drawing 15: Method 17 - Homing on CCW limit switch.....	67
Drawing 16: Method 18 - Homing on CW limit switch.....	68
Drawing 17: Method 3,4 - Homing on resolver zero impulse.....	68
Drawing 18: Linear interpolation.....	74
Drawing 19: Cubic interpolation	75
Drawing 20: Bit 4,5,6 controlword velocity mode.....	85
Drawing 21: vl velocity min max amount characteristic.....	87
Drawing 22: Electric Axis Phases.....	97
Drawing 23: Encoder mode “Channel A-B”.....	103
Drawing 25: Encoder mode “Channel CW-CCW”.....	103
Drawing 24: Encoder mode “Frequency and Direction”.....	103

1 Introducing CanOpen

1.1 Introduction

This document gives an indicative description of the structure and parameters of CanOpen on the HDT DGM drive. Further information on the communication parameters can be found in the CiA DS-301 document, while DS-402 should be consulted for more information on the specific control parameters of the drive.

2 CanOpen protocol

2.1 DGM parameters for CanOpen protocol setting

The CanOpen protocol is based on a serial bus transmission system, in which a unique address is assigned to each node, therefore the node address and a communication speed that is equal for all the connected devices must be set in each Can network.

The following parameters can be used in the **DGM** drive to set the connecting data:

- The **S.7000** parameter enables Can bus communication, by setting the "2-CAn" option.
- The **S.7200** parameter sets the drive address within a 1 to 127 range.
- The **S.7201** parameter selects the communication speed according to table 1.
- The **S.7202** parameter sets the internal timeout of communication of the drive in ms.
- The **S.7203** parameter enable the communication timeout related to parameter **S.7202** . If the master doesn't manage the Node-Guarding, it's possible to use this parameter to activate the internal control procedure of communication timeout. The error of communication is managed according to set made on parameter "*Abort connection option code*" (index 6007h).

BaudRate
1 Mbit/s
800 Kbit/s
500 Kbit/s
250 Kbit/s
125 Kbit/s
100 Kbit/s
50 Kbit/s
20 Kbit/s
10 Kbit/s

Table 1: BaudRate

3 DS-301 protocol

The main structure of the CanOpen is based on the use of the variables contained in the *Object Dictionary* of the device. Access to this set of parameters is managed by different types of telegram.

3.1 SDO

The SDO (Service Data Object) packet usually implies an exchange of data between master and slave (or client/server), where the master sends a request to the slave and the slave sends a confirmation reply which contains the data requested by the master.

3.2 PDO

PDO (Process Data Object) is a broadcast communication system where no confirming reply is requested by the node that receives the data. According to the DS-301 standard this implies PDO formatting, which is normally carried out by the network master. In this manner a PDO telegram is composed beforehand with indication of the set of variables belonging to the *Object Dictionary* that makes up the telegram.

There are two types of PDO on the basis of the transmission direction:

- **TPDO (Transmit-PDO)**: data telegram sent by the slave for the master
- **RPDO (Receive- PDO)**: data telegram sent by the master to a slave

3.3 SYNC

SYNC is a telegram generally sent by the master, with the purpose of supplying a regular clock to synchronise the events (such as sending PDOs) within the network.

3.4 EMERGENCY OBJECT

EMERGENCY OBJECT is a telegram that can be sent by any device to signal an emergency condition. The displayed alarm code can be among those indicated by the standard, or can be a specific device alarm code.

3.4.1 Emergency object structure

An emergency type message is made up of 8 bytes as described below:

Byte	Description
0..1	Emergency Error Code , see error code table
2	register error (object at the 1001h index)
3..7	Manufacturer specific Error Field : four of these five bytes are used to take all the alarms present towards the DGM drive; every single bit of bytes 3, 4, 5 and 6 identifies an alarm. If the alarm is present the corresponding bit is taken to one.

Table 2: Emergency message structure

Bit	DGM alarm bit description on <i>Manufacturer specific Error Field</i>
0	Parameter saving error
1	Hall effect sensors settings altered
2	Power Overcurrent
3	DC BUS overvoltage
4	No network voltage
5	No Phase
6	Resolver error
7	Secure Disable (STO)
8	Motor overheating
9	Braking Resistance
10	No 24V
11	DC BUS undervoltage
12	Position Error
13	Home Position Error
14	Inverter I2t warning
15	Motor I2t warning
16	Overspeeding warning
17	Secure Disable Error 1
18	Out of quota warning
19	Life Guard Error
20	Memory Fault
21	Error on phase motor
22	Secure Disable Error 2
23..31	

Table 3: DGM alarm bits on Manufacturer specific Error Field

3.4.2 Undervoltage Automatic Reset with DC BUS

If an undervoltage event occurs, the DGM drive can behave in two different ways depending on the setting of the “**S.4001**” parameter:

- **S.4001** parameter set to “off”: in this case the drive remains in a fault state, even when the voltage returns to its normal state. A reset must be sent in order to move to a “*Switch On Disable*” state.
- **S.4001** parameter set to “on”: when the voltage returns to its normal state the undervoltage alarm is cancelled and the drive automatically moves to the “*Switch On Disable*” state.

The undervoltage alarm appears on the DGM display with the code “FA 12”.

3.4.3 Overvoltage Automatic Reset with DC BUS

If an overvoltage event occurs, the DGM drive can behave in two different ways depending on the setting of the “**S.4000**” parameter:

- **S.4000** parameter set to “off”: the drive remains in a fault state, even when the voltage returns to below the maximum permitted threshold. A reset must be sent in order to move to a “*Switch On Disable*” state.
- **S.4000** parameter set to “on”: when the voltage returns to below the maximum threshold, the overvoltage alarm is cancelled and the drive automatically moves to the “*Switch On Disable*” state.

The undervoltage alarm appears on the DGM display with the code “FA 04”.

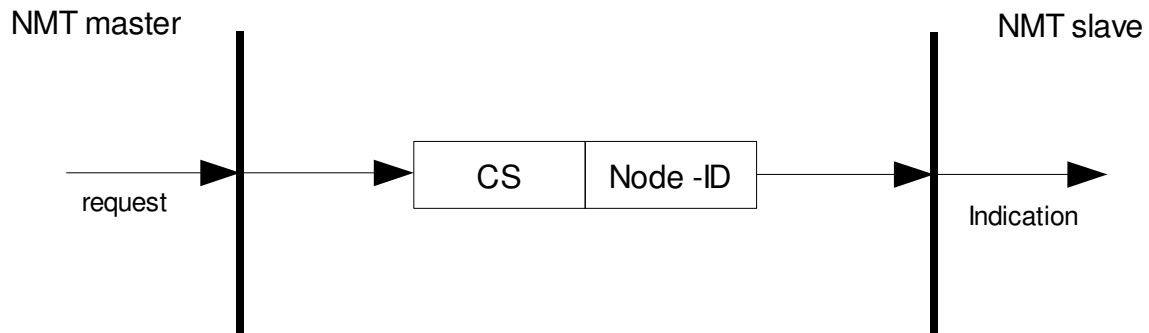
3.5 NETWORK MANAGEMENT OBJECTS

NETWORK MANAGEMENT OBJECTS (NMT) are telegrams used for handling *Network Management* type services in Master-Slave structures. The NMT structure manages communication through four states (*Initialising*, *Pre-operational*, *Operational*, *Stop*), using them to enable or disable the transmission of specific types of telegrams and to initialise the device.

Using NMT type telegrams, the master sets the slave state by sending a package made up of two pieces of data. The first is created by *the command specifier* (CS), and the second parameter contains the Node-ID.

	INITIALISING	PRE-OPERATIONAL	OPERATIONAL	STOPPED
PDO			X	
SDO		X	X	
Synchronisation Object		X	X	
Time Stamp Object		X	X	
Emergency Object			X	
Boot-Up Object	X			
Network Management Objects		X	X	X

Table 4: Communication States and Objects



Drawing 1: NMT telegram

Value	Command Specifier (CS)	Description
001	Start Remote Node	Switches to the Operational state
002	Stop Remote Node	Switches to the Stopped state
128	Enter Pre-Operational-State	Moves to the Pre-Operational state

129	Reset Node	Resets Node
130	Reset Communication	Resets communication

Table 5: NMT commands

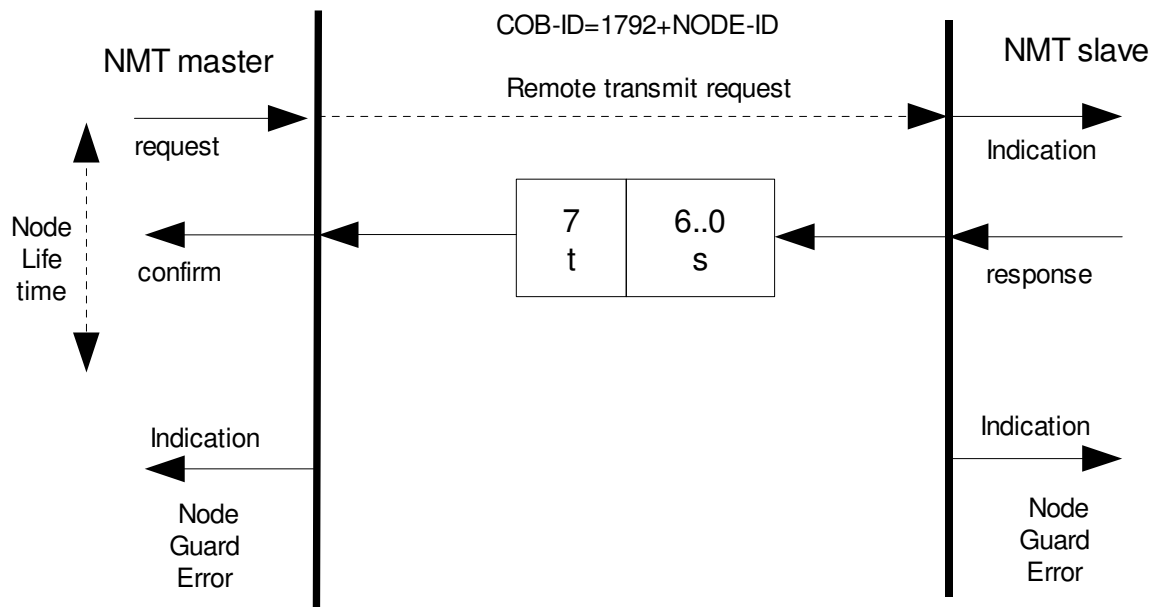
3.6 Node Guarding protocol

Node guarding is a protocol used to verify communication between the master and the network slaves. The master must send the request formed by the COB-ID= (1792 + Node-ID) to the slave within the time indicated by the *Node life time* parameter, otherwise the slave must identify the lack of communication and act as a result. Once the request has been received by the master, the slave must also reply with a confirmation package within the *Node life time*, otherwise the master identifies the lack of communication from the slave and sends notification of this error to the master application.

The *node guarding* protocol becomes active if the *Guard time* (index 100Ch) and the *Life time factor* (index 100Dh) parameters are not zero and when the master sends the first *node guarding* request.

The *Node life time* parameter is expressed in ms and is given by the product of the *Guard time* parameter and the *Life time factor*.

The DGM is set to manage *node guarding* request telegrams coming from the master; if there is no communication the action set in the *Abort connection option code* (index 6007h) parameter is undertaken.



Drawing 2: *Node Garding* telegram

The reply datum given by the slave is made up of one byte, where bit 7 is managed in "toggle". The remaining bits carry the slave NMT state.

3.7 DS-301 Protocol Communication Parameters

CanOpen communication in the DGM drive is managed by parameters included in the 1000h and 1FFFh interval of the *Object Dictionary*. This range includes the settings of the communication profile that is common to each device connected to the network (CanOpen standard DS-301).

Code	Variable type
UINT8	UNSIGNED8
INT8	INTERGER8
UINT16	UNSIGNED16
INT16	INTERGER16
UINT32	UNSIGNED32
INT32	INTERGER32
STR	STRING

Table 6: Variable Type Abbreviations

3.8 Communication Parameters

COMMUNICATION PARAMETERS							
Index (HEX)	Sub-Index	Parameter name Object Dictionary	Type	Value Range	Default	Map	Attributes
1000h		Device Type	UINT32	0..FFFFFFFFh	00020192h	No	ro
1001h		Error Register	UINT8	0..FFh	0	No	ro
1003h		Pre-defined Error Field	ARRAY				
	0	Number of error	UINT8	0..8	0	No	rw
	1..8	Standard error field	UINT32	0..FFFFFFFFh	0	No	ro
1008h		Manufacturer device name	STR	--		No	const
100Ah		Manufacturer software version	STR	--			
100Ch		Guard time	UINT16	0..FFFFh	100 [ms]	No	rw
1010h		Store Parameters					
	0	Max subindex number	UINT8	4	4	No	ro
	1..4	Save Mode	UINT32	--	--	No	rw
1014h		COB-ID Emergency Object	UINT32	81h..FFh	80h+Node-ID	No	ro
1018h		Identity Object	Record				
	0	Number of entries	UINT8	0..1	1	No	ro
	1	Vendor ID	UINT32	0..FFFFFFFFh	00000289h	No	ro
	2	Product Code	UINT32	0..FFFFFFFFh	000C0442h	No	ro
	3	Revision number	UINT32	0..FFFFFFFFh	--	No	ro
100Dh		Life time factor	UINT8	0..FFh	2	No	rw
1014h		COB-ID Emergency Object	UINT32	81h..FFh	80h+Node-ID	No	ro
1018h		Identity Object	RECORD				
	0	number of entries	UINT8	0..1	1	No	ro
	1	Vendor ID	UINT32	0..FFFFFFFFh	--	No	ro

Table 7: Communication Parameters

3.8.1 Description of the communication parameters

Index (HEX)	Parameter name	Description		
1000h	Device Type	<ul style="list-style-type: none"> • First field (16 high bits): Indicates the type of device and for the DGM drive is worth 0002h (servo drive). • Second field (16 low bits) Indicates the profile of the DS-402 device and in the DGM is equal to 0192h 		
1001h	Error Register	Register of the errors (according to CanOpen DS-301 standard). In the DGM drive only bit 0 is managed, which indicates the presence of a generic error. For further information on the alarms refer to these parameters: <ul style="list-style-type: none"> • Pre-defined Error Field (index 1003h) • Error Code (index 603Fh) • Emergency Object 		
1003h	Pre-defined Error Field	This parameter is made up of 9 fields: <ul style="list-style-type: none"> • sub index 0: one byte variable, indicates the number of alarms saved. The maximum readable value is 8, which corresponds to the maximum number of alarms that can be saved. To delete the saved alarms, zero the zero sub index, as specified in the DS-301. • sub index from 1 to 8: 32 bit variables, preset for saving possible alarms with 32 bit numeric codes. For a description of the meaning of the alarm codes refer to the "Error Codes" table. 		
1008h	Manufacturer device name	Device identification string, the DGM drive uses the "HDT Lovato-DGM" string.		
100Ah	Manufacturer software version	String that indicates the software version of the DGM drive		
100Ch	Guard time	Together with the <i>life time factor</i> , this parameter is necessary for calculating the <i>node life time</i> .		
100Dh	Life time factor	Together with the <i>guard time</i> , this parameter is necessary for calculating the <i>node life time</i> .		
1014h	OB-ID Emergency Object	The COB-ID for the emergency object is saved in this variable, COB-ID=80h+Node-ID		
1018h	Identity Object	Record of parameters that contain general information on the device.		
		Sub Index	Name	Description
		0	Number of entries	It's the number of elements that compose the structure
		1	Vendor ID	The value in this field is 00000289h
		2	Product Code	Product code is 000C0442h
		3	Revision Number	Hardware revision number

Table 8: Description of the communication parameters

3.8.2 Object 1010h – Store Parameters

Index (HEX)	Sub-Index	Name	Type	Range	Default	Unit	Attributes
1010h		Store Parameters	ARRAY				
	0	Number of entries	UINT8	4	4	No	ro
	1	Save all parameters	UINT32	--	--	No	rw
	2	Save parameters DS301	UINT32	--	--	No	rw
	3	Save parameters DS402	UINT32	--	--	No	rw
	4	Save parameters Drive	UINT32	--	--	No	rw

With vector *store parameters* it's possible save the drive parameters. According to the type of parameters you want to save you can write on the subindex 1,2,3 or 4 the 32bit data 65766173h (save command). This value corresponds to string “save” in ASCII format.

E	V	A	S
65h	76h	61h	73h

The reading of one of the subindex 1,2,3 or 4 returns the value 00000001h, that indicate the availability to save parameters on command.

3.8.3 Error Codes

The error codes are saved in a 16 bit variable, and are reported in the first two bytes of the *emergency* telegrams.

The codes in bold indicate specific DGM drive errors (alarms).

ERROR CODES (Alarms)		
Error Code	DS-301 description	Detailed Description
5510h	<i>Working memory</i>	Parameter saving error
FF01h	<i>Manufacturer Alarm</i>	Hall effect sensor settings altered
2340h	Short Circuit Alarm	Power Overcurrent
3210h	Over Voltage DC link	DC BUS overvoltage
3100h	<i>Mains Voltage</i>	No network voltage
3130h	<i>Phase Failure</i>	No Phase
7303h	Resolver Fault	Resolver error
7120h	<i>Motor Alarm</i>	Motor overheating
7110h	Brake Chopper	Braking Resistance
5112h	Supply low Voltage	No 24V
3220h	DC link Under Voltage	DC BUS undervoltage
8611h	Following Error	Position Error
FF02h	<i>Manufacturer Alarm</i>	Home Position Error
FF03h	<i>Manufacturer Alarm</i>	Inverter I2t warning
FF04h	<i>Manufacturer Alarm</i>	Motor I2t warning
FF05h	<i>Manufacturer Alarm</i>	Overspeeding warning
FF06h	<i>Manufacturer Alarm</i>	Out of quota warning
8130h	Life Guard Error	Life Guard Error
FF07h	<i>Manufacturer Alarm</i>	Memory Fault
FF08h	<i>Manufacturer Alarm</i>	Secure Disable (STO)
FF09h	<i>Manufacturer Alarm</i>	Secure Disable Error 1
FF0Ah	<i>Manufacturer Alarm</i>	Motor Phase Error
FF0Bh	<i>Manufacturer Alarm</i>	Secure Disable Error 2

Table 9: Error Codes

3.9 SDO parameters

SDO PARAMETERS (Server)							
Index (HEX)	Sub-Index	Parameter name Object Dictionary	Type	Value Range	Default	Map	Attributes
1200h		Server SDO Parameter	RECORD				
	0	number of entries	UINT8	0..2	2	No	ro
	1	COB-ID Client → Server (rx)	UINT32	601h..67Fh	600h+Node-ID	No	ro
	2	COB-ID Server → Client (tx)	UINT32	581h..5FFh	580h+Node-ID	No	ro

Table 10: PDO parameters

3.9.1 Description of the SDO parameters

The 1200h index contains a record giving the reception server COB-ID of the SDO telegrams and the transmission server COB-ID of the SDO telegrams.

3.10 RPDO parameters

PDO PARAMETERS (Reception)							
Index (HEX)	Sub-Index	Parameter name Object Dictionary	Type	Value Range	Default	Map	Attributes
1400h		Receive PDO parameter 1	RECORD				
	0	largest sub-index supported	UINT8	0..2	2	No	ro
	1	COB-ID used by PDO	UINT32	--	200h + Node-ID	No	ro
	2	Transmission type	UINT8	0..FFh	0	No	rw
1401h		Receive PDO parameter 2	RECORD				
	0	largest sub-index supported	UINT8	0..2	2	No	ro
	1	COB-ID used by PDO	UINT32	--	300h + Node-ID	No	ro
	2	Transmission type	UINT8	0..FFh	0	No	rw
1402h		Receive PDO parameter3	RECORD				
	0	largest sub-index supported	UINT8	0..2	2	No	ro
	1	COB-ID used by PDO	UINT32	--	400h + Node-ID	No	ro
	2	Transmission type	UINT8	0..FFh	0	No	rw
1403h		Receive PDO parameter4	RECORD				
	0	largest sub-index supported	UINT8	0..2	2	No	ro
	1	COB-ID used by PDO	UINT32	--	500h + Node-ID	No	ro
	2	Transmission type	UINT8	0..FFh	0	No	rw
1600h		Receive PDO mapping 1					
	0	number of mapped PDO	UINT8	0..8	--	No	rw
	1..8	PDO mapping	UINT32	--	--	No	rw
1601h		Receive PDO mapping 2					
	0	number of mapped PDO	UINT8	0..8	--	No	rw
	1..8	PDO mapping	UINT32	--	--	No	rw
1602h		Receive PDO mapping 3					
	0	number of mapped PDO	UINT8	0..8	--	No	rw
	1..8	PDO mapping	UINT32	--	--	No	rw
1603h		Receive PDO mapping 4					
	0	number of mapped PDO	UINT8	0..8	--	No	rw
	1..8	PDO mapping	UINT32	--	--	No	rw

Table 11: PDO Parameters (Reception)

Note:

1.The mapping defaults of the *Receive* PDO are given in the chapter on the DS-402 protocol

3.10.1 Description of the RPDO parameters

Index (HEX)	Parameter name	Description
1400h..1403h	Receive PDO parameter	These records contain the receiving COB-IDs of the RPDO telegrams and the parameters for configuring the type of reception.

Table 12: RPDO configuration parameters (reception)

In these parameters sub-index 1 indicates the COB-ID of the reception PDO, while sub-index 2 gives the transmission type settable for the RPDOs. On the basis of the value set in sub-index 2, the following types of communication can be obtained:

Transmission type	Description
0	Asynchronous cyclic communication: The data is sent by the master only when it changes, when the sync is received the values are updated in the drive parameters
1..240	Synchronous cyclic communication: The numeric value set in <i>Transmission type</i> indicates the number of syncs after which the master sends the data. The received data is updated in the drive parameters when the sync is received.
241..251	Reserved values
252-253	These options are only valid for the TPDO because managed with RTR (<i>Remote Transmission Request</i>) type telegrams
254	Asynchronous communication (<i>manufacturer specific</i>): option reserved for specific drive communications; in this case the DGM drive behaves in the same way as for option 255.
255	Asynchronous communication (<i>device profile specific</i>): The data received through the RPDOs is immediately updated in the drive parameters without waiting for the sync .

Table 13: Transmission types in the reception PDOs

3.10.2 Description of the RPDO mapping parameters

Index (HEX)	Parameter name	Description
1600h..1603h	Receive PDO mapping	Record of parameters for setting the PDO content in reception; dynamic mapping is permitted in the DGM drive.

