Tritex II

CANopen - Option

Contents

Introduction

1. CAN basics

The CAN Bus is an automotive bus developed by Robert Bosch, which has quickly gained acceptance into the automotive and aerospace industries. CAN is a serial bus protocol to connect individual systems and sensors as an alternative to conventional multi-wire looms. It allows automotive components to communicate on a single or dual-wire networked data bus up to 1Mbps

1.1. Data Frame

The CAN protocol uses a modified version of the Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA) technique used on Ethernet. Should two messages determine that they are both trying to send at the same time then instead of both backing off and re-trying later as is done with Ethernet, in the CAN scheme, the transmitters detect which message has the highest priority and only the lower priority message gets delayed. This means that a high priority message is sure of getting through

SOF	Message ID	RTR	Control	Data	CRC	ACK	EOF

Figure 1 - CAN Date frame

These are the normal message frames used to carry data.

- Start of frame (SOF)
- Message Identifier (MID) the Lower the value the Higher the priority of the message its length is either 11 or 29 bits long depending on the standard being used
- Remote Transmission Request (RTR) = 0
- Control field (CONTROL) this specifies the number of bytes of data to follow (0-8)
- Data Field (DATA) length 0 to 8 bytes
- CRC field containing a fifteen bit cyclic redundancy check code
- Acknowledge field (ACK) an empty slot which will be filled by any and every node that receives the frame it does NOT say that the node you intended the data for got it, just that at least one node on the whole network got it.
- End of Frame (EOF)

The way in which message collision is avoided is that each node as it transmits its MID looks on the bus to see what everyone else is seeing. If it is in conflict with a higher priority message identifier (one with a lower number) then the higher priority messages bit will hold the signal down (a zero bit is said to be dominant) and the lower priority node will stop transmitting.

1.2. Error Control

The CAN specification defines several error control mechanisms, which makes it a very reliable network and with a very low rate of undetected transmission errors. Each device in the network must be able to identify the occurrence of those errors and to inform the other elements that an error was detected.
A CAN network device has internal counters that are incremented every time a transmission or reception error is detected, and decremented when a telegram is sent or received with success. If a considerable amount of errors occurs, the device can be taken to the following conditions:

- *Warning*: when this counter exceeds a certain limit, the device enters the *warning* state, meaning the occurrence of a high error rate.

- *Error Passive*: when this value exceeds a higher limit, the device enters the *error passive* state, then it stops acting in the network when detecting that another device sent a telegram with error.

- *Bus Off*: to conclude, there is the *bus off* state, in which the device will no longer send or receive telegrams.

Refer to Status CANopen for more information.

1.3. Baud rate

The CANopen DS 102 V2.0 Physical Layer for Industrial Applications, recommended guidelines for cable length at several bit rates. See table below

Bit rate	Bus length	Bit time
1Mbits/s	25 meters	1 us
800 kbits/s	50 meters	1.25 us
500 kbits/s	100 meters	2 us
250 kbits/s	250 meters	4 us
125 kbits/s	500 meters	8 us
100 kbits/s	750 meters	10 us
50 kbits/s	1000 meters	20 us
20 kbits/s	2500 meters	50 us

Figure 2 - Recommended baud rates and bit timings

2. CANopen

CANopen is a communication protocol especially dedicated to industrial applications. It allows connecting up to 127 different devices on a same bus giving them the possibility to access the bus at any time. Simultaneous emissions are managed by an arbitration system that uses priority levels. This control hierarchy of data transfer guarantees that there is no frame collision on the bus while ensuring a high level of reliability in communications. The low priority messages are canceled and reissued after a delay.

The protocol defines several messages types characterized by their COB-ID (Communication Object Identifier) that determines the message priority level. The COB-ID is composed of a function code and the node identifier (between 1 and 127).

The node identifier is the device's address on the network. The function code specifies the priority and the purpose of the message.