Table 14: PDO mapping parameters (reception)

3.11 PDO parameters (transmission)

PDO parameters (transmission)							
Index (HEX)	Sub-Index	Parameter name Object Dictionary	Type	Value Range	Default	Map	Attributes
1800h		Transmit PDO parameter 1	RECORD				
	0	largest sub-index supported	UINT8	0..2	2	No	ro
	1	COB-ID used by PDO	UINT32	--	180h+ Node-ID	No	ro
	2	Transmission type	UINT8	0..FFh	0	No	rw
1801h		Transmit PDO parameter 2	RECORD				
	0	largest sub-index supported	UINT8	0..2	2	No	ro
	1	COB-ID used by PDO	UINT32	--	280h + Node-ID	No	ro
	2	Transmission type	UINT8	0..FFh	0	No	rw
1802h		Transmit PDO parameter3	RECORD				
	0	largest sub-index supported	UINT8	0..2	2	No	ro
	1	COB-ID used by PDO	UINT32	--	380h + Node-ID	No	ro
	2	Transmission type	UINT8	0..FFh	0	No	rw
1803h		Transmit PDO parameter4	RECORD				
	0	largest sub-index supported	UINT8	0..2	2	No	ro
	1	COB-ID used by PDO	UINT32	--	480h + Node-ID	No	ro
	2	Transmission type	UINT8	0..FFh	0	No	rw
1A00h		Transmit PDO mapping 1					
	0	number of mapped PDO	UINT8	0..8	--	No	rw
	1..8	PDO mapping	UINT32	--	--	No	rw
1A01h		Transmit PDO mapping 2					
	0	number of mapped PDO	UINT8	0..8	--	No	rw
	1..8	PDO mapping	UINT32	--	--	No	rw
1A02h		Transmit PDO mapping 3					
	0	number of mapped PDO	UINT8	0..8	--	No	rw
	1..8	PDO mapping	UINT32	--	--	No	rw
1A03h		Transmit PDO mapping 4					
	0	number of mapped PDO	UINT8	0..8	--	No	rw
	1..8	PDO mapping	UINT32	--	--	No	Rw

Table 15: TPDO parameters

Note:

1.The mapping defaults of the *Transmit* PDOs are given in the chapter on the DS-402 protocol

3.11.1 Description of the TPDO parameters

Index (HEX)	Parameter name	Description
1800h..1803h	Transmit PDO mapping	These records contain the transmission COB-IDs of the PDO telegrams and the parameters for configuring the type of transmission.

Table 16: TPDO configuration parameters (transmission)

Sub-index 1 indicates the COB-ID of the transmission PDOs, while sub-index 2 gives the transmission type settable for the TPDOs. On the basis of the value set in sub-index 2 the following types of communication can be obtained:

Transmission type	Description
0	Asynchronous cyclic communication: only mapped data that have been modified are sent to the sync reception.
1..240	Synchronous cyclic communication: In this case the data is always sent cyclically on the basis of the sync ; the numeric value set in <i>Transmission type</i> indicates the sync number after which the data are sent
241..251	Reserved values
252	Synchronous RTR communication: the data are sent to the sync reception after having received an RTR type telegram (<i>Remote Transmission Request</i>) with the same COD-ID as the TPDO to be sent.
253	Asynchronous RTR communication: the data are sent when an RTR type telegram (<i>Remote Transmission Request</i>) with the same COD-ID as the TPDO to be sent is received.
254	Asynchronous communication (manufacturer specific): option reserved for specific drive communications; in this case the DGM drive behaves in the same way as for option 255.
255	Asynchronous communication (device profile specific) communication managed on the basis of asynchronous events; the mapped data of the drive are sent when an RTR type telegram is received, just like option 253.

Table 17: Types of transmission in the transmission PDOs

3.11.2 Description of the TPDO mapping parameters

Index (HEX)	Parameter name	Description
1A00h..1A03h	Transmit PDO mapping	Record of parameters for setting the PDO content in transmission; dynamic mapping is permitted in the DGM drive.

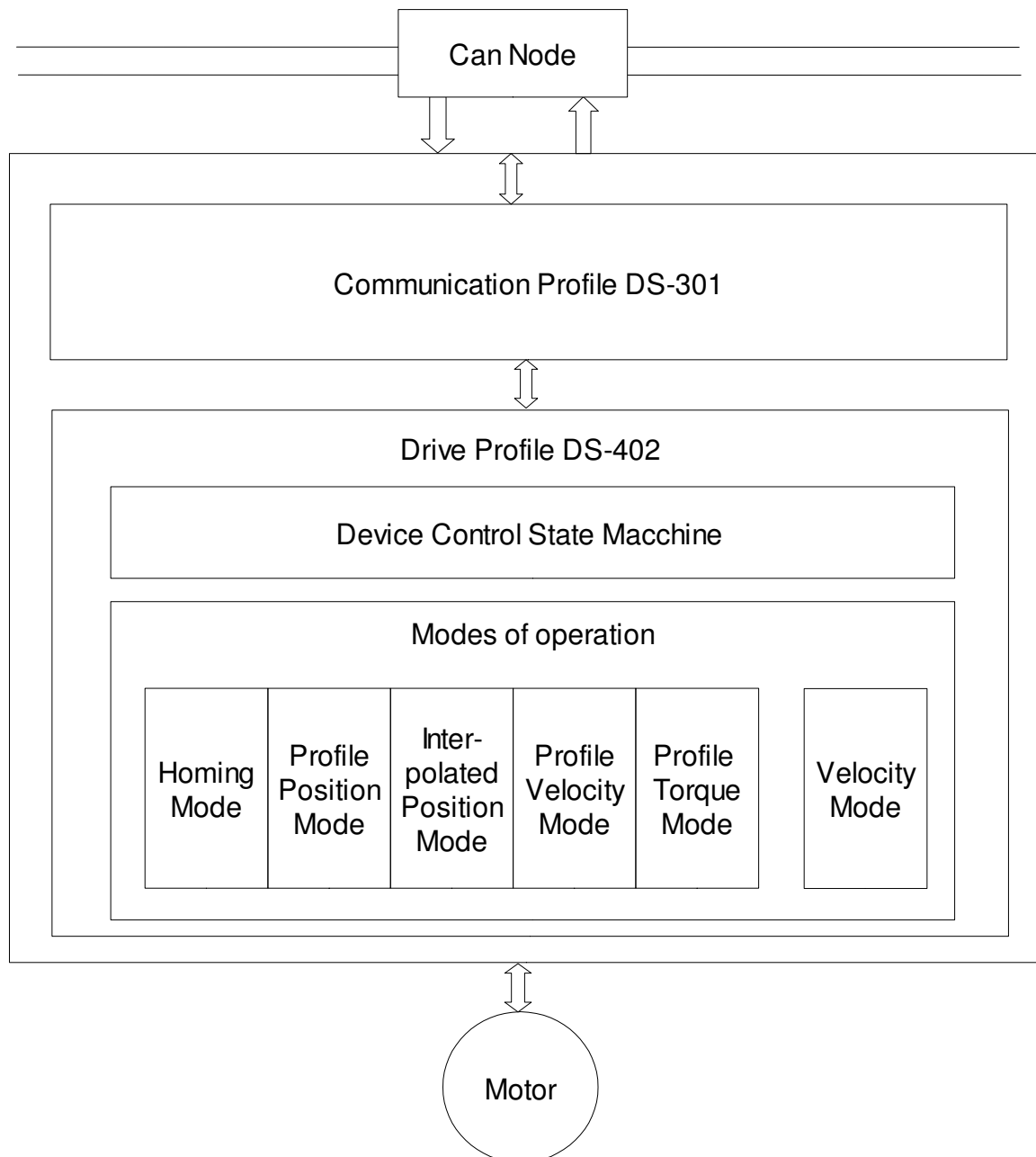
Table 18: mapping parameters (transmission)

4 DS-402 Protocol Parameters

4.1 DS-402 Standard

The DS-402 defines a set of common variables for setting the parameters of and managing drives for motor control, and in particular for the following types of control:

- Homing Position
- Position Profile (*Profile Position mode*)
- Position Interpolation (*Interpolated Position mode*)
- Speed profile (*Profile Velocity Mode*)
- Torque Profile (*Profile Torque Mode*)
- Speed Mode (*Velocity Mode*)



Drawing 3: Communication architecture

4.2 State Machine

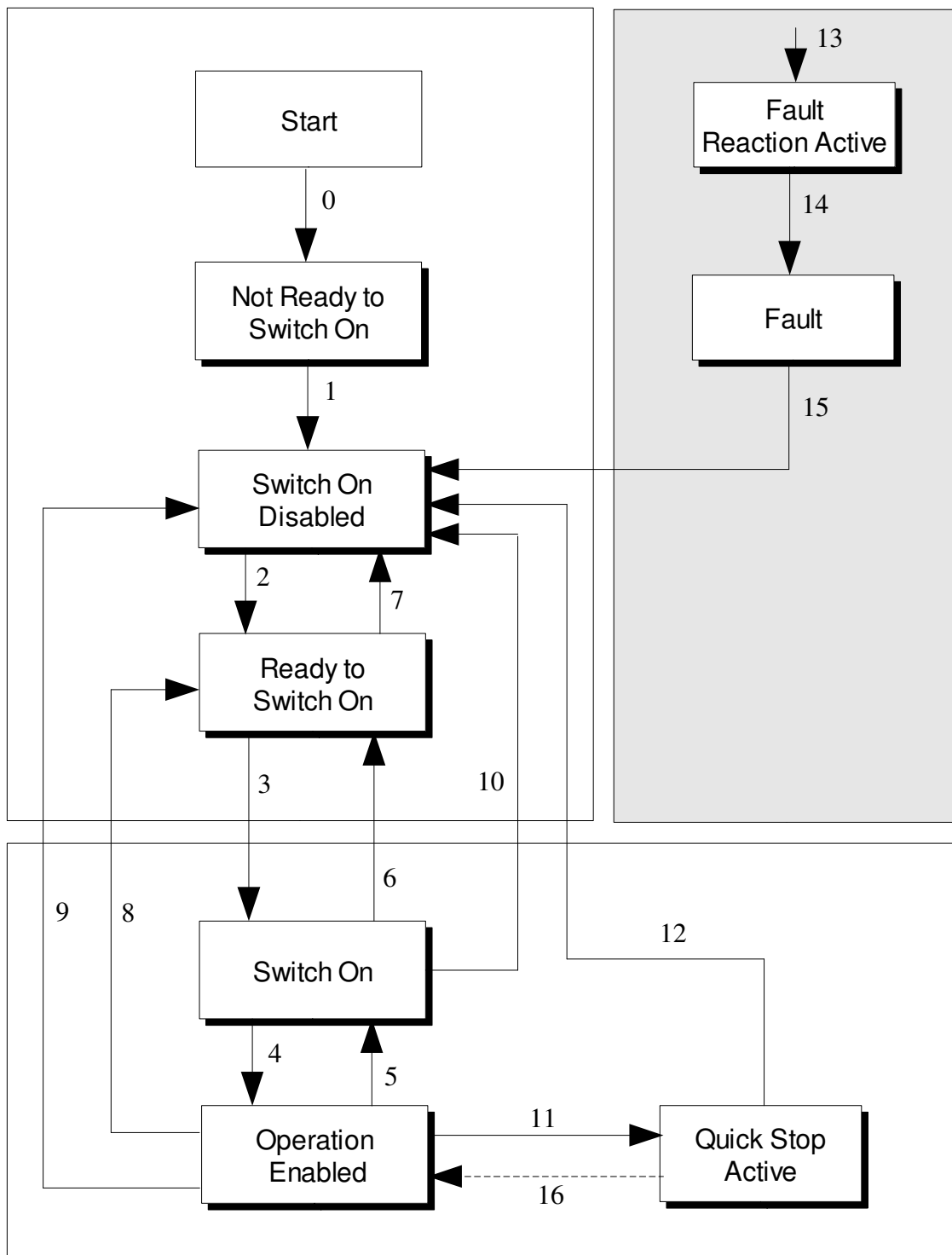
The drive state is managed by a state machine that controls the enabling and disabling phases of the drive and the *fault* caused by any drive alarm.

State	Description
NOT READY TO SWITCH ON	The drive is not ready, because it is in the loading and parameter initialisation phases.
SWITCH ON DISABLE	<ul style="list-style-type: none">•Parameter initialisation has terminated.•The values can be modified•The power is disconnected
READY TO SWITCH ON	<ul style="list-style-type: none">•The drive can be powered.•The values can be modified•The power is disconnected
SWITCH ON	<ul style="list-style-type: none">•The drive is being powered•The drive is ready•The values can be modified•The power is disconnected
OPERATION ENABLED	<ul style="list-style-type: none">•There is no drive alarm•The drive is ready•The values can be modified•The power is enabled•The motor is being powered
QUICK STOP ACTIVE	<ul style="list-style-type: none">•The master has sent the stop command, which is carried out by the preset emergency function•The values can be modified•The power is enabled•The motor is being powered
FAULT REACTION ACTIVE	<ul style="list-style-type: none">•An alarm has caused the preset emergency function to cut in•The values can be modified
FAULT	<ul style="list-style-type: none">•There is an alarm at the drive and the <i>fault reaction</i> function has terminated•The values can be modified•The power is disconnected

Table 19: State description (State Machine)

All the various phases that vary the drive state are given in *statusword*, a read only 16 bit variable that can be found in the 6041h index of the Object Dictionary. Reading this variable it is possible to check the state of the state machine at any moment.

The commands that allow communication between one state and another are instead managed by the master and are set by means of the *controlword*. This 16 bit variable can be found in index 6040h of the *Object Dictionary*.



Drawing 4: State Machine DS-402

4.3 PDO default mapping settings

As described in the communication parameters, the DGM has 4 RPDOs and 4 TPDOs

with specifically set default values. As the PDOs can be dynamically remapped, the preset data can be modified.

4.3.1 RPDO 2 default mapping (index 1600h)

Index	Sub-Index	Name	Default Value
1600h		<i>Receive PDO 1</i>	
	0	<i>Number of mapped objects</i>	3
	1	<i>Controlword</i>	6040 0010h
	2	<i>Target velocity (vI)</i>	6042 0010h
	3	<i>Home offset</i>	607C 0020h
	4..8	--	0

4.3.2 RPDO 2 default mapping (index 1601h)

Index	Sub-Index	Name	Default Value
1601h		<i>Receive PDO 2</i>	
	0	<i>Number of mapped objects</i>	2
	1	<i>Target position</i>	607A 0020h
	2	<i>Profile velocity</i>	6081 0020h
	3..8	--	0

4.3.3 RPDO 3 default mapping (index 1602h)

Index	Sub-Index	Name	Default Value
1602h		<i>Receive PDO 3</i>	
	0	<i>Number of mapped objects</i>	2
	1	<i>Profile acceleration</i>	6083 0020h
	2	<i>Profile deceleration</i>	6084 0020h
	3..8	--	0

4.3.4 RPDO 4 default mapping (index 1603h)

Index	Sub-Index	Name	Default Value
1603h		<i>Receive PDO 4</i>	
	0	<i>Number of mapped objects</i>	1
	1	<i>Interpolation data record (sub-index 1 Position)</i>	60C1 0120h
	2..8	--	0

4.3.5 TPDO 1 default mapping (index 1A00h)

Index	Sub-Index	Name	Default Value
1A00h		<i>Transmit PDO 1</i>	
	0	<i>Number of mapped objects</i>	3
	1	<i>Statusword</i>	6041 0010h
	2	<i>Modes of operation display</i>	6061 0008h
	3	<i>Error Register</i>	1001 0008h
	4..8	--	0

4.3.6 TPDO 2 default mapping (index 1A01h)

Index	Sub-Index	Name	Default Value
1A01h		<i>Transmit PDO 2</i>	
	0	<i>Number of mapped objects</i>	2
	1	<i>Position demand value</i>	6062 0020h
	2	<i>Position actual value</i>	6064 0020h
	3..8	--	0

4.3.7 TPDO 3 default mapping (index 1A02h)

Index	Sub-Index	Name	Default Value
1A02h		<i>Transmit PDO 2</i>	
	0	<i>Number of mapped objects</i>	2
	1	<i>vl velocity demand</i>	6043 0010h
	2	<i>vl control effort</i>	6044 0010h
	3..8	--	0

4.3.8 TPDO 4 default mapping (index 1A03h)

Index	Sub-Index	Name	Default Value
1A03h		<i>Transmit PDO 2</i>	
	0	<i>Number of mapped objects</i>	1
	1	<i>Digital inputs</i>	60FD 0020h
	2..8	--	0

4.4 DS-402 Control Variables

In addition to *controlword* and *statusword*, the DS-402 includes a set of variables that can be used to set the type of control and behaviour management of the drive in case of *fault*.

Index	Name	Type	Map	Attributes
6040h	<i>Controlword</i>	UINT16	Yes	rw
6041h	<i>Statusword</i>	UINT16	Yes	ro
6007h	<i>Abort connection option code</i>	INT16	No	rw
603Fh	<i>Error code</i>	UINT16	Yes	ro
6050h	<i>vl slow down time</i>	UINT32	Yes	rw
6051h	<i>vl quick stop time</i>	UINT32	Yes	rw
605Ah	<i>Quick stop option code</i>	INT16	No	rw
605Dh	<i>Halt option code</i>	INT16	No	rw
6060h	<i>Modes of operation</i>	INT8	Yes	rw
6061h	<i>Modes of operation display</i>	INT8	Yes	ro
60FDh	<i>Digital inputs</i>	UINT32	Yes	ro

Table 20: Control parameters

4.4.1 Object 6040h - *Controlword*


Index	Name	Type	Range	Default	Unit	Map	Attributes
6040h	<i>Controlword</i>	UINT16	0..FFFFh	0		Yes	rw

controlword bit description:

15..11	10..9	8	7	6..4	3	2	1	0
<i>manufacturer specific</i>	<i>reserved</i>	<i>halt</i>	<i>Fault reset</i>	<i>Operation mode specific</i>	<i>Enable operation</i>	<i>Quick stop</i>	<i>Enable voltage</i>	<i>Switch on</i>

4.4.1.1 *Controlword Bits 0..3 and 7*

Bits 0..3 and 7 manage the commands for *state machine* state commutation on the basis of the following table:

Commads	Bit of the <i>controlword</i>					Transitions
	<i>Fault reset</i>	<i>Enable operation</i>	<i>Quick stop</i>	<i>Enable voltage</i>	<i>Switch on</i>	
Shutdown	0	x	1	1	0	2,6,8
Switch on	0	0	1	1	1	3
Switch on	0	1	1	1	1	3
Disable voltage	0	x	x	0	x	7,9,10,12
Quick stop	0	x	0	1	x	7,10,11
Disable operation	0	0	1	1	1	5
Enable operation	0	1	1	1	1	4,16
Fault reset		x	x	x	x	15

4.4.1.2 *Controlword Bits 4,5,6 and 8*

Bits relative to the operation mode:

Bit	Operation mode					
	<i>Velocity mode</i>	<i>Profile position mode</i>	<i>Profile velocity mode</i>	<i>Profile torque mode</i>	<i>Homing mode</i>	<i>Interpolation position mode</i>
4	<i>rfg enable</i>	<i>New set-point</i>	<i>reserved</i>	<i>reserved</i>	<i>Homing operation start</i>	<i>Enable ip mode</i>
5	<i>rfg unlock</i>	<i>Change set immediately</i>	<i>reserved</i>	<i>reserved</i>	<i>reserved</i>	<i>reserved</i>
6	<i>rfg use ref</i>	<i>abs / rel</i>	<i>reserved</i>	<i>reserved</i>	<i>reserved</i>	<i>reserved</i>
8	<i>Halt</i>	<i>Halt</i>	<i>Halt</i>	<i>Halt</i>	<i>Halt</i>	<i>Halt</i>

Table 21: controlword – operation mode bit

4.4.2 Object 6041h - *Statusword*

Index	Name	Type	Range	Default	Unit	Map	Attributes
6041h	<i>statusword</i>	UINT16	0..FFFFh	0		Yes	ro

Statusword bit description:

Bit	Description
0	<i>Ready to switch on</i>
1	<i>Switched on</i>
2	<i>Operation enabled</i>
3	<i>Fault</i>
4	<i>Voltage enabled(*)</i>
5	<i>Quick stop</i>
6	<i>Switch on disabled</i>
7	<i>Warning</i>
8	<i>Manufacturer specific</i>
9	<i>Remote</i>
10	<i>Target reached</i>
11	<i>Internal limit active</i>
12..13	<i>Operation mode specific</i>
14..15	<i>Manufacturer specific</i>

Table 22: status word - bit description

(*)When bit 4 of *status word* is 1 it means that motor has power ON.

4.4.2.1 *Statusword Bits regarding the drive state*

State	<i>Statusword</i> bit
NOT READY TO SWITCH ON	xxxx xxxx x0xx 0000
SWITCH ON DISABLE	xxxx xxxx x1xx 0000
READY TO SWITCH ON	xxxx xxxx x01x 0001
SWITCH ON	xxxx xxxx x01x 0011
OPERATION ENABLED	xxxx xxxx x01x 0111
QUICK STOP ACTIVE	xxxx xxxx x00x 0111
FAULT REACTION ACTIVE	xxxx xxxx x0xx 1111
FAULT	xxxx xxxx x0xx 1000

Table 23: status word, bits relative to the state

The bits indicated with value x are irrelevant

4.4.2.2 Statusword Bits 12 and 13 - operation mode

Bit	Operation mode					
	<i>Velocity mode</i>	<i>Profile position mode</i>	<i>Profile velocity mode</i>	<i>Profile torque mode</i>	<i>Homing mode</i>	<i>Interpolation position mode</i>
12	<i>reserved</i>	<i>Set-point acknowledge</i>	<i>Speed</i>	<i>reserved</i>	<i>Homing attained</i>	<i>ip mode active</i>
13	<i>reserved</i>	<i>Following error</i>	<i>Max slippage error</i>	<i>reserved</i>	<i>Homing error</i>	<i>reserved</i>

Table 24: status word, bits relative to the operation mode

4.4.3 Object 6007h - *Abort connection option code*

Index	Name	Type	Values	Default	Unit	Map	Attributes
6007h	<i>Abort connection option code</i>	UINT16	0 <i>no action</i>	0		No	rw
			1 <i>malfunction</i>				
			2 <i>Device control command 'Disable Voltage'</i>				
			3 <i>Device control command 'Quick Stop'</i>				

This parameter indicates the action to be taken if communication with the network is lost.