There are 6 different message types:

Read/write requests:	SDO (Service Data Objects)
Real time transfers:	PDO (Process Data Object)
Nodes state management:	NMT (Network Management)
Warnings/ Errors:	EMCY (Emergency)
Synchronization events:	SYNC (Synchronization)
Node state indications:	Boot-up/Heartbeat and Node guarding

CANopen messages	Function Code	COB-ID
NMT	0000	0x000
SYNC	0001	0x080
Time Stamp	0010	0x100
EMCY	0001	0x080 + Module-ID
TPDO1	0011	0x180 + Module-ID
RPDO1	0100	0x200 + Module-ID
TPDO2	0101	0x280 + Module-ID
RPDO2	0110	0x300 + Module-ID
TPDO 3	0111	0x380 + Module-ID
RPDO 3	1000	0x400 + Module-ID
TPDO 4	1001	0x480 + Module-ID
RPDO 4	1010	0x500 + Module-ID
SDO (Tx)	1011	0x580 + Module-ID
SDO (Rx)	1100	0x600 +Module-ID
Heartbeat / Boot-up / Node Guarding	1110	0x700 + Module-ID

Figure 3 – COB-ID

3. NMT

The network management provides a set of cross-system services to control the communication status of CANopen devices. After start-up, a CANopen device passes through the state initializing and then automatically switches to the state pre-operational. With the NMT service Start Remote Node, a NMT master can set the state of a selected slave to operational.

The communication via SDO is possible for the states pre-operational and operational. The communication via PDO is only possible for the state operational.

3.1. NMT State Machine

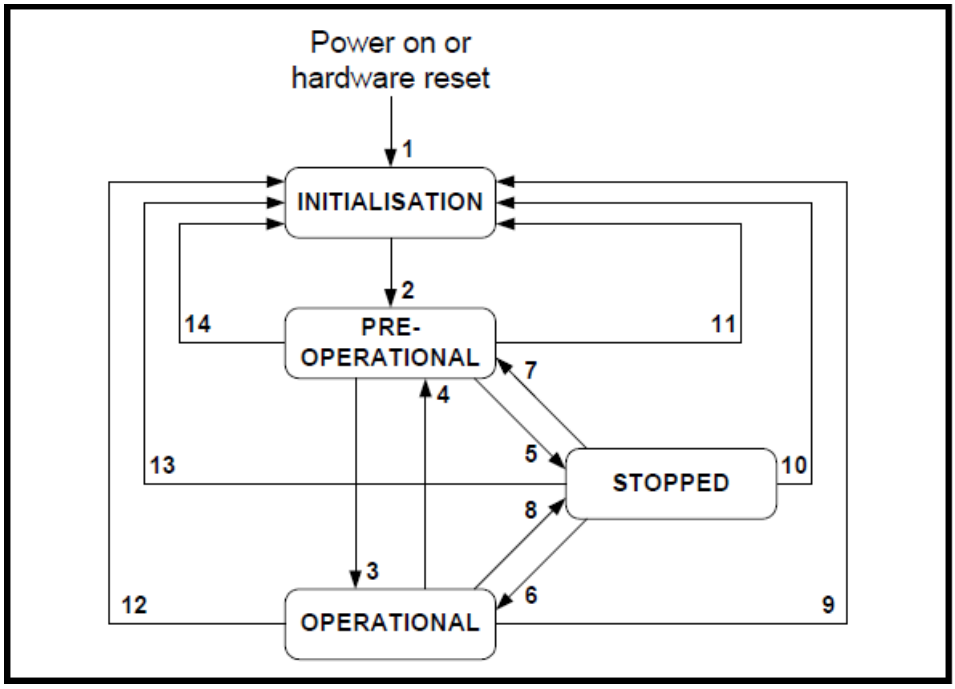


Figure 4 - NMT state machine

Transition	Transition initiated by
1	At power on or hardware reset, enter INITIALISATION automatically
2	INITIALISATION complete, enter PRE-OPERATIONAL automatically
3, 6	START_REMOTE_NODE
4, 7	ENTER_PRE_OPERATIONAL
5, 8	STOP REMOTE NODE
9, 10, 11	RESET_NODE
12, 13, 14	RESET_COMMUNICATION

Figure 5 - NMT state machine transitions

3.2. NMT Commands

Network Management (NMT) commands are low-level CANopen commands that are used to switch SM-CANopen between the different NMT states. NMT messages always have a CAN identifier of 0x000 and contain 2 data bytes

States	NMT Commands	Meaning (CANopen)
Pre- operational	0x80	Communication via SDO's possible PDO's inactive (No sending / receiving)
Operational	0x01	Communication via SDSO's possible PDOs active (sending /receiving)
Stop	0x02	No Communication except heartbeat + NMT Refer to controller access for more information
Reset Communication	0x81	Refer to controller access for more information Reset CANopen interface to startup condition except for baud-rate and ID.
Reset Node	0x82	Refer to controller access for more information CANopen interface will internally send DS402 module clear controlword – will not return other DS402 states to power-up condition

Figure 6 - NMT Commands

3.3. NMT States

NMT States	Value in Hex
Pre- operational	0x7F
Operational	0x05
Stopped	0x04

Figure 7 - NMT States

3.4. NMT States and Communication

The relation between NMT states and communication objects; services on the listed communication objects may only be executed if the CANopen devices involved in the communication are in the appropriate NMT states.

	Pre-Operational	Operational	Stopped
PDO		X	
SDO	X	X	
SYNC	X	X	
TIME	X	X	
EMCY	X	X	
Node control and error control	X	X	X

Figure 8 - NMT states and Communication Objects

3.5. NMT Error Control

3.5.1. Node guarding / Life guarding

The Node Guarding protocol is a functional monitoring for the drive. It requires that the drive is accessed at regular intervals by the CANopen master. The maximum time interval that is permitted between two Nodeguard telegrams is given by the product of the *Guard Time* and the *Life Time Factor*. If one of these two values is 0, then the response monitoring is de-activated.

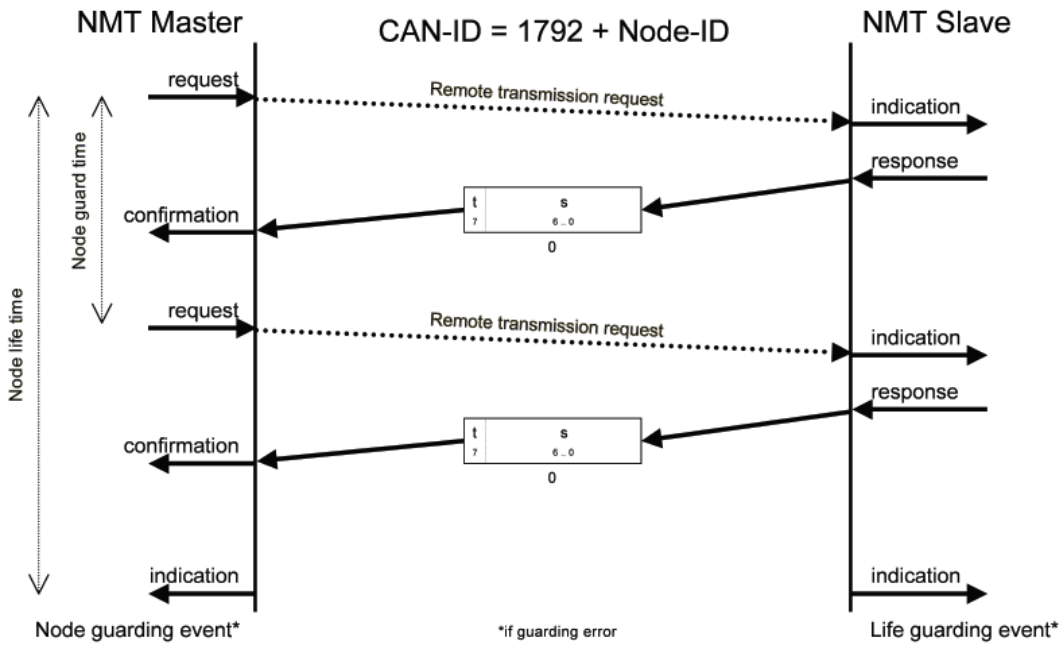


Figure 9 - Guard Time Protocol

The NMT master polls each NMT slave at regular time intervals. This time-interval is called the guard time and may be different for each NMT slave. The response of the NMT slave contains the NMT state of that NMT slave. The node lifetime is given by the guard time multiplied by the lifetime factor. The node lifetime may be different for each NMT slave. If the NMT slave has not been polled during its lifetime, a remote node error is indicated through the NMT service life guarding event.