Values	Abort connection option code	Description
0	<i>no action</i>	No action is taken
1	<i>malfunction</i>	<ul style="list-style-type: none"> • Power is disconnected and the motor is free to turn • The drive moves to the Switch On Disabled state
2	<i>Device control command 'Disable Voltage'</i>	<ul style="list-style-type: none"> • Power is disconnected and the motor is free to turn • The drive moves to the Switch On Disabled state
3	<i>Device control command 'Quick Stop'</i>	This give a <i>Quick Stop</i> with a deceleration calculated dividing <i>vl velocity reference</i> (index 604Eh) with parameter <i>vl quick stop time</i> (index 6051h) at the end the drive remains stopped in torque in the <i>Operation Enabled</i> state.

4.4.4 Object 603Fh - *Error code*

Index	Name	Type	Range	Default	Unit	Map	Attributes
603Fh	<i>Error code</i>	UINT16	0..FFFFh	0		Yes	ro

The last drive alarm is saved in this variable. The error code corresponds to the 16 lower bits of the 32 bit value found at sub-index 1 of the index 1003h *pre-defined error field* object.

4.4.5 Object 604Eh – *vl velocity reference*

Index	Name	Type	Range	Default	Unit	Map	Attributes
604Eh	<i>VI velocity reference</i>	UINT32	0..7FFFFFFFh	3000		No	rw

In the drive DGM the *vl velocity reference* is used to calculate the deceleration of the “Quick stop” or the “Halt”. This parameter have the same unit of measure of *vl target velocity* (index 6042h) and it's multiplied for the “*Numerator*” and divided for the “*Denominator*” of *vl dimension factor* (index604Ch) to obtain the speed in rpm.

4.4.6 Object 605Ah - *Quick stop option code*

Index	Name	Type	values		Default	Unit	Map	Attributes
605Ah	Quick stop option code	INT16	0	<i>disable drive function</i>	0		Yes	rw
			1	<i>slow down on slow down ramp</i>				
			2	<i>slow down on quick stop ramp</i>				
			5	<i>slow down on slow down ramp and stay in QUICK STOP</i>				
			6	<i>slow down on quick stop ramp and stay in QUICK STOP</i>				

The *Quick stop option code* parameter determines the action to be undertaken if the *Quick Stop Active* state is set on the drive. The following values can be set for the DGM drive.

4.4.7 Object 605Dh - *Halt option code*

Index	Name	Type	Values		Default	Unit	Map	Attributes
605Dh	Halt option code	INT16	0	<i>disable drive, motor is free to rotate</i>	0		Yes	rw
			1	<i>slow down on slow down ramp</i>				
			2	<i>slow down on quick stop ramp</i>				

The *Halt option code* parameter can be used to select which action to carry out when bit 8 (preset for halt function) of the *controlword* is set at 1.

In the DGM drive, the following values can be assigned to this parameter:

Values	<i>Halt option code</i>	Description
0	<i>disable drive, motor is free to rotate</i>	<ul style="list-style-type: none"> • Power is disconnected and the motor is free to turn • The drive moves to <i>Switch On Disabled</i>

1	<i>slow down on slow down ramp</i>	<ul style="list-style-type: none"> • The drive moves to speed control and a stop in ramp is commanded; ramp deceleration is calculated dividing <i>vl velocity reference</i> (index 604Eh) with <i>vl slow down time</i> index 0x6050 parameter. • At ramp termination the drive remains in the <i>Operation Enabled</i> state.
2	<i>slow down on quick stop ramp</i>	<ul style="list-style-type: none"> • The drive moves to speed control and a stop in ramp is commanded; ramp deceleration is calculated dividing <i>vl velocity reference</i> (index 604Eh) with <i>vl quick stop time</i> index 0x6051h parameter. • At ramp termination the drive remains in the <i>Operation Enabled</i> state.

4.4.8 Object 6050h - *vl slow down time*

Index	Name	Type	Range	Default	Unit	Map	Attributes
6050h	<i>vl slow down time</i>	UINT32	0..60000	1000	ms	Yes	rw

This parameter indicates the time used to calculate the deceleration obtained with the division *vl velocity reference* (index 604Eh) and *vl slow down time*; the motor shut down is managed in speed control with constant deceleration ramp.

4.4.9 Object 6051h - *vl quick stop time*

Index	Name	Type	Range	Default	Unit	Map	Attributes
6051h	<i>vl quick stop time</i>	UINT32	0..60000	1000	ms	Yes	rw

This parameter is used to calculate the deceleration in case of a *Quick Stop* command. The value of the ramp is given dividing *vl velocity reference* (index 604Eh) and *vl quick stop time*.

The options that may require the use of *vl quick stop time* are specified in the *Quick stop option code* at index 605Ah.

4.4.10 Object 6060h - *Modes of operation*

Index	Name	Type	Values		Default	Unit	Map	Attributes
6060h	<i>Modes of operation</i>	INT8	1	<i>Profile Position Mode</i>	2		Yes	rw

			2	<i>Velocity Mode</i>				
			3	<i>Profile Velocity Mode</i>				
			6	<i>Homing Mode</i>				
			7	<i>Interpolated Position Mode</i>				

This parameter can be used to set the type of control required according to application requirements. The permitted values of the DGM drive are given below.

Values	<i>Modes of operation</i>	Description
-1	<i>Electric Axis</i>	This type of control allows to follow a position reference. The position reference is given by the product of the number of pulses of the external encoder and the electric axis ratio.
1	<i>Profile Position Mode</i>	The current, speed and position loops are activated in this mode; by setting the required quota the drive can carry out the movement with a trapezoidal profile or with "S" ramps regulated by the Jerk. The jerk can be set using the keyboard in parameters P.5000 and P.5001 (see DGM manual)
2	<i>Velocity Mode</i>	The current and speed loops are enabled and the drive follows the speed reference sent by the master according to the " <i>Velocity Mode</i> " parameters
3	<i>Profile Velocity Mode</i>	The current and speed loops are set and the drive follows the speed reference sent by the master on the basis of the " <i>Profile Velocity Mode</i> " parameters
4	<i>Profile Torque Mode</i>	In this mode, the torque is given controlling the quadrature current which is adjusted from the control loop that supply the torque current to the motor.
6	<i>Homing Mode</i>	The current, speed and position loops are activated, the drive prepares itself to carry out a <i>home position</i> on the basis of the set parameters.
7	<i>Interpolated Position Mode</i>	The current, speed and position loops are activated and the drive carries out position data interpolation sent by the master.

4.4.11 Object 6061h - ***Modes of operation display***

Index	Name	Type	Range	Default	Unit	Map	Attributes
6061h	<i>Modes of operation display</i>	INT8				Yes	ro

This parameter gives the operation mode, namely the type of control currently active in the drive, set with the *Modes of operation* index 6060h parameter.

4.4.12 Object 60FDh - *Digital inputs*

Index	Name	Type	Range	Default	Unit	Map	Attributes
60FDh	Digital inputs	UINT32	-----	0		Yes	ro

The DGM drive has 8 digital inputs, and these bits are mapped between bit 16 and bit 24 of the variable at index 60FDh.

The CW, CCW and Home limit switches are used for homing and must be connected at a hardware level to the preset inputs, which are:

- **Input 1:** for the CW limit switch
- **Input 2:** for the CCW limit switch
- **Input 6:** for the home limit switch

In addition to being at bits 17, 18 and 23, the preset inputs for the limit switches are also at bits 0, 1, 2 as indicated in the DS-402 standard.

To connect the inputs refer to the drive hardware manual.

Bit Input	Description
0	CCW limit switch input
1	CW limit switch input
2	Home limit switch input
0..15	Not used
16	Input 0: Power Enable consent (if enabled)
17	Input 1: CW limit switch
18	Input 2: CCW limit switch
19	Input 3: generic
20	Input 4: generic
21	Input 5: generic
23	Input 6: home limit switch
24	Input 7: generic
25..31	Not used

Table 25: DGM Drive input

Input 0 can be set in parameter I.0000 using the interface prepared for managing the DGM drive parameters (only active when CanOpen control has been enabled through parameter **S.70000**).

If set at “on”, this input allows passage to the “Operation Enable” state but only if a 24 V voltage is present at input. If the drive is in the “Operation Enable” state and the input is

taken to zero voltage, the drive moves to the “Switched On” state independently from the value of the control word. If parameter **I.0000** is set at “off”, the voltage entering input 0 is irrelevant.

4.5 Conversion Factors

This section gives the conversion factors used by the DGM drive to condition the speed and acceleration position variables.

Index	Name	Type	Map	Attributes
6089h	<i>Position notation index</i>	INT8	No	rw
608Ah	<i>Position dimension index</i>	INT8	No	rw
608Bh	<i>Velocity notation index</i>	INT8	No	rw
608Ch	<i>Velocity dimension index</i>	INT8	No	rw
608Dh	<i>Acceleration notation index</i>	INT8	No	rw
608Eh	<i>Acceleration dimension index</i>	INT8	No	rw
6093h	<i>Position factor</i>	(ARRAY) UINT32	No	rw
6094h	<i>Velocity encoder factor</i>	(ARRAY) UINT32	No	rw
6097h	<i>Acceleration factor</i>	(ARRAY) UINT32	No	rw
607Eh	<i>Polarity</i>	UINT8	No	rw

Table 26: Conversion Factors

4.5.1 Object 604Ch – *vl dimension factor*

Index	Sub Index	Name	Type	Range	Default	Unit	Map	Attributes
604Ch		<i>VI dimension factor</i>	ARRAY					
	0	<i>Number of entries</i>	UINT8	2	2		No	ro
	1	<i>Numerator</i>	INT32	±7FFFFFFFh(*)	1		No	rw
	2	<i>Denominator</i>	INT32	±7FFFFFFFh(*)	1		No	rw

(*) Excluded zero value.

The *vl dimension factor* is composed by two values of 32 bit, the “numerator” that multiply *vl target velocity* (index 6042h) and “Denominator” that divides the result obtained. These operations are used to convert *vl target velocity* to rpm unit. The value obtained is the input reference of the generator of speed ramps.

$$speed\ reference\ [rpm] = \frac{(Numerator \times vl\ target\ velocity)}{(Denominator)}$$

4.5.2 Object 6089h - *Position notation index*

Index	Name	Type	Range	Default	Unit	Map	Attributes
6089h	<i>Position notation index</i>	INT8	0	0	--	No	rw

This parameter indicates the size order of the position unit of measurement. In the DGM drive this parameter is set at value 0. *Position Factor* (index 6093h) can be used to vary the position size order.

4.5.3 Object 608Ah - *Position dimension index*

Index	Name	Type	Range	Default	Unit	Map	Attributes
608Ah	<i>Position dimension index</i>	INT8	ACh	ACh	--	No	rw

Position dimension index defines the position unit of measurement. This parameter is fixed at ACh, which in the drive represents resolver impulses. To modify the unit of measurement use *Position Factor* (index 6093h).

Note:

1. One rotation of the motor axis corresponds to 65536 resolver impulses.

4.5.4 Object 608Bh - *Velocity notation index*

Index	Name	Type	Range	Default	Unit	Map	Attributes
608Bh	<i>Velocity notation index</i>	INT8	0	0	--	No	rw

This parameter indicates the size order of the position unit of measurement. In the DGM drive this value is fixed at 0. To vary the size order of the speed use the *Velocity encoder factor* (index 6094h).

4.5.5 Object 608Ch - *Velocity dimension index*

Index	Name	Type	Range	Default	Unit	Map	Attributes
608Ch	<i>Velocity dimension index</i>	INT8	A4h	A4h	--	No	rw

Velocity dimension index defines the unit of measurement of the speed. This parameter is fixed at A4h, which in the DGM drive represents revolutions/s [rpm]. To modify the unit of measurement, use *Velocity encoder factor* (index 6094h).

4.5.6 Object 608Dh - *Acceleration notation index*

Index	Name	Type	Range	Default	Unit	Map	Attributes
608Dh	<i>Acceleration notation index</i>	INT8	0	0	--	No	rw

This parameter indicates the size order of the acceleration and deceleration unit of measurement. In the DGM drive this parameter is fixed at 0. To vary the acceleration size order use *Acceleration factor* (index 6097h).

4.5.7 Object 608Eh - *Acceleration dimension index*

Index	Name	Type	Range	Default	Unit	Map	Attributes
608Eh	<i>Acceleration dimension index</i>	INT8	A3h	A3h	--	No	rw

Acceleration dimension index defines the acceleration and deceleration unit of measurement. This parameter is fixed at A3h, which in the DGM drive represents revolutions/s² or [rpm/s]. To modify the unit of measurement, use *Acceleration factor* (index 6097h)

4.5.8 Object 6093h - *Position factor*

Index	Sub-Index	Name	Type	Range	Default	Unit	Map	Attributes
6093h		<i>Position factor</i>	ARRAY					
	0	<i>number of entries</i>	UINT8	0..2	2		No	ro
	1	<i>Numerator</i>	UINT32	1..FFFFFFFF	65536		No	rw
	2	<i>Feed constant</i>	UINT32	1..FFFFFFFF	65536		No	rw

The *Position factor* includes two 32 bit values that are used to transform the quota to be reached at the internal position of the drive. The operation is carried out multiplying the *Numerator* by the *Target position* (index 607Ah), and the result obtained is divided by the *Feed constant*.

$$internal\ position\ reference = \frac{(Numerator \times Target\ position)}{(Feed\ constant)}$$

In practice, in the DGM the *Numerator* corresponds to the number of resolver impulses per revolution, which equals 65536.

Feed constant instead represents a numeric value that expresses the space covered by one motor axis revolution in the same unit of measurement as the *Target position*.

Modifying the *Numerator* and *Feed constant* values influences the value of the internal position reference calculation, therefore the quota unit of measurement.

4.5.9 Object 6094h - *Velocity encoder factor*

Index	Sub-Index	Name	Type	Range	Default	Unit	Map	Attributes
6094h		<i>Velocity encoder factor</i>	ARRAY					
	0	<i>number of entries</i>	UINT8	0..2	2		No	ro
	1	<i>Numerator</i>	UINT32	1..FFFFFFFF	65536	--	No	rw
	2	<i>Divisor</i>	UINT32	1..FFFFFFFF	60	--	No	rw

The *velocity encoder factor* converts the speed value contained in the *Profile velocity* (index 6081h) into the drive internal speed reference which, for the DGM drive, is expressed in resolver impulses at the second. The conversion ratio is given by:

$$internal\ speed\ reference = \frac{(Numerator \times Profile\ velocity)}{Divisor}$$

In the DGM drive, the *Profile velocity* unit of measurement is preset in [rpm]. Conversion into resolver impulses at the second can be obtained by setting *Numerator*=65536 and *Divisor*=60.

4.5.10 Object 6097h - *Acceleration factor*

Index	Sub-Index	Name	Type	Range	Default	Unit	Map	Attributes
6097h		<i>Acceleration factor</i>	ARRAY					
	0	<i>number of entries</i>	UINT8	0..2	2		No	ro
	1	<i>Numerator</i>	UINT32	1..FFFFFFFF	65536		No	rw
	2	<i>Divisor</i>	UINT32	1..FFFFFFFF	60		No	rw

The *Acceleration factor* converts the acceleration value contained in *Profile acceleration* (index 6083h) and *Profile deceleration* (index 6084h) into the corresponding drive acceleration internal value expressed in resolver impulses on s², using the formula below:

$$\text{internal acceleration reference} = \frac{(\text{Numerator} \times \text{Profile acceleration})}{\text{Divisor}}$$

In the DGM drive the acceleration and deceleration unit of measurement is preset in revolution/s² or [rpm/s]. The unit can be converted into *resolver impulses on s²* by setting *Numerator*=65536 and *Divisor*=60.

4.5.11 Object 607Eh - *Polarity*

Index	Name	Type	Range	Default	Unit	Map	Attributes
607Eh	<i>Polarity</i>	UINT8	0..FF	0	--	No	rw

Variables are associated with the single bits of the *Polarity* parameter. If the bit value is 0 the value of the variable remains unchanged, and if set at 1 the value becomes negative.

Bit	Description
7	<i>Position polarity</i> referred to the <i>Target position</i> (index 607Ah)
6	<i>Velocity polarity</i> referred to the <i>Target velocity</i> (index 60FFh)
0..5	<i>reserved</i>

Table 27: Bit Polarity

4.6 Position Control Parameters

This section gives the position control parameters for the *Profile Position Mode* and the *Interpolated Position Mode*.

Index	Name	Type	Map	Attributes
6062h	<i>Position demand value</i>	INT32	Yes	ro
6063h	<i>Position actual value*</i>	INT32	Yes	ro
6064h	<i>Position actual value</i>	INT32	Yes	ro
6065h	<i>Following error window</i>	UINT32	No	rw
6066h	<i>Following error time out</i>	UINT16	No	rw
607Ch	<i>Home offset</i>	INT32	Yes	rw
607Dh	<i>Software position limit</i>	(ARRAY) UINT32	No	rw
60F4h	<i>Following error actual value</i>	INT32	Yes	ro
60FCh	<i>Position demand value*</i>	INT32	Yes	ro

Table 28: Position control parameters

Note:

1. The parameters indicated by the asterisk represent normalised values, namely values in the internal drive format.

4.6.1 Object 6062h - *Position demand value*

Index	Name	Type	Range	Default	Unit	Map	Attributes
6062h	<i>Position demand value</i>	INT32	INT32			Yes	ro

This parameter represents the position reference supplied at the position loop input. The value read at index 6062h is converted into the *Target position* (index 607Ah) unit of measurement using the *position factor* (index 6093h), using the following formula:

$$\text{Position demand value} = \frac{(\text{Feed constant} \times \text{internal position reference})}{(\text{Numerator})}$$

4.6.2 Object 6063h - *Position actual value**

Index	Name	Type	Range	Default	Unit	Map	Attributes
6063h	<i>Position actual value*</i>	INT32				Yes	ro

This read only variable supplies the position value measured in internal drive units. For the DGM drive this value corresponds to resolver units, where the numeric value of 65536 corresponds to one motor axis revolution.

4.6.3 Object 6064h - *Position actual value*

Index	Name	Type	Range	Default	Unit	Map	Attributes
6064h	<i>Position actual value</i>	INT32				Yes	ro

The variable at index 6064h supplies the measured position converted into the same unit of measurement as the *Target position* (index 607Ah) using the *position factor* (index 6093h); the value of this datum is obtained using drive internal value giving the position measurement:

$$\text{Position actual value} = \frac{(\text{Feed constant} \times \text{internal measured position})}{(\text{Numerator})}$$

The *Feed constant* and *position factor Numerator* default values are both 65536, therefore in this case the *Position actual value* corresponds to that of index 6063h.

4.6.4 Object 6065h - *Following error window*

Index	Name	Type	Range	Default	Unit	Map	Attributes
6065h	<i>Following error window</i>	UINT32	UINT32	910		No	rw

This parameter indicates the maximum position error that can be tolerated in an application that works in position control.

The *Following error window* unit of measurement is the same as that of the *Target position* (index 607Ah); this parameter is converted into drive internal error units using the position factor (index 6093h).

Considering that the default values of the Feed constant and the Numerator are both 65536 and that one revolution of the motor axis corresponds to 65536 resolver impulses, the default value of 910 corresponds to approx. 5°.

Internally, the maximum value of the position error is limited to 32768 resolver impulses, which corresponds to 180°.

4.6.5 Object 6066h - *Following error time out*

Index	Name	Type	Range	Default	Unit	Map	Attributes
6066h	<i>Following error time out</i>	UINT16	1..10000	200	ms	No	rw

This parameter indicates the maximum time error during which the position error can exceed the *Following error window*. If this occurs, bit 13 in the *statusword* is set at one and the drive moves to the *fault* status.

4.6.6 Object 607Ch - *Home offset*

Index	Name	Type	Range	Default	Unit	Map	Attributes
607Ch	<i>Home offset</i>	INT32	0..FFFFFFFF	0		Yes	rw

Home offset indicates the difference in position between the zero position of the application and the zero position of the drive, found using the *Homing Mode*.

This parameter affects on the measured position reported on the parameter “Position actual value” (index 6064h), and the relation is given by the following formula:

$$\text{Position actual value} = \text{sensor position} + \text{home offset}$$

The drive detects the value of the position sensor from the resolver. So this value is a 32bit+sign revised data. For a turn of the sensor correspond 65536 pulses. So it is possible to have ± 32767 round of the axis (in addition to the fraction of 1 turn). The measured position detected by the sensor, is set to zero at every switching on of the drive

or at every home position.

To set to zero the value of parameter *Position actual value* (index 6064h) you need to write in *Home Offset* the the value of the position measured by the sensor with the sign inverted. To obtain this value it is possible to set to 0 (zero) the value *Home offset* then reading the value in *Position actual value* (index 6064h).

In case the bit 7 of parameter *Polarity* (index 607Eh) is active, the value to write on parameter *Home Offset* to set to zero the *Position Actual value* (index 6064h) is always the position measured by the sensor but in this case you don't need to invert the sign.

4.6.7 Oggetto 607Dh - *Software position limit*

Index	Sub-Index	Nome	Tipo	Range	Default	Unit	Map	Attributi
607Dh		<i>Software position limit</i>	ARRAY					
	0	<i>number of entries</i>	UINT8	2	2		No	ro
	1	<i>Min position range limit</i>	INT32	0x80000000..7FFFFFFFh	0x80000000		Si	rw
	2	<i>Max position range limit</i>	INT32	0x80000000..7FFFFFFFh	+7FFFFFFFh		Si	rw

The *software position limit* vector contains two parameters that impose limits on the absolute position when the drive works in a positioner or interpolation, the value to the sub-index 1 represents the minimum value while the value of the sub-index 2 indicates the maximum value; the measure unit is the same of the *Target position* parameter (index 607Ah).

The position limits are activated by placing 1 at the Enable Position Limit parameter (201Ah index), the reporting limit of the reference position in absolute mode relative to the origin, is indicated in bit11 "*internal limit active*" of the status word.

4.6.8 Object 60F4h – *Following error actual value*

Index	Name	Type	Range	Default	Unit	Map	Attributes
60F4h	<i>Following error actual value</i>	INT32	± 7FFFFFFFh			Yes	ro

This parameter give the position error and is converted in the same unit of measure of the *Target position* (index 607Ah) using the *position factor* (index 6093h); the error of position is obtained from the error calculated internally at the drive using the following formula:

$$\text{Following error actual value} = \frac{(\text{Feed constant} \times \text{internal error position})}{(\text{Numerator})}$$

4.6.9 Object 60FCh - *Position demand value**

Index	Name	Type	Range	Default	Unit	Map	Attributes
60FCh	<i>Position demand value</i> *	INT32	± FFFFFFFFh	0		Yes	ro

Value expressed as internal drive units and which corresponds to the position reference at position loop input.

4.7 Positioner (*Profile Position Mode*)

Variable used in position control, positioner mode.