A remote node error is indicated through the NMT service node guarding event if
- The RTR is not confirmed within the node life time
- The reported NMT slave state does not match the expected state

3.5.2. Heartbeat

The Heartbeat protocol is for error control purposes and signals the presence of a node and its state. The Heartbeat message is a periodic message of the node to one or several other nodes. It indicates that the sending node is still working properly.

COB-ID	Byte 0
0x700 + Node-ID	NMT-State

Figure 10 – NMT Heartbeat

NMT States	Value in Hex
Pre- operational	0x7F
Operational	0x05
Stopped	0x04

Figure 11- Heartbeat NMT States

If the heartbeat producer time is configured on a CANopen device the heartbeat protocol begins immediately. If a CANopen device starts with a value for the heartbeat producer time unequal to 0 the heartbeat protocol starts on the transition from the NMT state Initialization to the NMT state Preoperational. In this case the boot-up message is regarded as first heartbeat message. It is not allowed to use both error control mechanisms guarding protocol and heartbeat protocol on one NMT slave at the same time. If the heartbeat producer time is unequal 0 the heartbeat protocol is used

3.6. Boot up Message

A device sends the Boot-up message to indicate to the NMT master that it has reached the state Preoperational.
This occurs whenever the device initially boots-up but also after a power-out during operation. The Boot-up message has the same identifier as the Heartbeat object, but its data content is zero.

COB-ID	Byte 0
0x700 + Node-ID	0

Figure 12 – NMT Boot up

3.7. SYNC Service

The Synchronization (SYNC) Object is periodically broadcast by the SYNC Producer. The time period between SYNC messages is defined by the Communication Cycle Period, which may be reset by a configuration tool to the application devices during the boot-up process. There can be a time jitter in transmission by the SYNC Producer due to some other objects with higher prior identifiers or by one frame being transmitted just before the SYNC message. The SYNC message is mapped to a single CAN frame with the identifier 128 by default and does not carry any data.

3.8. Emergency

The Emergency object COB-ID is 0x81 to 0xFF. EMCY objects are fully defined in CiA DS 301. The structure of the emergency message is as follows:

Error code		Error register	Manufacture Error Codes				
CAN Data bytes							
0	1	2	3	4	5	6	7
Error code		1001.0[1]	0	0	0	0	0

Figure 13 - Emergency Message Format

Error Code	Description
0000	Drive Fault Reset command perform
5200	[2]System Error – module dependent fault
8144	CANopen Defaults loaded
8130	Heartbeat / Life Guard Error
8220	PDO Data Length
8180	CANopen Transmission Timeout Error
8100	CANopen Communication Error
FF00	CANopen No rights to command

Figure 14 - Emergency Error codes

(1) Data from CANopen Error Register (1001.0).

(2) Individual faults must be read through SDO's, PDOs or Modbus. Refer to Diagnostic section of manual for more information

4. Service Data Object (SDO)

A Service Data Object (SDO) is used to read entries from, or write entries to, a module's object dictionary. The SDO transport protocol allows transmitting objects of any size.

COB-ID	Command Specific	Index		Sub-Index	Data			
		(low)	(high)		(LSB)			(MSB)

Figure 15 - Expedited CANopen message frame

Access mode	Number of data bytes	Command specific
Read Request (Initiate Domain upload)		40h
Read Response (Initiate Domain upload)	1	4Fh
	2	4Bh
	3	47h
	4	43h
Write Request (Initiate Domain upload)	1	2Fh
	2	2Bh
	3	27h
	4	23h
Write Response (Initiate Domain Download)		60h
Error Response (SDO abort)		80h

Figure 16 - Expedited commands

Index and sub-index identity the parameters in the object dictionary.

4.1. SDO Abort Transfer Protocol

SDO messages use a request-response mechanism and the CANopen master will always expect a response from the slave device. If an error occurs with an SDO transfer Tritex will return an SDO abort code to indicate the reason for the failure, the SDO abort codes are listed below:

Error code		Description
0504	0001h	SDO transmission was aborted

Figure 17 - SDO Abort codes

5. Communication Profile

The Communication Profile Area contains the communication specific parameters for the CAN network. The Standardized Device Profile Area contains all data objects common to a class of devices that can be read or written via the network. The device profiles may use these entries to describe the device parameters and device functionality.

Index	sub index	Description	Type	Attr
1000	0	Device Type	UINT32	RO
1001	0	Error Register	UINT8	RO
1005	0	COB ID SYNC	UINT32	RW
100C	0	Guard Time	UINT16	RW
100D	0	Life Time factor	UINT8	RW
1010	-	Save Parameters	-	-
1014	0	COB-ID EMCY	UINT32	RW
1015	0	Inhibit time EMCY	UINT16	RW
1017	0	Producer Heartbeat time	UINT16	RW
1018	-	Identity	-	-
2000	0	Node ID	UINT8	RW
2001	0	CAN baud rate	UINT16	RW

Figure 18 - Communication Objects

5.1. Device Type (1000)

The object specifies the device profile used as well as the device type

Index	sub index	Description	Type	Default	Attr	NVM	Mappable
1000	0	Device Type	UINT32		RO		

Figure 19 - Device type

5.2. Error Register (1001)

Provide error information……. Expanded error/warnings are found CANopen status.

Index	sub index	Description	Type	Norm	Attr	NVM	Map able
1001	0	Error Register	UINT8		RO		*

Figure 20 - Error register object

Bit	Meaning
0	Generic error
1	Current
2	Voltage
3	Temperature
4	Communication error (overrun, error state)
5	DS402 Device profile error
6	Reserved (always 0)
7	Manufacture –specific (always 0)

Figure 21 - Error register flags

Note: In case of a device internal error, Bit 0 of Object 1001_h will be 1, further bits may be set additionally.

5.3. COB-ID SYNC (1005)

Indicate the configured COB-ID of the synchronization object (SYNC). Further, it defines whether the CANopen device generates the SYNC.

Index	sub index	Description	Type	Default	Attr	NVM	Map able
1005	0	COB ID SYNC	UINT32		RW	*	

Bit	Value	Description
31 (MSB)	x	--.
30	0	Device not generate SYNC message
	1	Device generates SYNC message (does not support)
29	0	11 Bit ID (CAN 2.0A)
	-	29 Bit ID (CAN 2.0B)
28 - 11	X	If Bit 29 = 1 => Bit - 28 of 29-bit SYNC COB-ID
	0	If Bit 29 = 0
10 - 0 (LSB)	x	Bit 0 ... 10 of SYNC COB-ID

Figure 22 – COB-SYNC ID

The device does not support the generation of SYNC-messages and only the 11-bit IDs. So the bits 11 to 30 are always 0.

5.4. Guard Time (100C)

The objects at index $100C_h$ and $100D_h$ configures the guarding time (1msec per count) of the Tritex. The life time factor multiplied with the guard time gives the life time for the life guarding protocol.

Index	sub index	Description	Type	Default	Attr	NVM	Map able
100C	0	Guard Time	UINT16	0	RW	*	

Figure 23 - Guard time object

5.5. Life Time Factor (100D)

The life time factor multiplied with the guard time gives the life time for the life guarding protocol

Index	sub index	Description	Type	Default	Attr	NVM	Map able
100D	0	Life Time factor	UINT8	0	RW	*	

Figure 24 - Life time factor object

5.6. Save Parameters (1010)

This object controls the saving of parameters in non-volatile memory

Index	sub index	Description	Type	Norm	Attr	NVM	Map able
1010	-	Save Parameters	-		-		
	0	Number of Entries	UINT8		RO		
	1	Save all Parameters[1)]	UINT32		RW		
	2	Save CAN Comm. Parameters	UINT32		RW		

Figure 25 - Save parameters object

1) Dependent if controller has write access.

In order to avoid storage of parameters by mistake, storage will be executed only when a specific signature is written to the appropriate sub-index. The signature that save parameters is "save"

MSB

e	v	a	s
65_h	76_h	61_h	73_h

Figure 26 - Storage write access signature

On read access to the appropriate sub-index the CANopen device will provide information about its storage functionality with the following format:

31	2	1	0
00000000000000000000000000000000		auto	cmd
MSB			LSB

Bit	Value	Description
auto	-	CANopen device does not save parameters autonomously
	1	CANopen device saves parameters autonomously
cmd	0	CANopen device does not save parameters on command
	1	CANopen device saves parameters on command

Figure 27 - Save parameters read parameters

5.7. COB-ID Emergency (1014)

This object indicates the configured COB-ID for the EMCY write service.