Index	Name	Type	Map	Attributes
607Ah	<i>Target position</i>	INT32	Yes	rw
6081h	<i>Profile velocity</i>	UINT32	Yes	rw
6083h	<i>Profile acceleration</i>	UINT32	Yes	rw
6084h	<i>Profile deceleration</i>	UINT32	Yes	rw
6086h	<i>Motion profile type</i>	INT16	No	rw

Table 29: Positioner parameters

Note: the parameter *Motion profile type* (index 6097h) is set to 3 “*Jerk-limited ramp*”, and it's also used the parameter *Jerk Position* (index 2011h)

4.7.1 Controlword in profile position mode (bit setting)

Bit	Description	
0	<i>Switch on</i>	
1	<i>Enable voltage</i>	
2	<i>Quick stop</i>	
3	<i>Enable operation</i>	
4	New set-point	
	Value	Description
	0	The drive remains stopped
	1	The set quota is acquired and quota start is enabled
5	Change set immediately	
	Value	Description
	0	Quotas carried out in succession, namely the data for the subsequent quota are only accepted when the positioning phase quota has ended.
	1	The quota and speed values are accepted and updated even during the positioning phase. For the DGM drive this mode functions only when the <i>Motion profile type</i> (index 6086h) is set at zero (<i>Linear ramp -trapezoidal profile</i>); in this case bit 6 "abs/rel" is not considered and the movements are carried out in the absolute mode.
6	abs / rel	
	Value	Description
	0	Carries out an absolute movement as regards the application home.
	1	Carries out a movement relative to the previous quota
7	<i>Fault reset</i>	
8	Halt	
	Value	Description
	0	Carries out the positioning quotas
	1	Stop in ramp on the basis of the <i>Profile deceleration</i> (index 6084h) parameter
10..9	<i>Reserved</i>	
15..11	<i>Manufacturer specific</i>	

4.7.2 Statusword in profile position mode bit value

Bit	Description		
0	Ready to switch on		
1	Switched on		
2	Operation enable		
3	Fault		
4	Voltage enabled		
5	Quick stop		
6	Switch on disabled		
7	Warning		
8	Manufacturer specific		
9	Remote		
10	Quota reached (Target reached)		
	Value	Halt controlword bit	Description
	0	0	Set quota not reached
		1	Axis deceleration phase
	1	0	Set quota reached
		1	Axis speed at zero
11	Internal limit active (valid when Enable Position Limit parameter (index 201Ah) is set to 1)		
	Value	Description	
	0	Absolute position reference is not limited by software position limit parameter (index 607Dh)	
	1	Absolute position reference is limited by software position limit parameter (index 607Dh)	
	12	Set-point acknowledge	
Value		Description	
0		Quota value not yet acquired by the function that generates the trajectories.	
1		Quota values acquired by the function that generates the trajectories	
13	Position error (Following error)		
	Value	Description	
	0	No position error found	
	1	Position error found	
14..15	Manufacturer specific		

4.7.3 Manage the Positioner through bit 5 (*Change set immediately*)

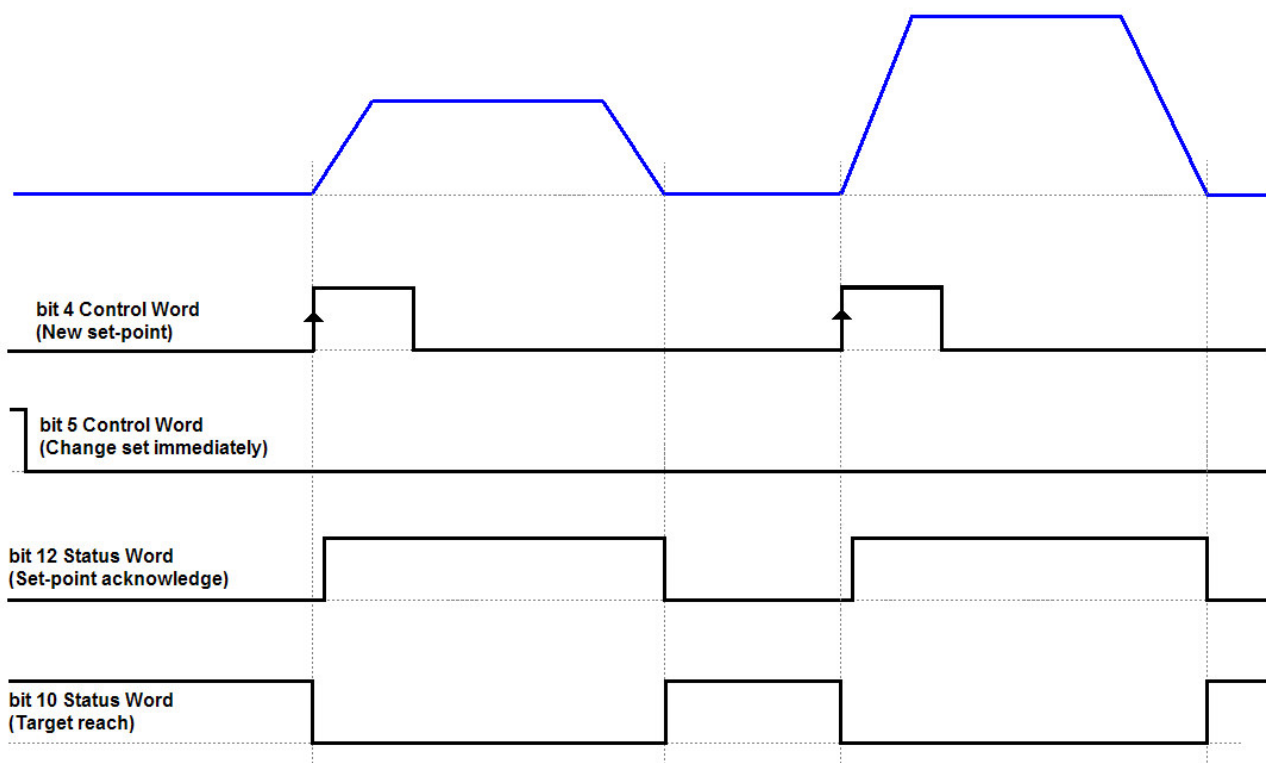
Controlword

Bit 5 (*Change set immediately*) of **Controlword** influences the management of position target. When it is set to zero the positioner is in “**Single Setpoint**” mode, while when it is set to 1 it change to “**Change set immediately**” mode.

In mode “**Single Setpoint**” the drive move from one position to another and a new position is accepted only when the positioning is finished and speed is zero.

The new position is stored and executed only when bit 4 (*New set-point*) of the **controlword** is placed at 1 and the 12 (*Set-point acknowledge*) of the **statusword** is 0, immediately after the bit 12 of the **statusword** is placed at 1 and the bit 10 (*Target reach*) of the **statusword** is set to zero; At the end of the positioning the bit 12 (*Set-point acknowledge*) of the **statusword** is set to zero and the bit 10 (*Target reach*) of the **statusword** is set to 1.

In mode “**Single Setpoint**” the parameter *Motion profile type* (index 6086h) can be set to “*Linear ramp (trapezoidal profile)*” or in “*Jerk-limited ramp*”.

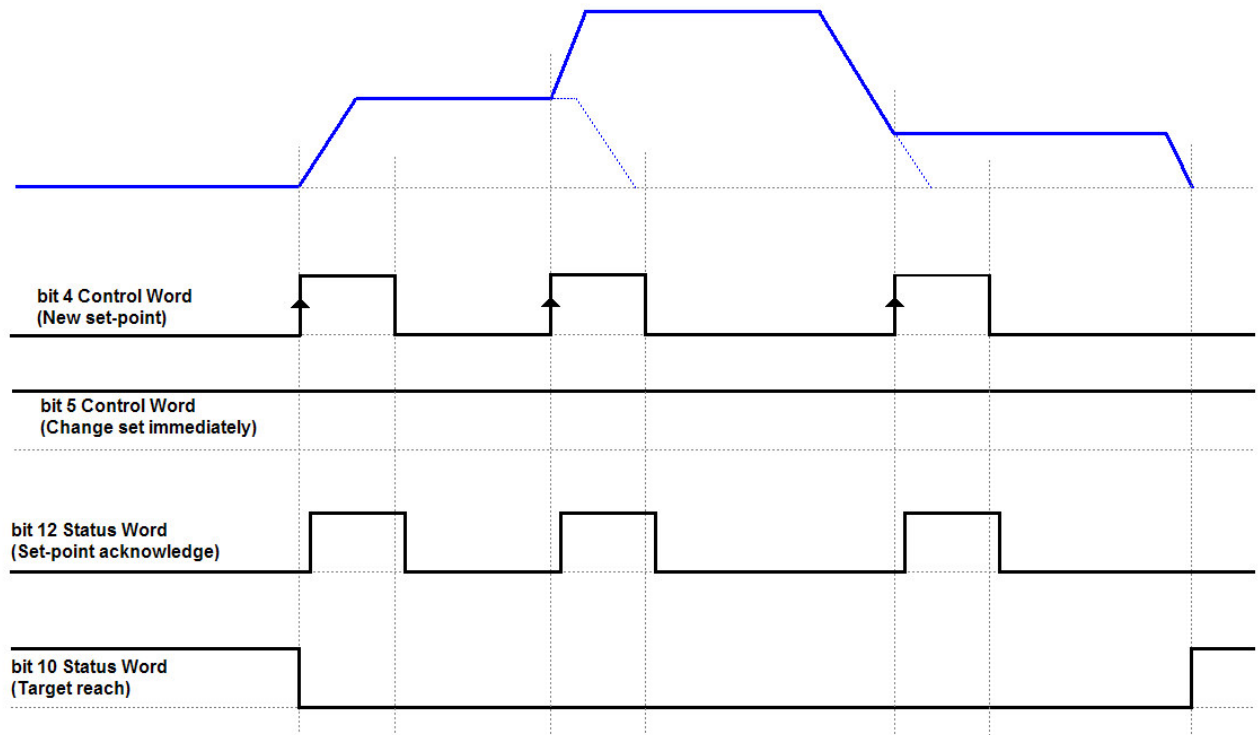


Drawing 5: Mode Single Setpoint

To activate the mode “**Change set immediately**”, give value 1 to bit 5 of the **controlword**. Also in this case on the rising edge of the bit 4 (*New set-point*) of the **controlword** is set the start of the position and the bit 12 (*Set-point acknowledge*) of the **statusword** is set to 1 to indicate that the position has been stored. When the bit 4 of the **controlword**, returns to zero, also the bit 12 (*Set-point acknowledge*) of the **statusword**, returns to

zero to inform that the drive is ready to accept and execute a new position even if the present positioning is not still finished; the bit 10 (*Target reach*) of the **statusword** works as in the previous case.

The mode “**Change set immediately**” works only when parameter *Motion profile type* (index 6086h) is set to “*Linear ramp (trapezoidal profile)*”. If in this case the bit 6 (*abs / rel*) of the **controlword** is set to One, the new position is calculated considering the present position.



Drawing 6: Mode Change Set Immediately

4.7.4 Object 607Ah - *Target position*

Index	Name	Type	Range	Default	Unit	Map	Attributes
607Ah	Target position	INT32	± FFFFFFFFh	0		Yes	rw

Target position indicates the position quota to be reached. The value of this parameter is expressed in resolver impulses¹ and is converted into drive internal units using the data in *position factor* (index 6093h). The conversion formula is:

$$internal\ position\ reference = \frac{(Numerator \times Target\ position)}{(Feed\ constant)}$$

Note:

1. One motor axis revolution corresponds to 65536 resolver impulses.

4.7.5 Object 6081h - *Profile velocity*

Index	Name	Type	Range	Default	Unit	Map	Attributes
6081h	<i>Profile velocity</i>	UINT32	1..9999	1000	rpm	Yes	rw

Profile velocity indicates the value in module of the speed (expressed in [rpm]), at acceleration ramp termination, when a movement profile is carried out to reach a specific quota.

The value of this parameter is converted into drive internal speed units using the data held in the *velocity encoder factor* (index 6094h) *array*. The formula for calculating the internal speed is:

$$internal\ speed\ reference = \frac{(Numerator \times Profile\ velocity)}{Divisor}$$

With the default values the *velocity encoder factor*, that are *Numerator* =65536 and *Divisor*=60, the *Profile Velocity* is expressed in revolution/s or [rpm]

4.7.6 Object 6083h - *Profile acceleration*

Index	Name	Type	Range	Default	Unit	Map	Attributes
6083h	<i>Profile acceleration</i>	UINT32	1..6000000	1000	rpm /s	Yes	rw

Parameter for setting the quota acceleration speed. The datum of this variable (expressed in [rpm/s]) is converted in drive internal units by the *Acceleration factor* (index 6097h), on the basis of this formula:

$$internal\ acceleration\ reference = \frac{(Numerator \times Profile\ acceleration)}{Divisor}$$

Using the default values of the *Acceleration factor*, that are *Numerator* =65536 and *Divisor*=60, the *Profile Acceleration* is expressed in revolution/s² or [rpm/s]

4.7.7 Object 6084h - *Profile deceleration*

Index	Name	Type	Range	Default	Unit	Map	Attributes
6084h	<i>Profile deceleration</i>	UINT32	1..6000000	1000	rpm /s	Yes	rw

Parameter for setting the quota deceleration speed. The datum of this variable (expressed in [rpm/s]) is converted into drive internal units by the *Acceleration factor* (index 6097h), as described for the *Profile acceleration* (index 6083h).

4.7.8 Object 6086h - *Motion profile type*

Index	Name	Type	Range		Default	Unit	Map	Attributes
6086h	<i>Motion profile type</i>	INT16	-1	<i>Linear ramp (trapezoidal profile) with torque limit</i>	3		No	rw
			0	<i>Linear ramp (trapezoidal profile)</i>				
			3	<i>Jerk-limited ramp</i>				

The *motion profile type* makes it possible to select which type of profile to use for carrying out the movements of a quota. This parameter can not be modified during “operational” mode.

Values	<i>Motion profile type</i>	Description
0	<i>Linear ramp (trapezoidal profile)</i>	<ul style="list-style-type: none"> ● In this mode the position profile is generated starting from constant accelerations. This causes a trapezoidal type speed progress, while the position profile assumes a quadratic type progress. ● When bit 5 of the controlword (Change set immediately) is set at 1, the control allows the immediate change of the quota (Target position index 607Ah) and the speed (Profile velocity index 6081h). The speed profile and position are adapted instantaneously, The acceleration ramps (Profile acceleration index 6083h) and deceleration ramps (Profile acceleration index 6084h) are adapted when the speed zeroes. In this case bit 6 “abs/rel” of the control word is not considered and the movements are carried out in absolute mode.
3	<i>Jerk-limited ramp</i>	<ul style="list-style-type: none"> ● With this setting acceleration and deceleration vary in a trapezoidal manner; the result is a quadratic speed profile and position progress of the cubic type. The characteristic of these profiles causes the mechanical stress to decrease as regards <i>Linear ramp</i> positioning, and in addition there is better precision during deceleration when the drive reaches the set quota. ● In this case bit 5 of the controlword (Change set immediately) is not considered. The new quota is only accepted when the previous one has terminated.
-1	<i>Linear Ramp</i>	<ul style="list-style-type: none"> ● This mode works as the mode 0. It make movement

	(trapezoidal profile) with torque limit	<p>using trapezoidals speed profiles. Moreover in case the positioning will be blocked by an obstacle, the algorithmic that generates the profile is interrupted but the drive torque is maintained. This torque can be set using the parameter <i>current limit</i> (<i>index 2006h</i>); when the drive is not more on torque limit, the profile of positioning is resumed to complete the positioning reaching the quota.</p> <ul style="list-style-type: none"> ● In this mode, the error of position can be not measured as the position profile adapts to the measured position when the limit torque intervenes.
--	---	--

4.8 Homing (*Homing Mode*)

This chapter describes the modes and parameters that manage the drive homing procedure. According to the case limit switches may or may not be used to condition the homing function.

Index	Name	Type	Map	Attributes
6098h	<i>Homing method</i>	INT8	No	rw
6099h	<i>Homing speeds</i>	(ARRAY) UINT32	Yes	rw
609Ah	<i>Homing acceleration</i>	UINT32	No	rw

Table 30: Homing parameters

4.8.1 Controlword in Homing Mode (bit setting)

Bit	Description	
0	<i>Switch on</i>	
1	<i>Enable voltage</i>	
2	<i>Quick stop</i>	
3	<i>Enable operation</i>	
4	Homing operation start	
	Value	Description
	0	<i>Homing mode inactive</i>
	0 → 1	Enable <i>home</i> procedure starting
	1	<i>Homing mode active</i>
	1 → 0	Interrupt <i>home</i> procedure
5..6	<i>Reserved</i>	
7	<i>Fault reset</i>	
8	Halt	
	Value	Description
	0	Carry out the instructions of bit 4
	1	Stop in speed ramp, in which the deceleration value corresponds to that of the <i>homing</i> procedure acceleration ramp.
10..9	<i>Reserved</i>	
15..11	<i>Manufacturer specific</i>	

4.8.2 Statusword in Homing Mode (bit value)

Bit	Description		
0	Ready to switch on		
1	Switched on		
2	Operation enable		
3	Fault		
4	Voltage enabled		
5	Quick stop		
6	Switch on disabled		
7	Warning		
8	Manufacturer specific		
9	Remote		
10	Home reached (Target reached)		
	Value	Controlword Halt bit	Description
	0	0	Home position not reached
		1	Axis deceleration phase
	1	0	Home position reached
		1	Axis speed at zero
11	Internal limit active		
12	Homing attained		
	Value	Description	
	0	Home procedure not yet complete	
	1	Home procedure completed correctly	
13	Home Error (Homing error)		
	Value	Description	
	0	No home position error found	
	1	Home position error found	
14..15	Manufacturer specific		

4.8.3 Object 6098h - *Homing method*

Index	Name	Type	Values	Default	Unit	Map	Attributes
6098h	<i>Homing method</i>	INT8	0..35 (except 15,16, 31 and 32)	0		No	rw

The *Homing method* parameter makes it possible to select which method to use for carrying out the homing procedure. The DGM drive accepts all *homing* methods used by the CanOpen, which range from value 0 to value 35, except for methods 15, 16, 31 and 32 which are reserved.

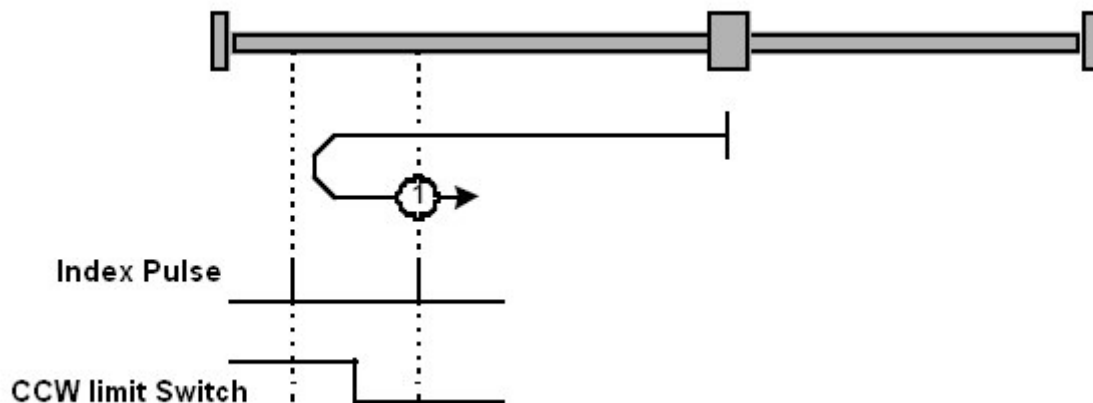
4.8.3.1 Method 0 - *No homing operation required*

At switch on the value of the measured position is zeroed and set as the drive home position.

4.8.3.2 Method 1 - *Homing on the negative limit switch and index pulse*

The drive carries out the home procedure moving in a negative sense towards the CCW limit switch sensor.

Once it has touched the sensor it returns to exit from the limit switch sensor at low speed and always moves in the opposite sense on the resolver index pulse. The point reached in this manner becomes the drive home.

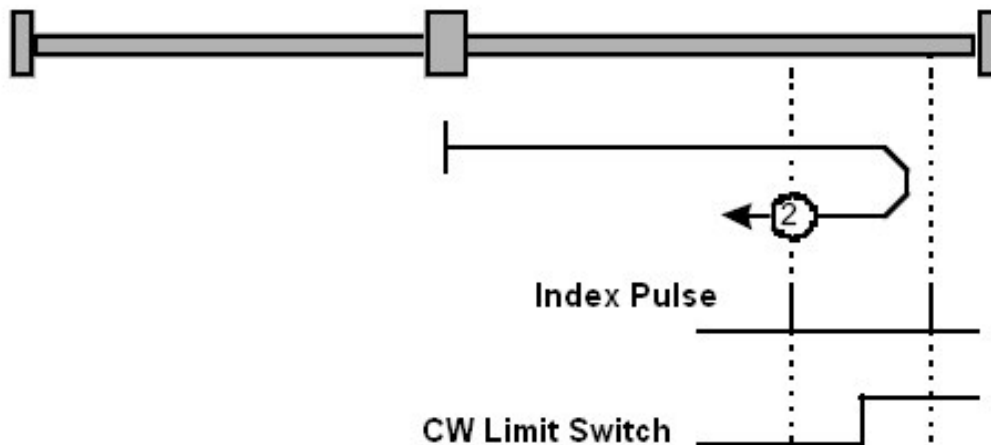


Drawing 7: Method 1 - Homing on CCW limit switch and zero resolver impulse

4.8.3.3 Method 2 - *Homing on the positive limit switch and index pulse*

The drive carries out the home procedure moving in a positive sense towards the CW limit switch sensor.

Once it has touched the sensor it returns to exit from the limit switch sensor at low speed and always moves in the opposite sense on the resolver index pulse. The point reached in this manner becomes the drive home.