Index	sub index	Description	Type	Default	Attr	NVM	Map able
1014	0	COB-ID EMCY	UINT32	80_h + Node - ID	RW	*	

Figure 28 - Emergency COB-ID object

Bit	Value	Description
31 (MSB)	x	--.
30	0	Device not generate EMCY message
	1	Device generates EMCY message
29	0	11 Bit ID (CAN 2.0A)
	-	29 Bit ID (CAN 2.0B)
28 - 11	X	If Bit 29 = 1 => Bit - 28 of 29-bit EMCY COB-ID
	0	If Bit 29 = 0
10 - 0 (LSB)	x	Bit 0 ... 10 of EMCY COB-ID

Figure 29 - Emergency COB-ID format

5.8. Inhibit time EMCY (1015)

This object indicates the configured inhibit time for the EMCY message. The value is given in multiples of 100 us. The value 0 will disable the inhibit time.

Index	sub index	Description	Type	Default value	Attr	NVM	Map able
1015	0	Inhibit time Emergency	UINT16	0	RW	*	

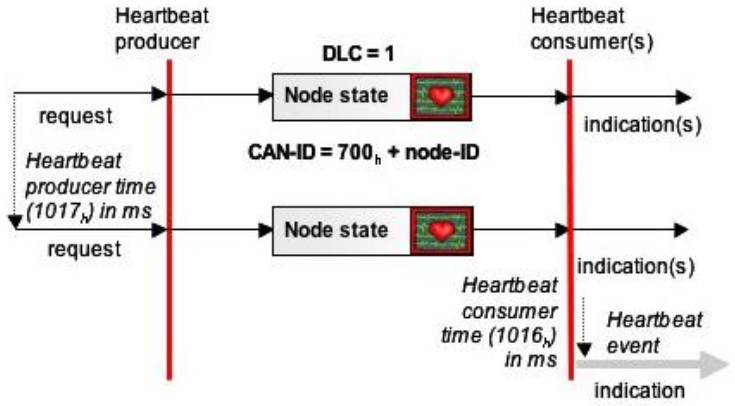
Figure 30 - Inhibit Time Object

5.9. Heartbeat timer (1017)

The producer heartbeat time indicates the configured cycle time of the heartbeat. The value is given in multiples of 1 ms. The value 0 disables the producer heartbeat

Index	sub index	Description	Type	Default	Attr	NVM	Map able
1017	0	Producer Heartbeat time	UINT16	0	RW	*	

Figure 31 - Heartbeat time Object



5.10. Identity (1018)

Describes the identity of the device as a node in the in a CANopen network

Index	sub index	Description	Type	Default	Attr	NVM	Map able
1018	-	Identity	-		-		
	0	Number of Entries	UINT8		RO		
	1	Vendor ID	UINT32		RO		
	2	Product Code	UINT32		RO		

Figure 32 - Identity Object

5.11. Node ID (2000)

CANopen Node ID of the Tritex.

Index	sub index	Description	Type	Default value	Attr	NVM	Map able
2000	0	Node ID	UINT8	127	RW	*	

Figure 33 - Node ID object

Changes do not take effect until, save operation and parameters are reloaded.

5.12. CAN baud rate (2001)

CAN bit-rate of the device. For a new bit-rate to be valid all parameters have to be stored and the device must be reset.

Warning: Operating devices with different bitrates in one CAN network can cause serious problems in the whole network. Therefore, change all devices to one bit-rate before switching on the voltage supply.

Index	sub index	Description	Type	Default value	Attr	NVM	Map able
2001	0	CAN baud rate	UINT16	3 (125 kbits/s)	RW	*	

Figure 34 - CAN baud rate Object

Baudrate	Entry
20 kbits/s	0
50 kbits/s	1
100 kbits/s	2
125 kbits/s	3
250 kbits/s	4
500 kbits/s	5
800 kbits/s	6
1000 kbits/s	7

Figure 35 – Baud rates

6. Process Data Objects (PDOs)

Process data objects (PDOs) are used in CANopen for broadcasting high-priority control- and status information. A PDO consists of a single CAN frame and enables to communicate up to 8 byte of pure application data.