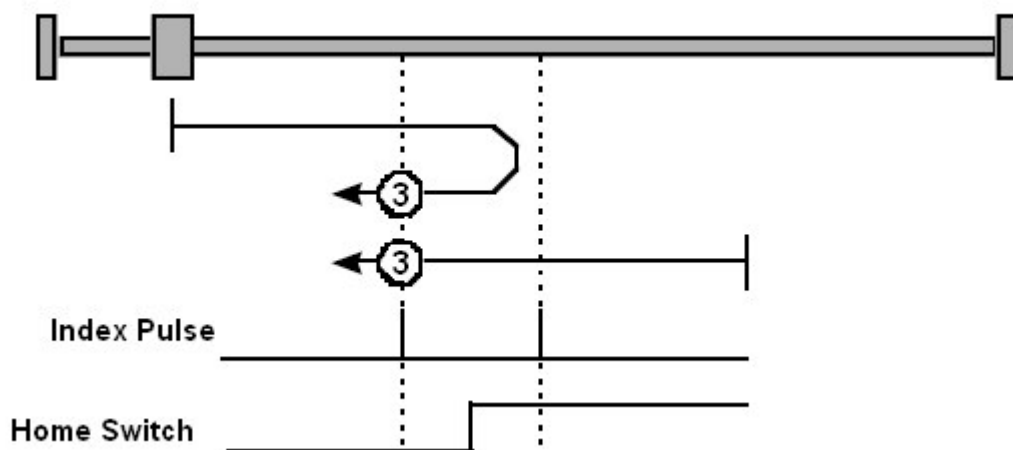
Drawing 8: Method 2 - Homing on CW limit switch and zero resolver impulse

4.8.3.4 Method 3 - Homing on the positive home mark and index pulse

The home sensor input state determines the search direction of the sensor.

If the home input is low the motor is commanded to rotate clockwise. When home sensor commutation is found the motor is stopped and subsequently positioned at the resolver home index pulse with an anticlockwise movement.

If the home input is high the motor is commanded to rotate anticlockwise. When home sensor commutation is found the motor is stopped and subsequently positioned at the resolver home index pulse with a clockwise movement.



Drawing 9: Method 3 - Homing in home sensor

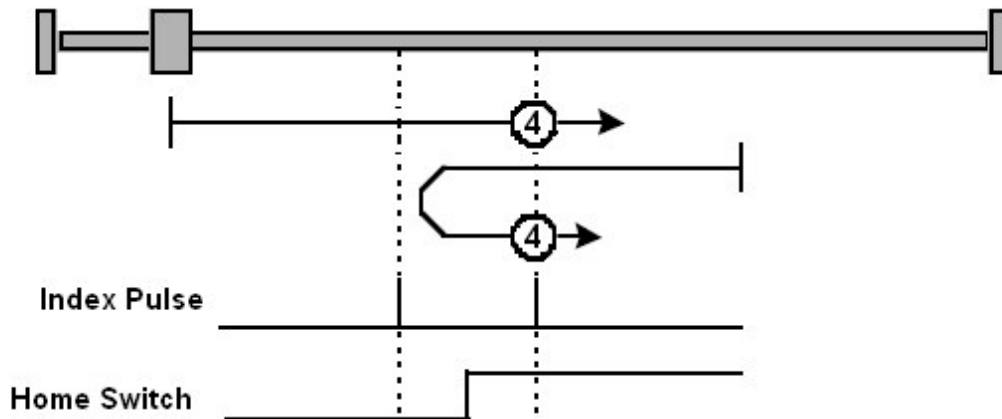
4.8.3.5 Method 4 - Homing on the positive home mark and index pulse

The home sensor input state determines the search direction of the sensor.

If the home input is high the motor is commanded to rotate anticlockwise. When home sensor commutation is found the motor is stopped and subsequently positioned at the

resolver home index pulse with a clockwise movement.

If the home input is low the motor is commanded to rotate clockwise. When home sensor commutation is found the motor is stopped and subsequently positioned at the resolver home index pulse with a clockwise movement.



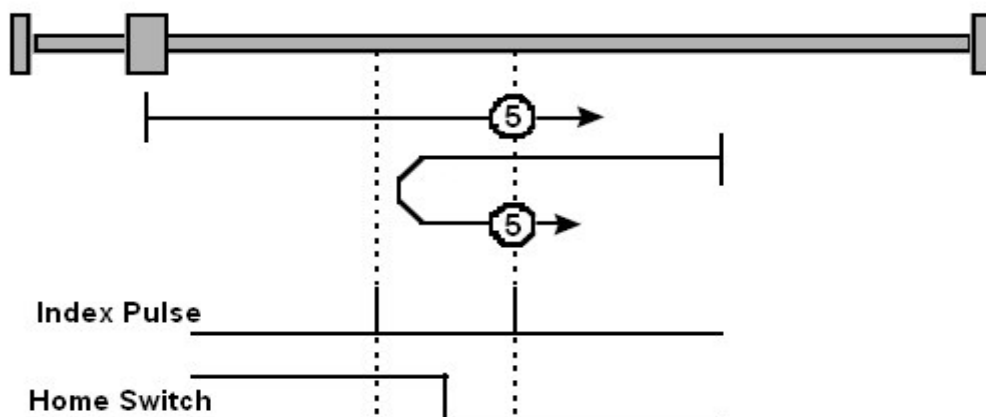
Drawing 10: Method 4 - Homing on home sensor

4.8.3.6 Method 5 - Homing on the negative home mark and index pulse

The home sensor input state determines the search direction of the sensor.

If the home input is low the motor is commanded to rotate anticlockwise. When home sensor commutation is found the motor is stopped and subsequently positioned at the resolver home index pulse with a clockwise movement.

If the home input is high the motor is commanded to rotate clockwise. When home sensor commutation is found the motor is stopped and subsequently positioned at the resolver home index pulse with a clockwise movement.



Drawing 11: Method 5 - Homing on home sensor

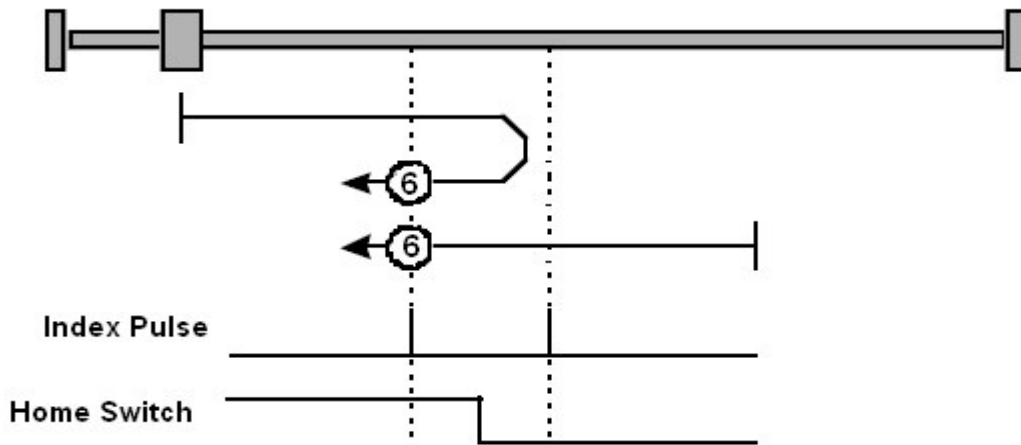
4.8.3.7 Method 6 - Homing on the negative home mark and index pulse

The home sensor input state determines the search direction of the sensor.

If the home input is high the motor is commanded to rotate clockwise. When home sensor

commutation is found the motor is stopped and subsequently positioned at the resolver home index pulse with an anticlockwise movement.

If the home input is low the motor is commanded to rotate anticlockwise. When home sensor commutation is found the motor is stopped and subsequently positioned at the resolver home index pulse with an anticlockwise movement.



Drawing 12: Method 6 - Homing on home sensor

4.8.3.8 Method 7 - Homing on the home mark and index pulse

The search direction of the home sensor is clockwise. Once the home sensor has been found the drive moves the motor at low speed and in an anticlockwise direction to exit from the sensor. After this it moves anticlockwise on the resolver home index pulse.

If the CW limit switch is touched, the sense of rotation is inverted in order to take the motor to the home sensor.

4.8.3.9 Method 8 - Homing on the home mark and index pulse

The search direction of the home sensor is clockwise. Once the home sensor has been found the drive moves the motor at low speed and in an anticlockwise direction to exit from the sensor. After this it moves clockwise on the resolver index pulse.

If the CW limit switch is touched, the sense of rotation is inverted in order to take the motor to the home sensor.

4.8.3.10 Method 9 - Homing on the home mark and index pulse

The search direction of the home sensor is clockwise. Once the home sensor has been found the drive moves the motor at low speed and in a clockwise direction to exit from the sensor. After this it moves anticlockwise on the resolver index pulse.

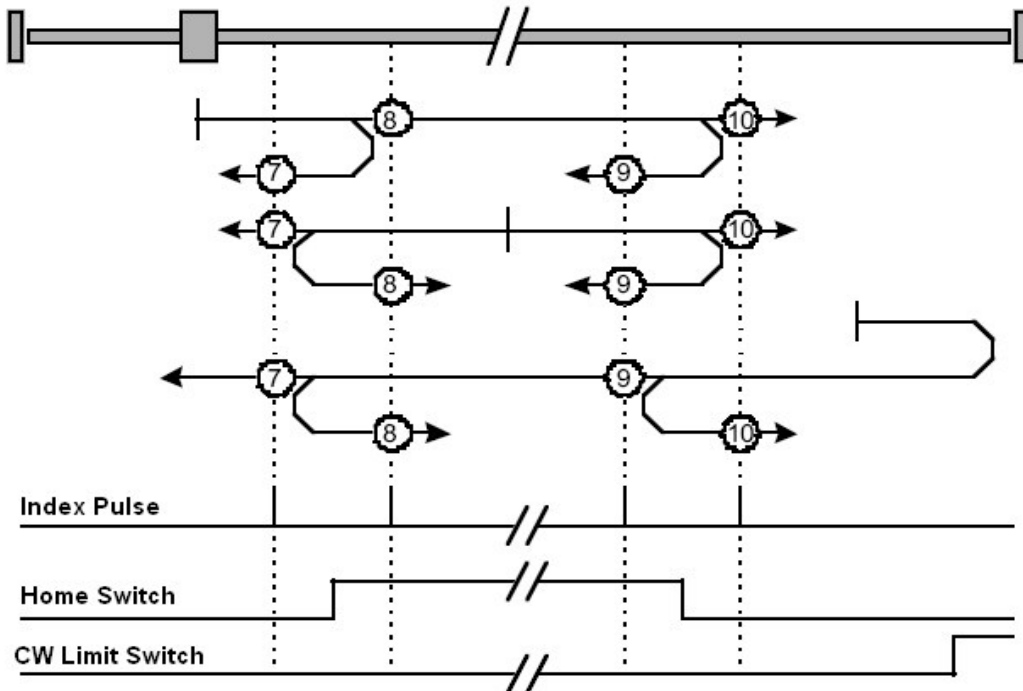
If the CW limit switch is touched, the sense of rotation is inverted in order to take the motor to the home sensor.

4.8.3.11 Method 10 - Homing on the home mark and index pulse

The search direction of the home sensor is clockwise. Once the home sensor has been found the drive moves the motor at low speed and in a clockwise direction to exit from the sensor. After this it moves clockwise on the resolver index pulse.

If the CW limit switch is touched, the sense of rotation is inverted in order to take the motor

to the home sensor.



Drawing 13: Method 7,8,9,10 - Homing on home sensor

4.8.3.12 Method 11 - Homing on the home mark and index pulse

The search direction of the home sensor is anticlockwise. Once the home sensor has been found the drive moves the motor at low speed and in a clockwise direction to exit from the sensor. After this it moves clockwise on the resolver index pulse.

If the CCW limit switch is touched, the sense of rotation is inverted in order to take the motor to the home sensor.

4.8.3.13 Method 12 - Homing on the home mark and index pulse

The search direction of the home sensor is anticlockwise. Once the home sensor has been found the drive moves the motor at low speed and in a clockwise direction to exit from the sensor. After this it moves anticlockwise on the resolver index pulse.

If the CCW limit switch is touched, the sense of rotation is inverted in order to take the motor to the home sensor.

4.8.3.14 Method 13 - Homing on the home mark and index pulse

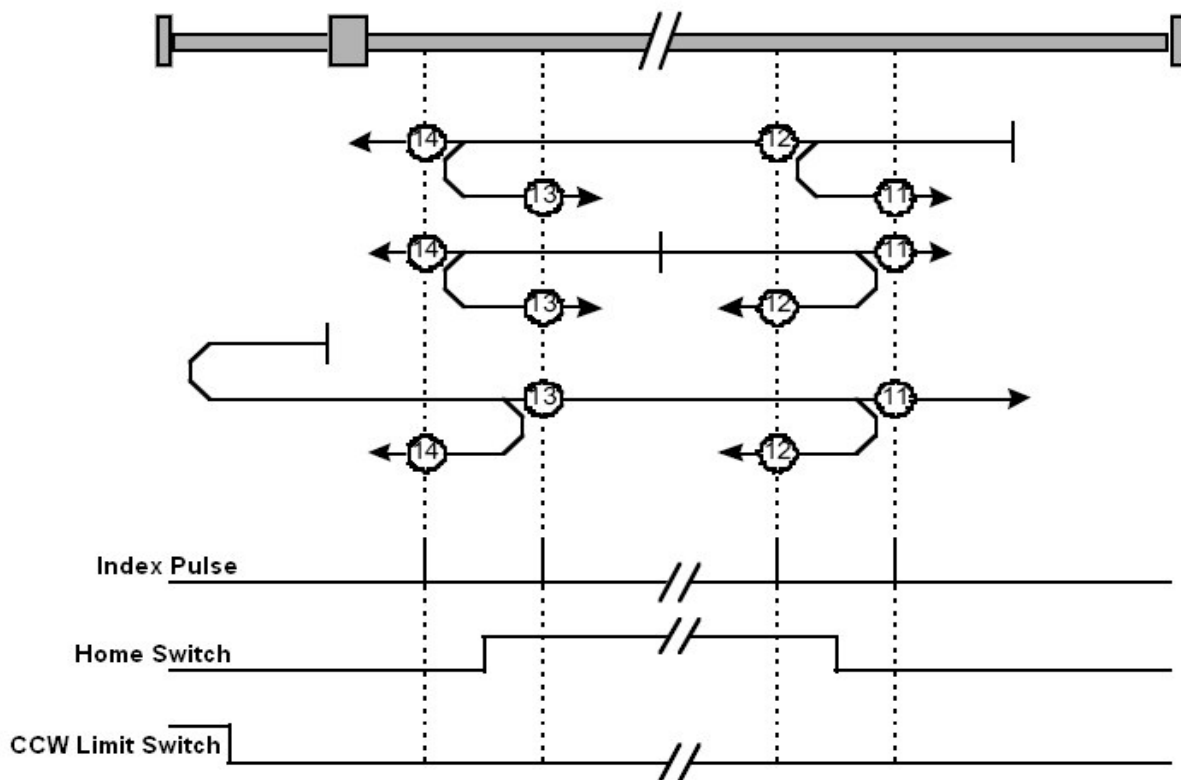
The search direction of the home sensor is anticlockwise. Once the home sensor has been found the drive moves the motor at low speed and in an anticlockwise direction to exit from the sensor. After this it moves clockwise on the resolver index pulse.

If the CCW limit switch is touched, the sense of rotation is inverted in order to take the motor to the home sensor.

4.8.3.15 Method 14 - Homing on the home mark and index pulse

The search direction of the home sensor is anticlockwise. Once the home sensor has been found the drive moves the motor at low speed and in an anticlockwise direction to exit from the sensor. After this it moves anticlockwise on the resolver index pulse.

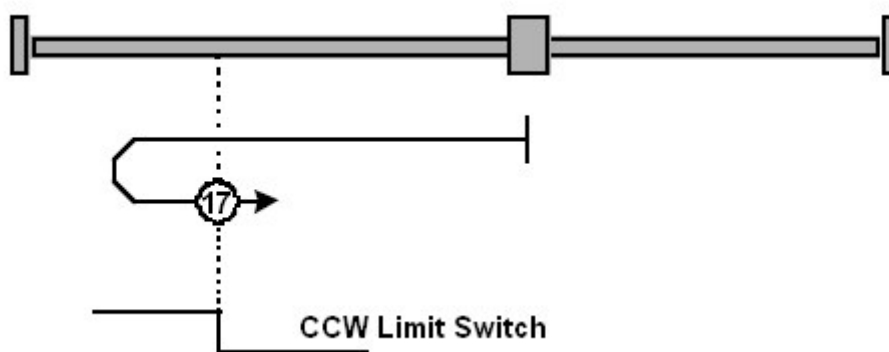
If the CCW limit switch is touched, the sense of rotation is inverted in order to take the motor to the home sensor.



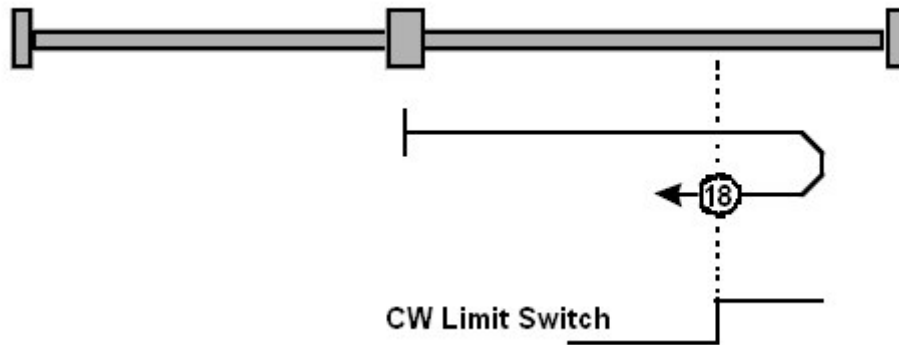
Drawing 14: Method 11,12, 13, 14 - Homing on home sensor

4.8.3.16 Methods from 17 to 30

The homing methods that go from 17 to 30 correspond respectively to the methods that go from 1 to 14, the only difference being that in this case the sensor home index pulse is not searched for. As an example, methods 17 and 18 are carried out as shown in the drawings that follow.



Drawing 15: Method 17 - Homing on CCW limit switch



Drawing 16: Method 18 - Homing on CW limit switch

4.8.3.17 Method 33 - Homing on index pulse (negative direction)

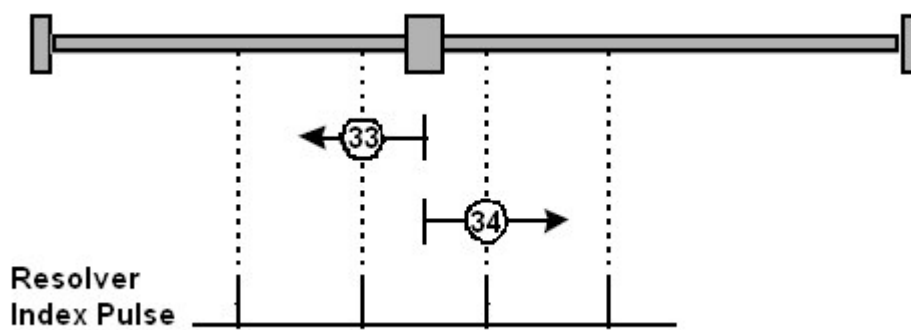
Starting from the point in which it is, the drive moves negatively on the resolver home index pulse. The point reached in this way becomes the drive home.

4.8.3.18 Method 34 - Homing on index pulse (positive direction)

Starting from the point in which it is, the drive moves positively on the resolver home index pulse. The point reached in this way becomes the drive home.

4.8.3.19 Method 35 - Homing on the current position

The current position is taken as the drive home.



Drawing 17: Method 3,4 - Homing on resolver zero impulse

Object 6099h - Homing speeds

Index	Sub-Index	Name	Type	Range	Default	Unit	Map	Attributes
6099h		Homing speeds	ARRAY					
	0	<i>number of entries</i>	UINT8	0..2	2		No	ro
	1	<i>Speed during search for switch</i>	UINT32	1..9999	100	rpm	Yes	rw
	2	<i>Speed during search for zero</i>	UINT32	1..999	10	rpm	Yes	rw

The *Homing speeds* vector has a *Speed during search for switch* parameter that represents the search speed of the limit switch, while *Speed during search for home* is the value of the movement speed on the resolver home index pulse.

The value of these two pieces of information is converted into drive internal units by the *velocity encoder factor* (index 6094h) on the basis of these formulas:

$$\text{internal limit switch search speed internal} = \frac{(\text{Numerator} \times \text{Speed during search for switch})}{\text{Divisor}}$$

$$\text{internal index pulse search speed} = \frac{(\text{Numerator} \times \text{Speed during search for zero})}{\text{Divisor}}$$

4.8.4 Object 609Ah – Homing acceleration

Index	Name	Type	Range	Default	Unit	Map	Attributes
609Ah	Homing acceleration	UINT32	1..6000000	1000	rpm /s	No	rw

Parameter for setting the movement acceleration and deceleration value during the homing phase. The datum of this variable (expressed in [rpm/s]) is converted into drive internal units by the *Acceleration factor* (index 6097h), on the basis of this formula:

$$\text{internal acceleration reference} = \frac{(\text{Numerator} \times \text{Homing acceleration})}{\text{Divisor}}$$

4.9 Specific limit switch settings

When CanOpen commutation is set in the DGM drive using the **S.7000** parameter, the sensor activation mode can be set in parameters **I.1001**, **I.2001** and **I.6001** relative to the inputs prepared for the limit switches:

DGM parameter	Name	Description
I.1001	CW limit switch contact type	<p>If the value is set at 24, the input prepared for the CW limit switch sensor is considered active when there is a 24 V voltage at input.</p> <p>If set at 0 the input is considered as being active at 0 V.</p>
I.2001	CCW limit switch contact type	<p>If the value is set at 24, the input prepared for the CCW limit switch sensor is considered active when there is a 24 V voltage at input.</p> <p>If set at 0 the input is considered as being active at 0 V.</p>
I.6001	Home limit switch contact type	<p>If the value is set at 24, the input prepared for the Home limit switch sensor is considered active when there is a 24 V voltage at input.</p> <p>If set at 0 the input is considered as being active at 0 V.</p>

4.10 Interpolated Position Mode

This control method makes it possible to carry out movements in space on the basis of data sent by the master. The variables for linear interpolation are made up of position and time, while additional data is sent for more sophisticated profiles.