PDO's are validated when drive enters NMT state operational. Refer to Fault / warning section for configuration error detection.

Index	Object	Description	Type	Attr
1400	Record	RPDO-1 communication parameter		
1401	Record	RPDO-2 communication parameter		
1402	Record	RPDO-3 communication parameter		
1403	Record	RPDO-4 communication parameter		
1600	Record	RPDO-1 mapping parameter		
1601	Record	RPDO-2 mapping parameter		
1602	Record	RPDO-3 mapping parameter		
1603	Record	RPDO-4 mapping parameter		
1800	Record	TPDO-1 communication parameter		
1801	Record	TPDO-2 communication parameter		
1802	Record	TPDO-3 communication parameter		
1803	Record	TPDO-4 communication parameter		
1A00	Record	TPDO-1 mapping parameter		
1A01	Record	TPDO-2 mapping parameter		
1A02	Record	TPDO-3 mapping parameter		
1A03	Record	TPDO-4 mapping parameter		

Figure 36 - Dictionary - PDOs

Mapping

Typically there are several RxPDO and TxPDOs available for the process data exchange. The default allocation of input or output data to these PDOs is called default mapping and is defined in the device profiles. Tritex will support 4 RxPDO and 4 TxPDOs.

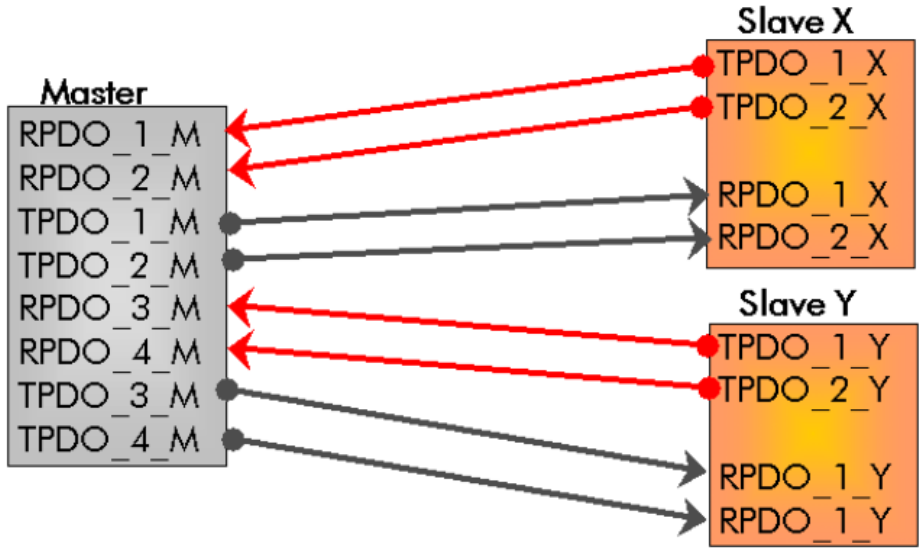


Figure 37 - PDOs Mapping

Communication parameters

The PDOs can be given different communication parameters according to the requirements of the master. Like all parameters they can be found in the object dictionary of the device. The parameters are located in the object directory from index 0x1400 (RxPDOs) and index 0x1800 (TxPDO).

6.1. PDO Transmission Types

The transmission types determine the PDO trigger behavior when receiving and transmitting data. Here is the overall behavior of the transmission types. Note, receive and transmit PDOs do not have the same allowed configuration.

Transmission Type	PDO Transmission				
	Cyclic	acyclic	synchronous	asynchronous	RTR only
0		valid	valid		
1 to 240	valid		valid		
241 to 251	reserved				
252			valid		valid
253				valid	valid
254				valid	
255				valid	

Figure 38 - CANopen Transmission types

Transmission Type	Transmit PDOs		Inhibit Timer *Valid*	Event Timer *Valid*	Trigger Event	
0 [2) 4)]	Source data is read on SYNC TxPDO only transmit if data has changed	Synchronous /acyclic	NO	NO	On change	
1 to 240 [2)]	Source data is read on SYNC TxPDO every nth SYNC message	Synchronous /cyclic	NO	NO	On Sync	Cyclic (Sync Count)
241 to 251 [3)]	- - - - -					
252	Source data is read when the SYNC message is received - TxPDO will only be transmitted when RTR id received	Asynchronous	YES	YES	On Sync On RTR	
253	Source data is read and TxPDO is transmitted when the RTR message is received	Asynchronous	YES	YES	ON RTR	
254, 255	Source data is read and TxPDO is transmitted base on internal event, can be inhibit with Event timer	Asynchronous	YES	YES	On Change	