Index	Name	Type	Map	Attributes
60C0h	<i>Interpolation sub mode select</i>	INT16	No	rw
60C1h	<i>Interpolation data record</i>	(ARRAY) INT32	Yes	rw
60C2h	<i>Interpolation time period</i>	RECORD	No	rw

Table 31: Interpolator parameters

4.10.1 Controlword in Interpolated Position Mode bit setting

Bit	Description	
0	<i>Switch on</i>	
1	<i>Enable voltage</i>	
2	<i>Quick stop</i>	
3	<i>Enable operation</i>	
4	Enable ip mode	
	Value	Description
	0	Disable <i>Interpolated position mode</i>
	1	Enable <i>Interpolated position mode</i>
5..6	<i>Reserved</i>	
7	<i>Fault reset</i>	
8	Halt	
	Value	Description
	0	Carry out the instructions of bit 4
	1	Stop in speed ramp
10..9	<i>Reserved</i>	
15..11	<i>Manufacturer specific</i>	

4.10.2 Statusword in *Interpolated Position Mode* bit value

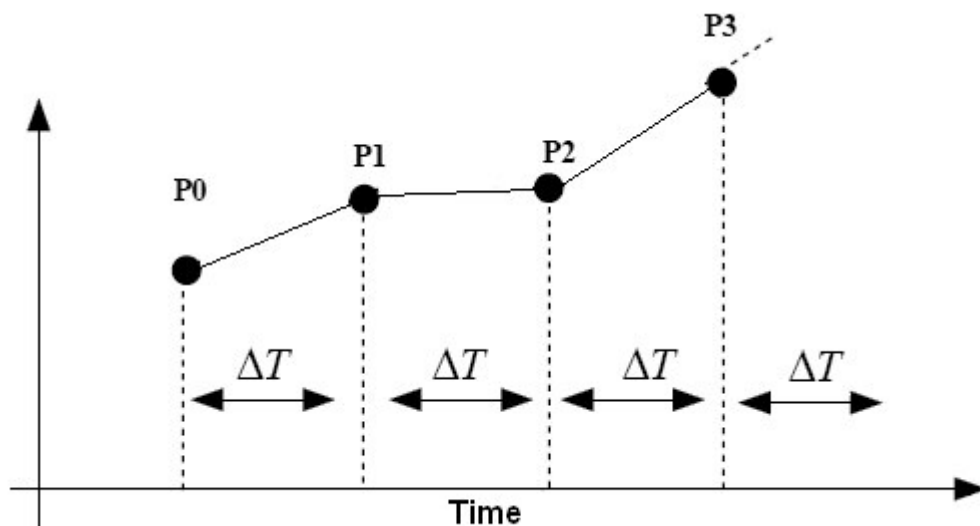
Bit	Description		
0	Ready to switch on		
1	Switched on		
2	Operation enable		
3	Fault		
4	Voltage enabled		
5	Quick stop		
6	Switch on disabled		
7	Warning		
8	Manufacturer specific		
9	Remote		
10	Position reached (Target reached)		
	Value	controlword Halt Bit	Description
	0	0	Position not yet reached
		1	Axis deceleration phase
	1	0	Position reached
		1	Axis speed at zero
11	Internal limit active (valid when Enable Position Limit parameter (index 201Ah) is set to 1)		
	Value	Description	
	0	Absolute position reference is not limited by software position limit parameter (index 607Dh)	
	1	Absolute position reference is limited by software position limit parameter (index 607Dh)	
12	ip mode active		
	Value	Description	
	0	Interpolated position mode not active	
	1	Interpolated position mode active	
13	Reserved		
14..15	Manufacturer specific		

4.10.3 Object 60C0h - *Interpolation sub mode select*

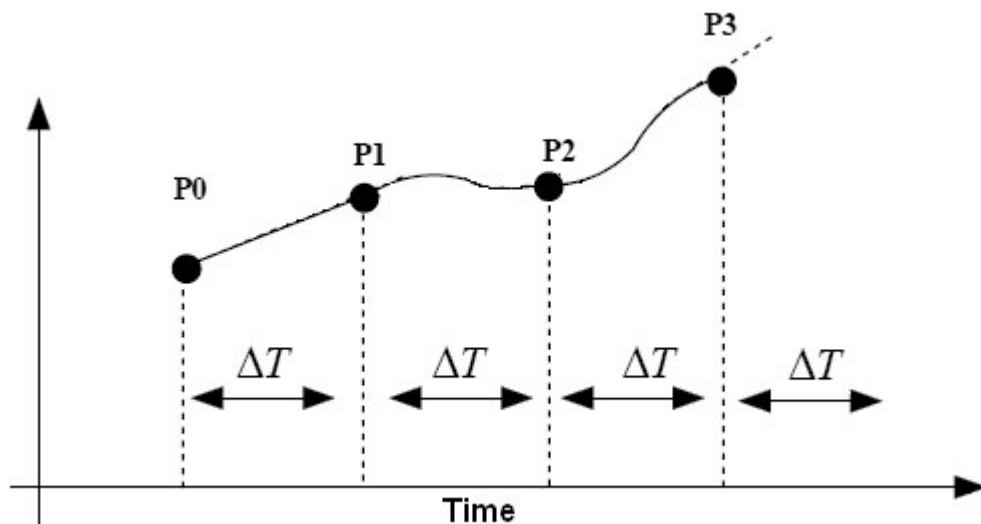
Index	Name	Type	Range		Default	Unit	Map	Attributes
60C0h	Motion profile type	INT16	0	<i>Linear interpolation</i>	0		No	rw
			-1	<i>Cubic Spline (PV)</i>				

This parameter makes it possible to select which type of profile to use for carrying out the movements of a quota.

Values	Motion profile type	Description
0	<i>Linear interpolation</i>	The drive follows linear interpolation from point to point during the time period set in the <i>Interpolation time period</i> (index 60C2h) parameter. For this type of interpolation the master only has to send the position data.
-1	<i>Cubic Spline (PV)</i>	In this case interpolation from one point to another is cubic because the function for calculating the profile progress results as being third level. The interpolation time is given by the <i>Interpolation time period</i> (index 60C2h) For this type of interpolation the master has to send the interpolation point position and speed data.



Drawing 18: Linear interpolation



Drawing 19: Cubic interpolation

4.10.4 Object 60C1h - *Interpolation data record*

Index	Sub-Index	Name	Type	Range	Default	Unit	Map	Attributes
60C1h		<i>Interpolation data record</i>	ARRAY					
	0	<i>number of entries</i>	UINT8	0..2	2		No	ro
	1	<i>first parameter of ip function</i>	INT32				Yes	rw
	2	<i>second parameter of ip function</i>	INT32				Yes	rw

This parameter saves the interpolation data coming from the master. On the basis of the value of the ***Motion profile type*** (index 60C0h) parameter, sub-indexes 1 and 2 can take on the following meaning:

<i>Motion profile type</i>	Name Sub-Index <i>Interpolation data record</i>	Meaning	Type
0 = <i>Linear interpolation</i>	<i>first parameter of ip function</i>	Position datum	INT32
	<i>second parameter of ip function</i>	Not used	INT32

<i>Motion profile type</i>	Name Sub-Index <i>Interpolation data record</i>	Meaning	Type
-1 = <i>Cubic Spline (PV)</i>	<i>first parameter of ip function</i>	Position datum	INT32
	<i>second parameter of ip function</i>	Speed datum	INT32

The position datum is transformed into internal position units for the drive using the *position factor* (index 6093h) and the following formula:

$$\text{internal position reference} = \frac{(\text{Numerator} \times \text{Position IP data})}{(\text{Feed constant})}$$

The speed value is transformed into internal position units for the drive using the *velocity encoder factor* (index 6094h) and the following formula:

$$\text{internal speed reference} = \frac{(\text{Numerator} \times \text{Velocity IP data})}{\text{Divisor}}$$

4.10.5 Object 60C2h - *Interpolation time period*

Index	Sub-Index	Name	Type	Range	Default	Unit	Map	Attributes
60C2h		<i>Interpolation time period</i>	RECORD					
	0	<i>number of entries</i>	UINT8	0..2	2		No	ro
	1	<i>Interpolation time units</i>	UINT8	1..255	2	ms	No	rw
	2	<i>Interpolation time index</i>	INT8	-3	-3		No	ro

The *Interpolation time period* defines the time interval that passes between two position values: sub-index 1 defines the numeric value and sub-index 2 defines the time measurement unit. In the DGM drive the sub-index is read only and equals -3; this numeric value indicates that the *interpolation time units* are expressed in ms.

$$\text{Time interval between two points} = \text{interpolation time units} [ms]$$

4.11 Profile Velocity Mode

Setting the *profile velocity mode* activates the speed loop. The speed reference is conditioned by the acceleration and deceleration ramps and is managed with the *Profile acceleration* (index 6083h) and *Profile deceleration* (index 6084h) parameters relative to the profile position mode.

The parameters of the *profile velocity mode* are given in the following list:

Index	Name	Type	Map	Attributes
6069h	<i>Velocity sensor actual value</i>	INT32	Yes	ro
606Bh	<i>Velocity demand value</i>	INT32	Yes	ro
606Ch	<i>Velocity actual value</i>	INT32	Yes	ro
606Dh	<i>Velocity window</i>	UINT16	No	rw
606Eh	<i>Velocity window time</i>	UINT16	No	rw
606Fh	<i>Velocity threshold</i>	UINT16	No	rw
6070h	<i>Velocity threshold time</i>	UINT16	No	rw
60FFh	<i>Target velocity</i>	INT32	Yes	rw

Table 32: Profile velocity mode parameters

4.11.1 Controlword in Profile Velocity Mode (bits setting)

Bit	Description	
0	<i>Switch on</i>	
1	<i>Enable voltage</i>	
2	<i>Quick stop</i>	
3	<i>Enable operation</i>	
4..6	<i>Reserved</i>	
7	<i>Fault reset</i>	
8	Halt	
	Value	Description
	0	Carries out axis movement on the basis of the speed reference
	1	Stop in speed ramp
10..9	<i>Reserved</i>	
15..11	<i>Manufacturer specific</i>	

4.11.2 Statusword in Profile Velocity Mode (bits value)

Bit	Description		
0	Ready to switch on		
1	Switched on		
2	Operation enable		
3	Fault		
4	Voltage enabled		
5	Quick stop		
6	Switch on disabled		
7	Warning		
8	Manufacturer specific		
9	Remote		
10	Reference reached (Target reached)		
	Value	controlword Halt bit	Description
	0	0	Reference not yet reached
		1	Axis deceleration phase
	1	0	Reference reached
		1	Axis speed at zero
11	Internal limit active		
12	Speed (Speed)		
	Value	Description	
	0	Axis speed not zero	
	1	Axis speed equal to zero	
13	(Max Slippage Error) – Not managed		
14..15	Manufacturer specific		

4.11.3 Object 6069h - *Velocity sensor actual value*

Index	Name	Type	Range	Default	Unit	Map	Attributes
6069h	<i>Velocity sensor actual value</i>	INT32				Yes	ro

The *Velocity sensor actual value* supplies the speed value calculated by the position sender given in resolver increases per second [increases/second].

4.11.4 Object 606Bh - *Velocity demand value*

Index	Name	Type	Range	Default	Unit	Map	Attributes
606Bh	<i>Velocity demand value</i>	INT32			rpm	Yes	ro

The *Velocity demand value* corresponds to the output speed of the function prepared for generating the ramp speed profile. The unit of measurement of this parameter is expressed in [rpm].

4.11.5 Object 606Ch - *Velocity actual value*

Index	Name	Type	Range	Default	Unit	Map	Attributes
606Ch	<i>Velocity actual value</i>	INT32			rpm	Yes	ro

The *velocity actual value* is the measured speed value expressed in [rpm]. It represents the speed feedback value that is compared with the *Velocity demand value* (index 606Bh) inside the speed loop.

Inside the drive the speed measured is expressed in pulses of the resolver on seconds (pulses/s) and it's reported on the same unit of measure of the *Target Velocity* (index 60FFh) multiplying for the “*divisor*” and dividing for the “*numerator*” of the *velocity encoder factor* (index 6094h) following this formula:

$$\text{Velocity actual value} = \frac{(\text{Internal Speed of drive} \times \text{Divisor})}{\text{Numerator}}$$

When maintaining the default values of *velocity encoder factor* that are *Numerator* = 65536 and *Divisor* = 60, then the measured speed is expressed in rpm

4.11.6 Object 606Dh - *Velocity window*

Index	Name	Type	Range	Default	Unit	Map	Attributes
606Dh	<i>Velocity window</i>	UINT16	9999	20	rpm	No	rw

The *velocity window* parameter indicates a speed interval used to verify if the measured speed is kept within a specific range as regards the datum reference:

$$[\text{target velocity} - \text{velocity window} .. \text{target velocity} + \text{velocity window}]$$

4.11.7 Object 606Eh - *Velocity window time*

Index	Name	Type	Range	Default	Unit	Map	Attributes
606Eh	<i>Velocity window time</i>	UINT16	20000	500	ms	No	rw

The *Velocity window time* parameter is used to verify if the measured speed is within the interval specified in *Velocity window* (index 606Eh) during period of time considered in *Velocity window time*, if so bit 10 of the *status word* is put to 1.

4.11.8 Object 606Fh - *Velocity threshold*

Index	Name	Type	Range	Default	Unit	Map	Attributes
606Fh	<i>Velocity threshold</i>	UINT16	9999	20	rpm	No	rw

The *Velocity threshold* indicates a speed threshold as to the zero, if the absolute value of the measured speed exceeds this speed threshold for the time set in the *Velocity threshold time* (index 6070h), bit 12 of the *status word* is placed at 0. The opposite happens in the complementary situation, where bit 12 is placed at 1 and indicates that the shaft is stationary.

4.11.9 Object 6070h - *Velocity threshold time*

Index	Name	Type	Range	Default	Unit	Map	Attributes
6070h	<i>Velocity threshold time</i>	UINT16	20000	500	ms	No	rw

The *velocity threshold time* parameter indicates the time during which it is verified if the measured speed exceeds the value set in *Velocity threshold* (index 606Fh).

4.11.10 Object 60FFh – *Target Velocity*

Index	Name	Type	Range	Default	Unit	Map	Attributes
60FFh	<i>Target Velocity</i>	INT32	±9999	0	rpm	Yes	rw

Target velocity is a speed reference expressed in [rpm] at the input of the function that generates the acceleration and deceleration ramps.

This parameter is multiplied for the “*Numerator*” and divided for the “*Divisor*” of the *velocity encoder factor* (index 6094h), if the *Numerator*=65536 and *Divisor*=60, then the *Target Velocity* is expressed in rpm

4.12 Velocity Mode

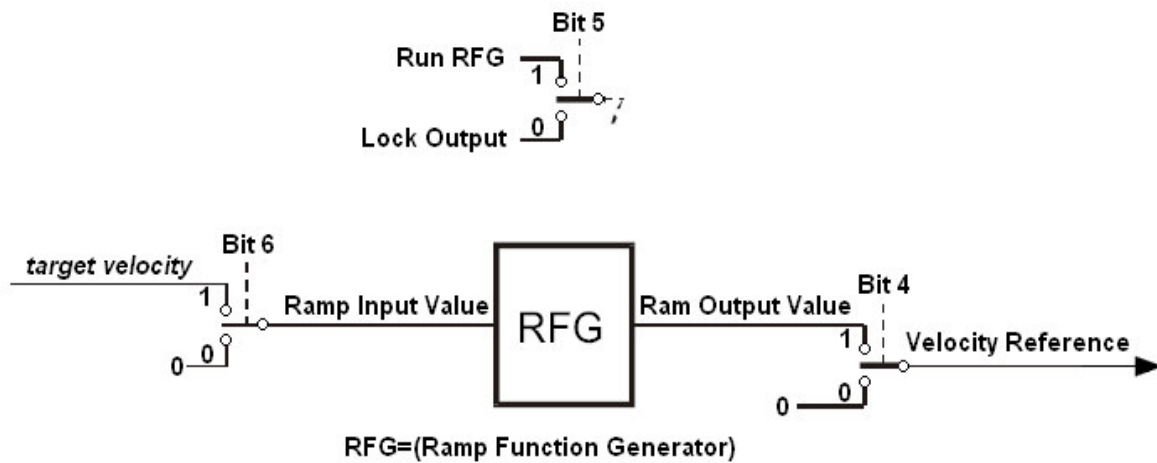
Setting the velocity mode activates the speed loop; in this mode a set of variables is made available for modifying and limiting the speed reference. It is also possible to condition the speed reference with acceleration and deceleration ramps.

<i>Index</i>	<i>Name</i>	<i>Type</i>	<i>Map</i>	<i>Attributes</i>
6042h	<i>vl target velocity</i>	<i>INT16</i>	<i>Yes</i>	<i>rw</i>
6043h	<i>vl velocity demand</i>	<i>INT16</i>	<i>Yes</i>	<i>ro</i>
6044h	<i>vl control effort</i>	<i>INT16</i>	<i>Yes</i>	<i>rw</i>
6046h	<i>vl velocity min max amount</i>	<i>(ARRAY) UINT32</i>	<i>No</i>	<i>rw</i>
6048h	<i>vl velocity acceleration</i>	<i>RECORD</i>	<i>Yes</i>	<i>rw</i>
6049h	<i>vl velocity deceleration</i>	<i>RECORD</i>	<i>Yes</i>	<i>rw</i>

Table 33: Velocity mode parameters

4.12.1 Controlword in Velocity Mode (bits setting)

Bit	Description	
0	Switch on	
1	Enable voltage	
2	Quick stop	
3	Enable operation	
4	rfg enable	
	Value	Description
	0	The drive remains stopped in torque with null speed
	1	Speed loop reference enabled and corresponding to the output variable of the acceleration and deceleration ramps.
5	rfg unlock	
	Value	Description
	0	Blocks the ramp output variable at the current value.
	1	Output variable of the enabled ramps, updated by the function set for generating the acceleration or deceleration ramps on the basis of the vl velocity acceleration (index 6048h) e vl velocity deceleration (index 6048h) parameters.
6	rfg use ref	
	Value	Description
	0	Places the input of the function that generates the ramps at zero
	1	The vl target velocity (index 6042h) value is assigned as an input to the function that generates the speed ramps
7	Fault reset	
8	Halt	
	Value	Description
	0	Carries out axis movement on the basis of speed reference
	1	Stop in ramp
15..9	Manufacturer specific	



Drawing 20: Bit 4,5,6 controlword velocity mode

4.12.2 Statusword in profile position mode (bits value)

Bit	Description
0	Ready to switch on
1	Switched on
2	Operation enable
3	Fault
4	Voltage enabled
5	Quick stop
6	Switch on disabled
7	Warning
8	Manufacturer specific
9	Remote
10	reserved
11	Internal limit active
12..13	reserved
14..15	Manufacturer specific

4.12.3 Object 6042h - vl target velocity

Index	Name	Type	Range	Default	Unit	Map	Attributes
6042h	vl target velocity	INT16	±7FFF	100		Yes	rw

This parameter is multiplied for the “*Numerator*” and divided for the “*Denominator*” of the *vl dimension factor* (index 604Ch) and it represents the speed value at input of the function that generates the acceleration and deceleration ramps. The unit of measurement for this datum is in rpm.

4.12.4 Object 6043h - vl velocity demand

Index	Name	Type	Range	Default	Unit	Map	Attributes
6043h	vl velocity demand	INT16	±7FFF			Yes	ro

The *vl velocity demand* indicates the speed reference conditioned by the function that generates the ramps. This datum corresponds to the input of the speed control loop and is expressed with the same unit of measures of the *vl target velocity* (index 6042h).

4.12.5 Object 6044h - vl control effort

Index	Name	Type	Range	Default	Unit	Map	Attributes
6044h	vl control effort	INT16			rpm	Yes	ro

The *vl control effort* parameter corresponds to the speed measured at the motor axis expressed in the same unit of measure of ***vl target velocity*** (index6042h)

4.12.6 Object 6046h - vl velocity min max amount

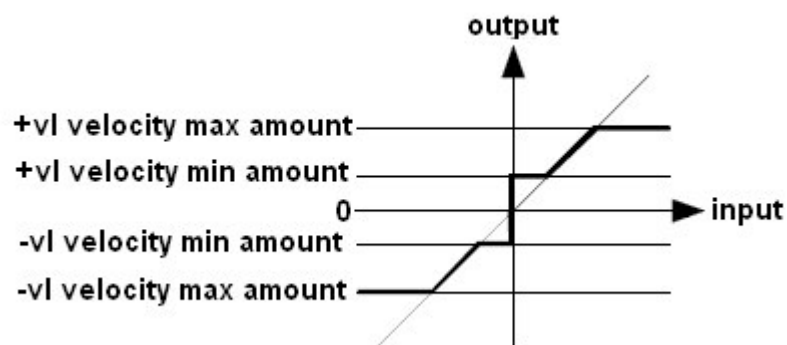
Index	Sub-Index	Name	Type	Range	Default	Unit	Map	Attributes
6046h		vl velocity min max amount	ARRAY					
	0	number of entries	UINT8	0..2	2		No	ro
	1	vl velocity min amount	UINT32	1..9999	0	rpm	No	rw
	2	vl velocity max amount	UINT32	1..9999	5000	rpm	No	rw

The *vl velocity min max amount* array contains two values that limit in module, within a specific interval, the speed reference corresponding to parameter ***vl target velocity*** (index 6042h).

The parameter indicates the minimum value in module that the speed reference can assume. This means that if the speed reference is positive, its minimum value is *+vl velocity min amount*, while for negative speed values the maximum value becomes *-vl velocity min amount*.

The *vl velocity max amount* parameter indicates the maximum value in module that the speed reference can assume. This means that with a positive speed the maximum value is *+vl velocity max amount*, while for negative speed values the minimum value becomes *-vl velocity max amount*.

The parameter *vl velocity max amount* must be greater than the *vl velocity min amount*. If you try to set *vl velocity min amount* at a value greater than the *vl velocity max amount* then an error code is given back (in hexadecimal) 0090030.



Drawing 21: vl velocity min max amount characteristic

4.12.7 Object 6048h - vl velocity acceleration

Index	Sub-Index	Name	Type	Range	Default	Unit	Map	Attributes
6048h		vl velocity acceleration	RECORD					
	0	number of entries	UINT8	0..2	2		No	ro
	1	Delta speed	UINT32	1..FFFFFFFFh	1000		Yes	rw
	2	Delta time	UINT16	0..FFFFh	1	s	Yes	rw

The *vl velocity acceleration* record contains the variables for setting acceleration.

The *Delta speed* parameter indicates the variation of the speed in time and it is expressed in the same unit of measure of the *vl target velocity* (index 6042h), while Delta time gives the time value expressed in s (seconds) . Acceleration can be calculated using the formula below:

$$acceleration\ reference = \frac{(Delta\ speed)}{(Delta\ time)}$$

If the *Delta time* variable is null the reference is followed instantly and a positive speed variation is no longer conditioned by acceleration.

4.12.8 Object 6049h - vl velocity acceleration

<i>Index</i>	<i>Sub-Index</i>	<i>Name</i>	<i>Type</i>	<i>Range</i>	<i>Default</i>	<i>Unit</i>	<i>Map</i>	<i>Attributes</i>
6049h		vl velocity deceleration	RECORD					
	0	number of entries	UINT8	0..2	2		No	ro
	1	Delta speed	UINT32	1..FFFFFFFFh	1000		Yes	rw
	2	Delta time	UINT16	1..FFFFh	1	s	Yes	rw

Deceleration is defined by means of variables contained in the *vl velocity deceleration* record.

The Delta speed parameter indicates the speed variation expressed in the same unit of measure of the *vl target velocity* (index 6042h), while *Delta time* contains the time value in s (seconds). Deceleration can be calculated using the following formula:

$$deceleration\ reference = \frac{(Delta\ speed)}{(Delta\ time)}$$

If the Delta time variable is null the reference is followed instantly and a positive speed variation is no longer conditioned by acceleration.

4.13 Profile Torque Mode

Torque control is enabled writing value 4 on the parameters *Modes of operation* (index6060h), in this case it is active only the current loop control in quadrature that gives the torque to the motor. The torque reference is given in per thousand (‰) of the value *Motor rated current* (index 2001h). For example if in the parameter *Target Torque* (index 6071h) you write 1000, then the motor will receive a torque current corresponding to the rated current of the motor.

The *Target torque* can be managed directly or using linear torque ramps on the bases of the parameter *Torque profile type* (index 6088h), the bit 10 of the *status word* is set to 1 when the torque ramp is finished and the reference correspond to those set .

<i>Index</i>	<i>Name</i>	<i>Type</i>	<i>Map</i>	<i>Attributes</i>
6071h	<i>Target torque</i>	INT16	Yes	rw
6077h	<i>Torque actual value</i>	INT16	Yes	ro
6087h	<i>Torque slope</i>	UINR32	No	rw
6088h	<i>Torque profile type</i>	INT8	No	rw

Table 34: Parameters of Profile Torque Mode

4.13.1 Setting bit **Controlword** in **Profile Torque Mode**

Bit	Description
0	Switch on
1	Enable voltage
2	Quick stop
3	Enable operation
4..6	Reserved
7	Fault reset
8	Halt
	Value
	Description
	0 It moves the axis on the base of torque reference
	1 Coasting slow down
10..9	Reserved
15..11	Manufacturer specific

4.13.2 Valore bit **Statusword** in **Profile Torque Mode**

Bit	Description
0	Ready to switch on
1	Switched on
2	Operation enable
3	Fault
4	Voltage enabled
5	Quick stop
6	Switch on disabled
7	Warning
8	Manufacturer specific
9	Remote
10	Riferimento raggiunto (Target reached)
	Valore
	Description
	0 Riferimento non ancora raggiunto
	1 Riferimento raggiunto
11	Internal limit active
12..15	Manufacturer specific

4.13.3 Object 6071 - *Target torque*

Index	Name	type	Range	Default	Unit	Map	Attributes
6071h	Target torque	INT16	±2500	0		Si	rw

Il parametro *Target torque* rappresenta il riferimento di coppia ed è espresso in per mille rispetto al parametro *Corrente nominale motore* (index 2001h).

$$\text{Corrente di coppia} = \frac{(\text{Target Torque} \times \text{Corrente nominale motore})}{(1000)}$$

4.13.4 Object 6077 - *Torque actual value*

Index	Name	type	Range	Default	Unit	Map	Attributes
6077h	Target torque	INT16	±2500	0		Si	ro

This parameter give back the value of the torque measured in the same unit of measure of the *Target torque* (index 6071h).

4.13.5 Object 6078 - *Torque slope*

Index	Name	type	Range	Default	Unit	Map	Attributes
6078h	Torque slope	UINT32	1..25000	1000		No	rw

The parameter *Torque slope* set the variation of torque current per second expressed in for thousand (‰) compared with the rated current of the motor. The vlaue of this parameter represent the slope of linear ramps when the parameter *Torque profile type* (index 6088h) is set to 0.

4.13.6 Object 6088 - *Torque profile type*

Index	Name	type	Range		Default	Unit	Map	Attributes
6088h	Torque profile type	INT8	-1	<i>Immediate</i>	0		No	rw
			0	<i>Linear ramp</i>				

With the parameter *Torque profile type* is possible to choose the way to apply the torque current to the regulator controller that manage the current supplied to the motor.

Values	<i>Torque profile type</i>	Description
-1	<i>Immediate</i>	<ul style="list-style-type: none"> ● The reference <i>Target torque</i> (index 6071h) is immediately converted in a torque current expressed in “for thousand” (‰) compared to <i>Motor rated current</i> (index 2001h), the value obtained is immediately applied to the motor.
0	<i>Rampa lineare</i>	<ul style="list-style-type: none"> ● In this case the current reference given to the regulator that manage the current of the motor change following the linear ramps setting through the parameter <i>Torque slope</i> (index 6078h)

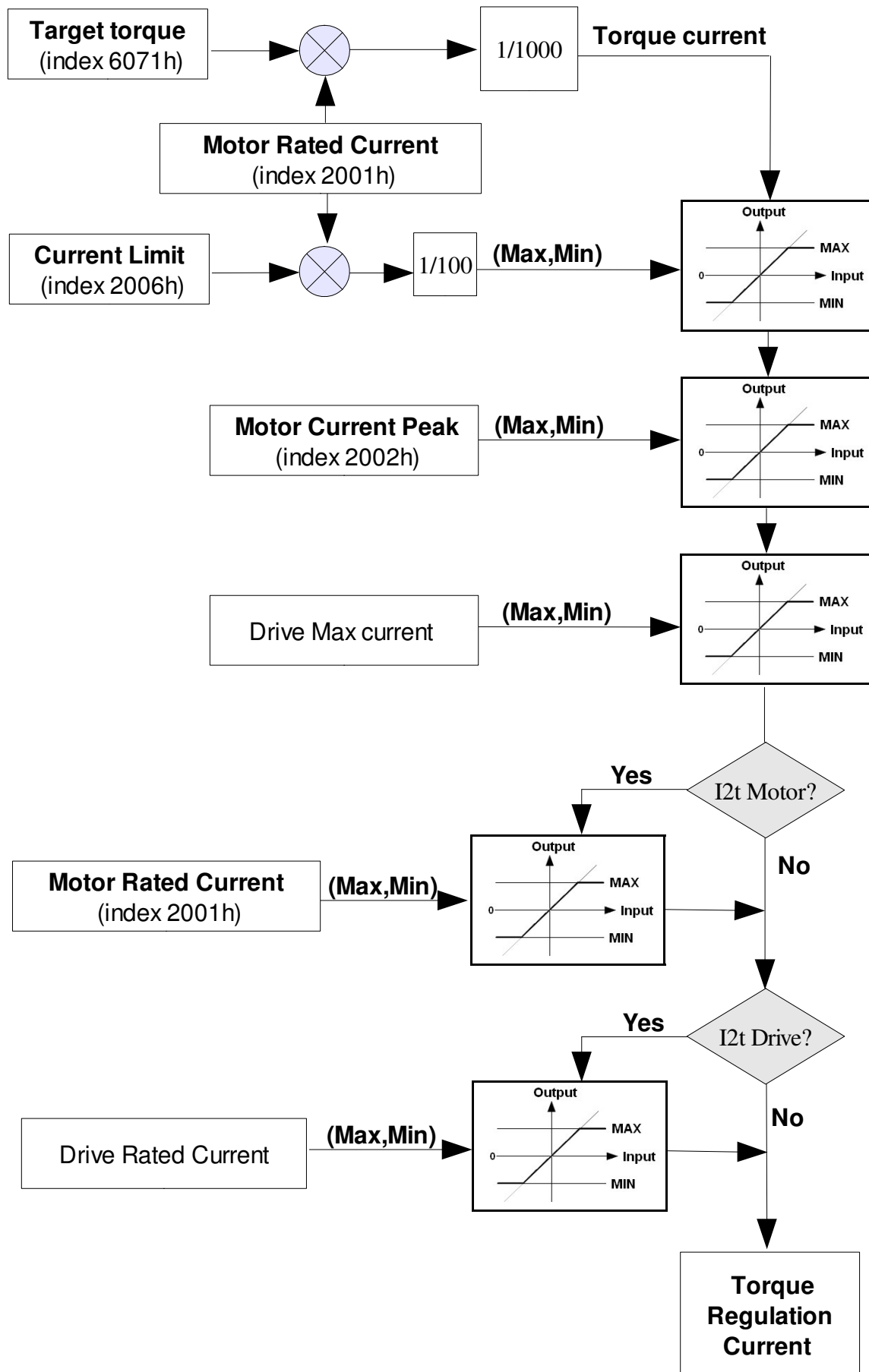
4.13.7 Torque current limitations

The value of the torque current obtained from the product of *Target torque* (index 6071h) for the *Motor Rated current* (index 2001h) and from the division for 1000, it is successively limited by other internal parameters of the drive.

The first limitation is set by the current data obtained from the product of parameter *Current Limit* (index 2006h) for the *Motor rated current* (index 2001h) and from the division of the product obtained for 100. The value of the current obtained correspond to the max limit, while the minimum limit can be obtained from the maximum with negative sign.

Successively the torque current is limited by the Motor Peak Current (index 2002h) and by the max current deliverable by the drive that is an internal parameter not changeable.

In case of interventio of the I2t motor alarm, the max torque current is limited by the *Rated motor current* (index 2001h), in analogic mode if the alarm of I2t of the drive intervein, the torque current is limited by the rated current value that the drive can supply.



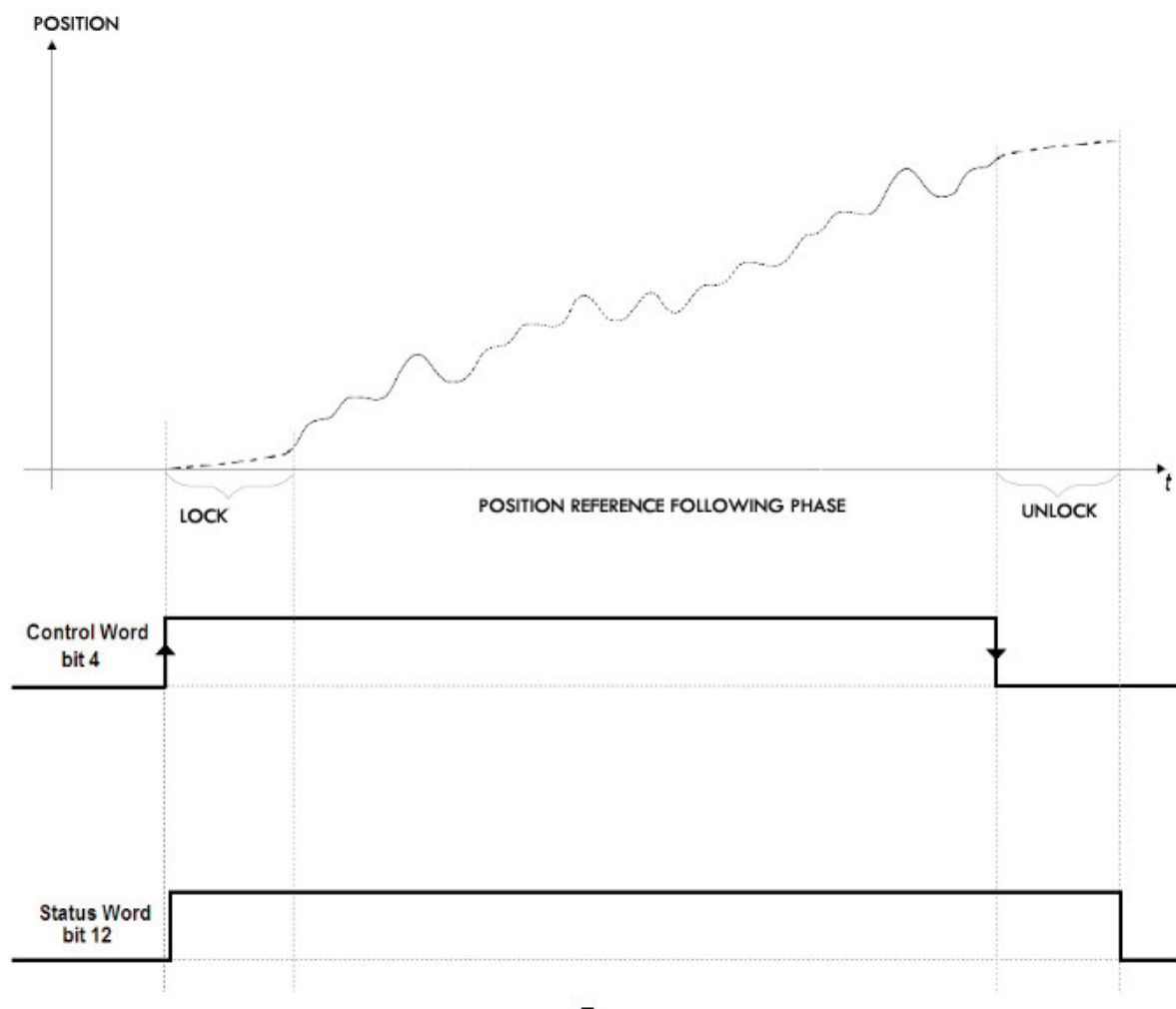
4.14 Electric Axis

With *Electric axis* mode the drive works in position control, the pulses of the external encoder are multiplied by *Numerator Ratio* (index 2200h) and divided for the parameter *Denominator Ratio* (index 2201h), the value obtained represent the reference for the position loop that works with a measured position of 65536 pulses/r.

To change to Axis Electric Mode, you need to set value at -1 in parameter *Modes of operation* (index 6060h), the activation of the electric axis function starts switching bit 4 of the *control word* from 0 to 1, This event activates the function of lock (hooking) that allows to the motor shaft to reach the same speed as the position reference calculated multiplying the pulses of the external encoder for the electric axis ratio. When the lock phase is completed, the drive changes to a phase in which it follows the position reference of the external encoder. In this phase the parameters of *Following error window* (index 6065h) and *Following error time out* (index 6066h) are active. These parameters set the limit for the position error control. To disable the electric axis is possible switching the bit 4 of the *control word* from 1 to 0, and this action activates the unlock function that are used to stop the shaft motor following the type of unlock function set. The bit 12 of status word give back the status of the activation of the electric axis, when the bit is 1 it means that the electric axis is active and the drive is making one of the following phases: lock phase, position following, unlock phase.

Index	Name	type	Map	Attributes
2200h	<i>Numerator Ratio</i>	INT16	Yes	rw
2201h	<i>Denominator Ratio</i>	UINT16	Yes	rw
2202h	<i>Lock mode</i>	UINT8	No	rw
2203h	<i>Unlock mode</i>	UINT8	No	rw
2204h	<i>Encoder mode</i>	UINT8	No	rw
2205h	<i>Pos. unlock absolute/relative</i>	UINT8	No	rw
2206h	<i>Lock in speed-acceleration</i>	UINT32	No	rw
2207h	<i>Lock in Speed-Jerk</i>	UINT32	No	rw
2208h	<i>Lock in Position-Quota</i>	UINT32	No	rw
2209h	<i>Unlock in Speed-Deceleration</i>	UINT32	No	rw
220Ah	<i>Unlock in Speed-Jerk</i>	UINT32	No	rw
220Bh	<i>Unlock in Position-Speed</i>	UINT16	No	rw
220Ch	<i>UnLock in speed-deceleration</i>	UINT16	No	rw
220Dh	<i>UnLock in Position-Quota</i>	INT32	No	rw
220Eh	<i>Encoder pulses</i>	UINT32	Yes	ro
220Fh	<i>Pulse Encoder master</i>	UINT32	No	rw

Tabella 35: Electric Axis Parameters



Drawing 22: Electric Axis Phases

4.14.1 Set bit *Controlword* in *Electric Axis*

Bit	Description	
0	<i>Switch on</i>	
1	<i>Enable voltage</i>	
2	<i>Quick stop</i>	
3	<i>Enable operation</i>	
4	<i>Electric Axis start</i>	
	Value	Description
	0 → 1	Enable procedure of Electric axis lock
	1 → 0	Disable procedure of Electric axis lock
5	<i>Reset encoder counter</i>	
	Value	Description
	0 → 1	When Electric Axis is disabled, the switching of this bit from 0 to 1, reset the Value of parameter <i>Encoder pulses Encoder</i> (index 220Eh).
6	<i>Reset position counter</i>	
	Value	Description
	0 → 1	When Electric Axis is disabled, the switching of this bit from 0 to 1, reset the Value of the position.
7	<i>Fault reset</i>	
8	<i>Halt</i>	
	Value	Description
	0	Execute instruction of bit 4
	1	Stop with ramp
10..9	<i>Reserved</i>	
15..11	<i>Manufacturer specific</i>	

4.14.2 Value bit *Statusword* in *Electric Axis*

Bit	Description	
0	<i>Ready to switch on</i>	
1	<i>Switched on</i>	
2	<i>Operation enable</i>	
3	<i>Fault</i>	
4	<i>Voltage enabled</i>	
5	<i>Quick stop</i>	
6	<i>Switch on disabled</i>	
7	<i>Warning</i>	
8	<i>Manufacturer specific</i>	
9	<i>Remote</i>	
10	Stato Halt	
	Bit Halt controlword	Description
	0	Command Halt inactive or in phase deceleration axis if it is activated the bit of halt and the axis is not still arrested
	1	Motor axis stop
11	<i>Internal limit active</i>	
12	Activation/deactivation of Electric Axis	
	Value	Description
	0	Electric Axis disabled, the drive is in stop
	1	Electric axis active, the drive can be in one of the following phases: Lock phase, position following, unlock phaseElectric.
13	Position Error (<i>Following error</i>)	
	Value	Description
	0	No error of position
	1	Position error found
14..15	<i>Manufacturer specific</i>	

4.14.3 Object 2200 – *Numerator Ratio*

Index	Name	type	Range	Default	Unit	Map	Attributes
2200h	<i>Numerator Ratio</i>	INT16	±7FFF	1		yes	rw

This Parameter represent the numerator of the ratio of the electric axis, together with the parameter *Denominator Ratio* (index 2201h), it allows to change the ratio between the drive position reference and the number of the pulses of the external encoder.

4.14.4 Object 2201 - *Denominator Ratio*

Index	Name	type	Range	Default	Unit	Map	Attributes
2201h	<i>Denominator Ratio</i>	UINT16	1..65535	1		Yes	rw

The parameter *Denominator Ratio* represent the denominator of the ratio of the electric axis.

4.14.5 Object 2202 – *Lock mode*

Index	Name	type	Range		Default	Unit	Map	Attributes
2202h	<i>Lock mode</i>	UINT8	0	<i>Immediate Lock</i>	1		No	rw
			1	<i>Lock in speed</i>				
			2	<i>Lock in position</i>				

The *Lock mode* allows to choose the mode to move from the stationary status with zero speed to the following reference status resulting from the calculation of the pulses of the external encoder multiplied for the electric axis ratio.

Value s	<i>Lock mode</i>	Description
0	<i>Immediate Lock</i>	<ul style="list-style-type: none"> ● In this mode the control of position is immediately activated to follow the reference given by an external encoder. ● This choice is suggested when at the activation of the electric axis, the rising edge of bit 4 of the <i>control word</i>, the pulses of the external encoder are null.

1	<i>Lock in speed</i>	<ul style="list-style-type: none"> ● In this mode the drive reach the speed of the position reference with a speed ramp starting from stop status ● The lock is executed on the base of parameters <i>Lock in speed-Acceleration</i> (index 2206h) and <i>Lock in speed-Jerk</i> (index 2207h).
2	<i>Lock in position</i>	<ul style="list-style-type: none"> ● In this way the drive, when the rising edge of the 4 bit of the <i>control word</i> activate the electric axis , reaches the same speed of the position reference covering the spase set in parameter <i>Lock in position-Quota</i> (index 2208h); The profile of the curve of lock is calculated following the polynomial curve of 5th degree.

4.14.6 Object 2203 – *Mode Unlock*

Index	Name	type	Range		Default	Unit	Map	Attributes
2203h	<i>Mode Unlock</i>	UINT8	0	<i>Immediate Unlock</i>	1		No	rw
			1	<i>Unlock in speed</i>				
			2	<i>Unlock in position</i>				

The *Unlock mode* set the mode of transition from the phase where the drive follows the position reference to the phase where the axis is in stop with a zero speed. This happens when the function electrica axis is disabled switching bit 4 of *control word* from 1 to 0.

Value s	<i>Modo Sgancio</i>	Description
0	<i>immediate unlock</i>	<ul style="list-style-type: none"> ● With this set the axis stops immediatly when Electric Axis is disabled. ● We suggest to use this function to disable the electric axis when encoder pulses are null.
1	<i>Unlock in speed</i>	<ul style="list-style-type: none"> ● In this mode the drive stops with a speed ramp. ● The unlock is executed following the parameters <i>Unlock in speed-Decelerazione</i> (index 2209h) and <i>Unlock in speed-Jerk</i> (index 220Ah).
2	<i>Unlock in position</i>	<ul style="list-style-type: none"> ● With this choice, the drive stops executing a position quota following a polynomial profile of 4th degree. ● The setting parameters for the unlock in position are the parameter <i>Position unlock absolute/relative</i> (index 2205h) that inidcates if the quota is relative or absolute, the parameter <i>Unlock in position-speed</i> (index 220Bh) that set the max speed of the quota, the parameter <i>Unlock in position-deceleration</i>

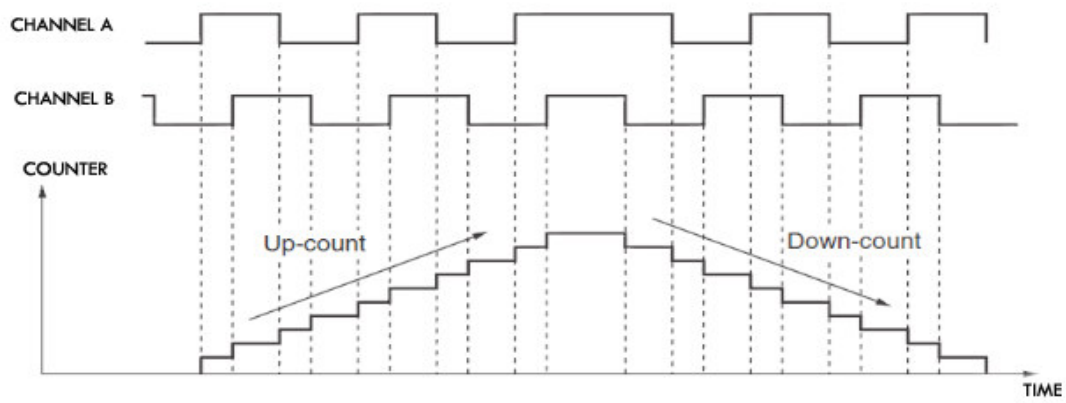
		(<i>index 220Ch</i>) tha set the max acceleration and the parameter and the parameter <i>unlock in position-quota</i> (index 220Dh) that is the space to cover
--	--	--

4.14.7 Object 2204 – *Encoder Mode*

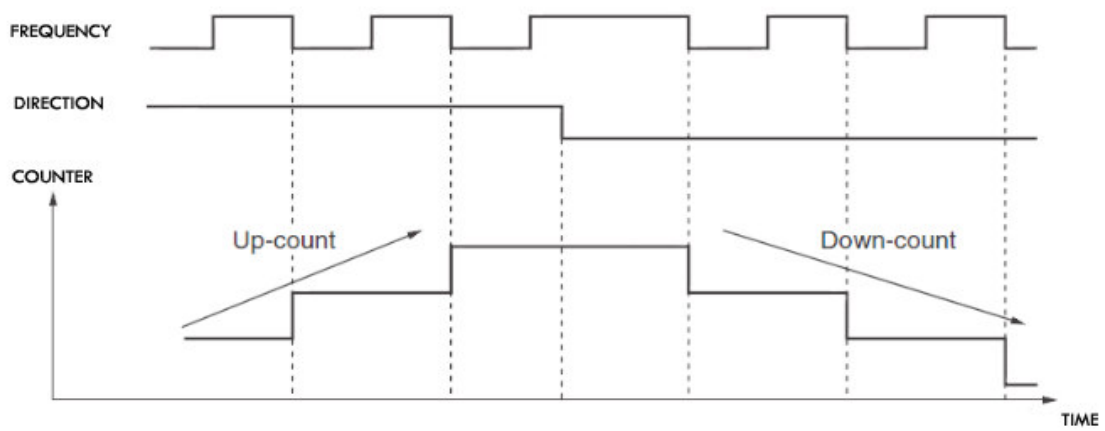
Index	Name	type	Range		Default	Unit	Map	Attributes
2204h	Encoder Mode	UINT8	0	<i>A-B Channel</i>	0		No	rw
			1	<i>Pulses-Direction</i>				
			2	<i>CW-CCW pulses</i>				

This parameter set the type of external encoder signal.

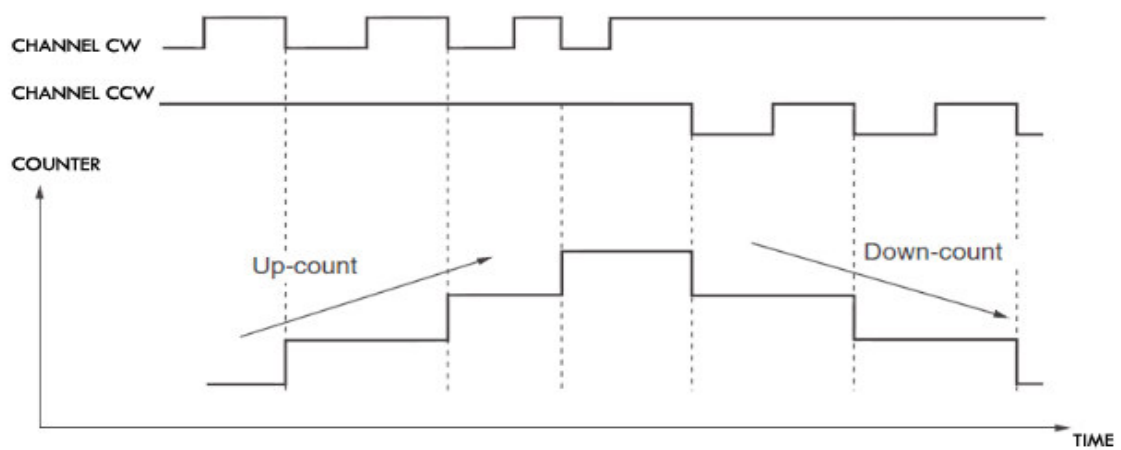
Selecting “A-B Channel”, consider that the drive is in the position to count all the pulses of the two channels of the encoder, that means that the resolution in 1 turn is 4 times the encoder resolution. For example if the encoder gives 1024 pulses/revolution the drive can count 4096, so you must consider this when you set parameter of *Numerator Ratio* (index 2200h) e *Denominator Ratio* (index 2201h) that manage the electric axis *ratios*.



Drawing 23: Encoder mode "Channel A-B"



Drawing 24: Encoder mode "Frequency and Direction"



Drawing 25: Encoder mode "Channel CW-CCW"

4.14.8 Object 2205 – *Unlock position absolute/relative*

Index	Name	type	Range		Default	Unit	Map	Attributes
2205h	<i>Unlock position abs/rel</i>	UINT8	0	<i>Absolute</i>	1		No	rw
			1	<i>Relative</i>				

The value of this parameter is used to set the type of unlock quota, and can be relative compared to the unlock position or absolute. Absolute means that is calculated starting from the zero of the axis.

4.14.9 Object 2206 – *Lock in Speed-acceleration*

Index	Name	type	Range	Default	Unit	Map	Attributes
2206h	<i>Lock in speed-acceleration</i>	UINT32	1..6000000	5000	rpm/s	No	rw

This parameter sets the max acceleration possible during the phase of lock in speed ramp.

4.14.10 Object 2207 – *Lock in speed-Jerk*

Index	Name	type	Range	Default	Unit	Map	Attributes
2207h	<i>Lock in speed-Jerk</i>	UINT32	1..6000000	30000	rpm/s ²	No	rw

This parameter sets the max variation of speed possible during the lock phase in speed ramp.

4.14.11 Object 2208 – *Lock in position-quota*

Index	Name	type	Range	Default	Unit	Map	Attributes
2208h	<i>Lock in position quota</i>	UINT32	0..2147483647	65536		No	rw

When the *Lock mode* (index 2202h) is set in “lock in position”, this parameter indicates the space to cover to bring the axis at the same speed of the position reference; The space to covered in the phase of lock correspond to the space given by the encoder multiplied for the electric axis during the lock time.

4.14.12 Object 2209 – *Unlock in speed-deceleration*

Index	Name	type	Range	Default	Unit	Map	Attributes
2209h	<i>Unlock in speed-deceleration</i>	UINT32	1..6000000	5000	rpm/s	No	rw

The parameter *Unlock in speed-deceleration* set the max deceleration during the phase of unlock in ramp of speed.

4.14.13 Object 220A – *Unlock in speed-Jerk*

Index	Name	type	Range	Default	Unit	Map	Attributes
220Ah	<i>Unlock in speed-Jerk</i>	UINT32	1..6000000	30000	rpm/s ²	No	rw

This parameter set the max variation of deceleration allowed during the phase of unlock in ramp of speed.

4.14.14 Object 220B - *Unlock in position-speed*

Index	Name	type	Range	Default	Unit	Map	Attributes
220Bh	<i>Unlock in position-speed</i>	UINT16	1..10000	1000	rpm	No	rw

The parameter *Unlock in position-speed* set the max speed reachable during the execution of the quota during the unlock phase

4.14.15 Object 220C - *Unlock in position-deceleration*

Index	Name	type	Range	Default	Unit	Map	Attributes
220Ch	<i>Unlock in position-deceleration</i>	UINT16	1..60000	2000	rpm/s	No	rw

This parameter set the max variation of the deceleration during the phase of unlock in position.

4.14.16 Object 220D - *Unlock in position-Quota*

Index	Name	type	Range	Default	Unit	Map	Attributes
220Dh	<i>Unlock in position-quota</i>	INT32	± 2147483647	65536		No	rw

This value indicates the quota (space) covered during the phase of unlock, following parameter *Unlock position absolute/relative* (index 2205h), the shift can be of type relative compared to the position at the moment of the unlock or absolute in reference to the position set in parameter *Position actual value* (index 6064h); The value *Position actual value* can be reset to zero switching the bit 6 of the *control word* from 0 to 1, when the electric axis function is not active.

4.14.17 Object 220E - *Encoder pulses*

Index	Name	type	Range	Default	Unit	Map	Attributes
220Eh	<i>Encoder pulses</i>	INT32	± 2147483647			Yes	ro

This parameter gives the pulses read by the external encoder. This value can be reset to zero switching the bit 5 of the *Questo* parameter. Il numero di impulsi letti dall'encoder esterno, il valore di questa variabile può essere azzerato commutando il bit 5 of the *control word* from 0 to 1, when the electric axis function is not active.

4.14.18 Object 220F – *Pulse Encoder master*

Index	Nome	Tipo	Range	Default	Unit	Map	Attributi
220Fh	<i>Pulse Encoder master</i>	UINT32	100..65536	1024		No	rw

This parameter sets the number of pulses per revolution encoder master, if there is a type encoder "ChannelA-B", the number of pulses per revolution encoder is internally multiplied by 4, because in this mode, the drive acquires the pulse encoder with a resolution 4 times greater.

5 Specific parameters of the Drive

This paragraph shows and explains some specific parameters of the drive, useful to obtain further controls and perform other settings

Index	Name	type	Map	Attributes
2000h	<i>Rated speed of the motor (*)</i>	UINT16	No	rw
2001h	<i>Rated current of the motor (*)</i>	UINT16	No	rw
2002h	<i>Peak current of the motor (*)</i>	UINT16	No	rw
2003h	<i>Pole pairs of the motor (*)</i>	UINT16	No	rw
2004h	<i>Offset resolver (*)</i>	UINT16	No	rw
2005h	<i>Speed limit</i>	UINT16	No	rw
2006h	<i>Current limit</i>	UINT16	No	rw
2007h	<i>Kp position regulator</i>	UINT16	No	rw
2008h	<i>Kp speed regulator</i>	UINT16	No	rw
2009h	<i>Ki speed regulator</i>	UINT16	No	rw
200Ah	<i>Kp current regulator</i>	UINT16	No	rw
200Bh	<i>Ki current regulator</i>	UINT16	No	rw
200Ch	<i>Enable Notch Filter (*)</i>	UINT8	No	rw
200Dh	<i>Notch filter frequency</i>	UINT16	No	rw
200Eh	<i>Band width of notch filter</i>	UINT16	No	rw
200Fh	<i>I2t management Mode(*)</i>	UINT8	No	rw
2010h	<i>“Enable” Input management (*)</i>	UINT8	No	rw
2011h	<i>Jerk position</i>	UINT32	No	rw
2012h	<i>Measured current</i>	INT16	Si	ro
2013h	<i>Motor brake time coupling</i>	UINT16	No	rw
2014h	<i>Motor Brake Release time</i>	UINT16	No	rw
2015h	<i>Brake motor deceleration</i>	UINT16	No	rw
2016h	<i>Brake motor speed intervention</i>	UINT16	No	rw
2017h	<i>Enable Brake motor (*)</i>	UINT8	No	rw
2018h	<i>Enable input 7</i>	UINT8	No	rw
2019h	<i>Relative shift offset input 7</i>	UINT32	No	rw
201Ah	<i>Enable Position Limit</i>	UINT8	No	rw
2020h	<i>ADC External</i>	INT16	Yes	ro

Tabella 36: Specific parameters of the Drive

(*) These parameters can be modified only when power is disabled, if you try to modify

one of these parameters when power is on you receive an alarm code (hexadecimal)
08000022.

5.1.1 Object 2000h – Motor Rated Speed

Index	Name	type	Range	Default	Unit	Map	Attributes
2000h	<i>Motor Rated Speed</i>	UINT16	±9999	3000	rpm	No	rw

This parameter indicates the rated speed of the motor expressed in rpm

5.1.2 Object 2001h – Motor Rated Current

Index	Name	type	Range	Default	Unit	Map	Attributes
2001h	<i>Motor Rated current</i>	UINT16	0..60000	100	A	No	rw

This parameter shows the rated current of the motor expressed in Ampere. The two less significant hexadecimal digits are interpreted by the drive as decimal fraction in Ampere , for example the value 100 is considered as 1,00 A.

5.1.3 Object 2002h – Motor Peak Current

Index	Name	type	Range	Default	Unit	Map	Attributes
2002h	<i>Motor peak current</i>	UINT16	0..60000	100	A	No	rw

This parameter indicates the peak current of the motor expressed in Ampere. The two less significant hexadecimal digits are interpreted by the drive as decimal fraction in Ampere, for example the value 200 is considered as 2,00 A.

5.1.4 Object 2003h – Pole pairs of the motor

Index	Name	type	Range	Default	Unit	Map	Attributes
2003h	<i>Pole pairs of the motor</i>	UINT16	1..8	3		No	rw

The parameter *Pole pairs of the motor* corresponds to half of the poles of the motor. The modification of this parameter is possible modifying this data; for example for a motor with 6 poles you must write value 3.

5.1.5 Object 2004h – Offset resolver

Index	Name	type	Range	Default	Unit	Map	Attributes
2004h	<i>Offset resolver</i>	UINT16	0..36000	0	°	No	rw

Parameter that indicates the position of offset of the resolver expressed in degree. The two less significant hexadecimal digits are interpreted by the drive as decimal fraction of degree, for example the value 18000 is considered as 180,00 °.

5.1.6 Object 2005h – Speed Limit

Index	Name	type	Range	Default	Unit	Map	Attributes
2005h	<i>Speed limit</i>	UINT16	±9999	5000	rpm	No	rw

This parameter indicates the max limit, in absolute value, expressed in rpm unit, of the speed reference.

5.1.7 Object 2006h – Current Limit

Index	Name	type	Range	Default	Unit	Map	Attributes
2006h	<i>Current limit</i>	UINT16	0..300	200	%	No	rw

This parameter allows to set the max limit, in absolute value, of the current, expressed in percent (%) of the rated current of the motor. For example if the rated current of the motor is 2,00 A and the limit of current is set at 200 %, the max current allowed will be 4,00 A.

5.1.8 Object 2007h – Kp position regulator

Index	Name	type	Range	Default	Unit	Map	Attributes
2007h	<i>Kp position regulator</i>	UINT16	0..4000	512		No	rw

This parameter is used to set the proportional gain of the position regulator.

5.1.9 Object 2008h – Kp speed regulator

Index	Name	type	Range	Default	Unit	Map	Attributes
2008h	<i>Kp speed regulator</i>	UINT16	1..3000	1000		No	rw

This parameter is used to set the proportional gain of the speed regulator.

5.1.10 Object 2009h – Ki speed regulator

Index	Name	type	Range	Default	Unit	Map	Attributes
2009h	<i>Ki speed regulator</i>	UINT16	0..3000	300		No	rw

This parameter is used to set the integral gain of the speed regulator.

5.1.11 Object 200Ah – Kp current regulator

Index	Name	type	Range	Default	Unit	Map	Attributes
200Ah	<i>Kp current regulator</i>	UINT16	1..2000	1200		No	rw

This parameter is (reserved) used to set the porportional gain of the current regulator.

5.1.12 Object 200Bh – Ki current regulator

Index	Name	type	Range	Default	Unit	Map	Attributes
200Bh	<i>Ki current regulator</i>	UINT16	1..2000	200		No	rw

This parameter is (reserved) used to set the integral gain of the current regulator.

5.1.13 Object 200Ch – Eanble Notch Filter

Index	Name	type	Range	Default	Unit	Map	Attributes
200Ch	<i>Eanble Notch Filter</i>	UINT8	0..1	0		No	rw

This parameter is used to enable/disable the Notch filter, writing the value 1 the filter is enabled, writing 0 it's disabled.

5.1.14 Object 200Dh – Notch filter frequency

Index	Name	type	Range	Default	Unit	Map	Attributes
200Dh	<i>Notch filter frequency</i>	UINT16	50..400	220		No	rw

This parameter is used to set the frequency of work of the Notch filter.

5.1.15 Object 200Eh – Band with of notch filter

Index	Name	type	Range	Default	Unit	Map	Attributes
200Dh	<i>Band with of notch filter</i>	UINT16	8000..9990	9500		No	rw

This parameter is used to change the band width of the Notch filter.

5.1.16 Object 200Fh – I2t management Mode

Index	Name	type	Range	Default	Unit	Map	Attributes
200Fh	<i>I2t management Mode</i>	UINT8	0..1	0		No	rw

The parameter allows to choose the mode to manage the status of I2t of the drive:

Value	Description
0	When the I2t of the drive intervenes the max current deliverable is limited to the rated current value, and this value becomes independent from the temperature of the drive.
1	When the I2t of the drive intervenes and the temperature estimated result to be below to the max temperature, the I2t alarm is deleted automatically.

Tabella 37: I2t management mode

5.1.17 Object 2010h – “Enable” Input management

Index	Name	type	Range	Default	Unit	Map	Attributes
2010h	<i>“Enable” input management</i>	UINT8	0..1	0		No	rw

This parameter is used to set the mode to manage the input 0:

Value	Description
0	In this case input 0 assume become a generic input.
1	In this case the changement to a status of “operation Eanble” (that means power enabled) is allowed only when at the input 0 is present a 24V, otherwise the power is disabled.

Tabella 38: “Enable” input management

5.1.18 Object 2011h – Jerk position

Index	Name	type	Range	Default	Unit	Map	Attri- butes
2011h	<i>Jerk position</i>	UINT32	1..FFFFFFFFh	100000		No	rw

The *Jerk position* is used in the positioner when parameter *Motion profile type* (index 6097h) is set with value 3 “*Jerk-limited ramp*”, and it manage th cubic shape of the position curve modifying the variation of acceleration in the positioning profiles. A high value of Jerk allows bigger variation of acceleration. The consequence is a time reduction in the execution of the position and a bigger machanical stress. The other way around reducing the Jerk, the accelertion and the time to positioning increase but the mechanical stress reduces.

The unit of measure of this parameter is an acceleration divided by a time. Inside the drive it is calculated as increments of the resover over second elevated at third [increments/s³].To convert it in the internal measure of the drive you must use the *Acceleration factor* (index 6097h) following this formula:

$$Jerk\ Internal\ position = \frac{(Numerator \times Jerk\ Position)}{Divisor}$$

Using the values of default of the *Acceleration factor*, that means *Numerator*=65536 and *Divisor*=60, the *Jerk position* is expressed in revolution/s³ or [rpm/s²]

5.1.19 Object 2012h – Measured current

Index	Name	type	Range	Default	Unit	Map	Attri- butes
2012h	<i>Measured current</i>	INT16			A	Yes	ro

This parameter give back an everage value of the torque current expressed in Ampere and this data is updated every 60 ms.

The value of the *Measured current* is writte in centesimal fo Ampere, for example when the read numeber is 132 this correspond to 1,32 A.

5.1.20 Object 2013h – Motor brake time coupling

Index	Name	type	Range	Default	Unit	Map	Attributes
2013h	<i>Motor brake time coupling</i>	UINT16	10..2000	200	ms	No	rw

This is the parameter that indicates the time necessary to activate the motor brake.

5.1.21 Object 2014h – Motor Brake Release time

Index	Name	type	Range	Default	Unit	Map	Attributes
2014h	<i>Motor Brake Release time</i>	UINT16	10..2000	200	ms	No	rw

This parameter indicates the time necessary to deactivate (release) the motor Brake

5.1.22 Object 2015h – Time of Brake motor deceleration

Index	Name	type	Range	Default	Unit	Map	Attributes
2015h	<i>Brake motor deceleration</i>	UINT16	1..60000	1000	rpm/s	No	rw

Value of the deceleration ramp of speed active during the phase of arrest of the motor.

5.1.23 Object 2016h – Brake motor speed intervention

Index	Name	type	Range	Default	Unit	Map	Attributes
2016h	<i>Brake motor speed intervention</i>	UINT16	1..500	4	rpm	No	rw

This parameter indicates the minimum speed below which is activated the motor brake during the phase of arrest.

5.1.24 Object 2017h – Enable Brake motor

Index	Name	type	Range	Default	Unit	Map	Attributes
2017h	<i>Enable Brake motor</i>	UINT8	0..1	0		No	rw

This parameter is used to activate the management of the motor brake. When it is set to value 1 the procedure that manages the brake is active. This parameter is changeable only when power is disabled.

5.1.25 Object 201Ah – Enable Position Limit

Index	Nome	Type	Range	Default	Unit	Map	Attributes
201Ah	Enable Position Limit	UINT8	0..1	0		No	rw

The position limits set in the vector *software position limit* (index 607Ah), are activated by placing 1 at the *Enable Position Limit*.

5.1.26 Object 2020h – ADC External

Index	Nome	Type	Range	Default	Unit	Map	Attributes
2020h	ADC External	INT16	-32678..+32677			Si	ro

This parameter give bak the voltage value acquired by the external ad-converter. The value of voltage have a renga from ± 10 V. The data detected have an accuracy of 16 bit.

5.2 Function of relative shift related to input 7

In the DGM drive there is a function that allows to execute a shift related to the position measured when input 7 is sampled at a high logic level; this mode can work only when parameter *Motion profile type* (index 6086h) is set to work with linear ramps, that means when it is set to value 0 or -1; in the following paragraphs are shown the others parameters to manage this function.

5.2.1 Object 2018h – Enable input 7

Index	Name	type	Range	Default	Unit	Map	Attributes
2018h	Enable input 7	UINT8	0..1	0		No	rw

Enable or disable through SDO the function of relative positioning in association with input 7. The possible values are:

- 0: function related to input 7 **disabilitata**.
- 1: function related to input 7 **abilitata**.

5.2.2 Object 2019h – Relative shift offset input 7

Index	Name	type	Range	Default	Unit	Map	Attri-
-------	------	------	-------	---------	------	-----	--------

							buses
2019h	<i>Relative shift offset input_7</i>	UINT32	0..7FFFFFFFh	65535		No	rw

It indicate the space to cover compared to the position measured on the logic high level of the input 7, used to start the function that manages the relative positioning; The data of this parameter is considered without sign, as the direction of the movement is calculated by the software in agreement with the direction of the measured speed.

5.2.3 Bit 14 status word for detection of positioning input 7

To verify the relative positioning has come, in consequence of the sampling of the first high logic value detected from input 7, the software allocates the bit 14 of the *status word*, that is active only when the parameter *Enable input 7* (index 2018h) is set at value 1; this bit is considered only when the quota is reached, that is when bit 10 (Target reached) of the *status word* is placed at value 1, in that moment the bit 14 can take one of these values:

- 0: quota reached without detection of high value on input 7
- 1: quota reached (position reached) with detection of high value at the input 7 and execution of a relative shift set in parameter *Relative shift offset input 7* (index 2019h).

5.2.4 Relative shiftment function management from input 7

The function of relative shift is activated from input 7 placing 1 in the parameter *Enable input 7* (index 2018h), In this mode the software of the drive automatically starts to manage the relative positioning routines from input 7 at the starting of every quota, that is when the bit 4 (New set-point) is placed at value 1.

When the first logic value high is detected from input 7, the drive executes a movement related to the position sampled, with the same direction of the speed detected and with the value set in the parameter *Relative shift offset input 7* (index 2019h).

When the position is reached, the bit 14 of the *status word*, is placed at 1 when is detected an high logic value that entails the shift set, while the *status word* is 0 in case input 7 is always at low logic level.