Figure 39 - TPDO's Transmission types

Transmission Type	Receive PDOs	
0 [2)] **1 to 240** [2)]	The data is passed to application after the SYNC – ignoring transmission type count. (DS301- Version: 4.0.2- 9.2.1.1 Transmission Modes)	Synchronous
241 to 251 [3)]	- - - - -	
252	Not valid [1)]	
253	Not valid [1)]	
254, 255	The data of asynchronous RPDOs is passed directly to the application	Asynchronous

Figure 40 - RPDOs Transmission types

*1) Receive PDOs do not support RTR only transmission type.
*2) if PDO Transmission type is less than 241, Event timer is not started.
*3) reserved
**) Transmission is base on NMT state (see Communication Object table)
*4) SYNC but acyclic (not periodically), only if an event occurred before the SYNC

6.1.1.Synchronous transmission

The PDO can only be triggered by the Network SYNC message; this can be done in different ways.

6.1.2.Acyclic:

Remote request from another device 'pre-triggers' the PDO. A device (profile) specific event 'pre-triggers' the PDO.

6.1.3.Cyclic Synchronous

In transmission types 1-240 the PDO is transmitted cyclically: after every "n-th" SYNC (n = 1...240).

6.1.4.Asynchronous

The PDO can be triggered by:
Remote request from another device
Device (profile) specific event

6.1.5.Polled (RTR)

The PDOs can also be polled by data request telegrams (remote frames). In this way it is possible to get the input process image of event-driven inputs onto the bus, even when the inputs have not changed. RTRs are only valid for transmit PDOs and if device support heartbeat.

6.1.6.Event driven:

The "event" is the alteration of an input value, the data being transmitted immediately after this change. The event driven flow can make optimal use of the bus bandwidth, since instead of the whole process image it is only the changes in it that are transmitted. A short reaction time is achieved at the same time, since when an input value changes it is not necessary to wait for the next interrogation from a master.

The following define the valid transmission types for transmit and receive PDOs:

6.2. PDO Timers

6.2.1. Inhibit time

Can be set for asynchronous PDOs. The value is to be given as a multiple of 100 microseconds (0.0001 seconds). The inhibit time is useful when frequent uncontrolled alterations of input values occur. If the inhibit time is configured, the node may not transmit the relevant PDO again before expiry of the inhibit time; this ensures that there is no inadmissibly high busload. The inhibit time is only used for TxPDOs. It has no significance for RxPDOs

6.2.2. Event time

Asynchronous TxPDOs can be transmitted cyclically with the event timer, subindex 5. If its value is greater than 0, it becomes a millisecond timer. When this is expired, the PDO is transmitted. Transmission therefore takes place both when an external device input is altered and when the event timer is lapsed. This subindex is also only significant for transmit-PDOs.

6.3. The PDO mapping procedure

Tritex supports variable mapping; Variable PDO mapping describes that the mapping entries of a PDO can be changed only in NMT pre-operational mode and the mapping entries can only be changed by following CiA301 specification (version 4.0.2).:

The following procedure is used for re-mapping, which may take place during the NMT state Pre-operational and during the NMT state Operational, if supported:

1. Destroy PDO by setting bit *valid* to 1b of sub-index 01h of the according PDO communication parameter.
2. Disable mapping by setting sub-index 00h to 00h.
3. Modify mapping by changing the values of the corresponding sub-indices.
4. Enable mapping by setting sub-index 00h to the number of mapped objects.
5. Create RPDO by setting bit *valid* to 0b of sub-index 01h of the according PDO communication parameter.

6.4. Receive PDO

Four Receive PDO's (RPDO) are available. Their parameters are defined in the objects 1400h, 1401h, 1402h and 1403h. The assignment of the data to be received is defined in the mapping objects 1600h, 1601h, 1602h and 1603h.

6.4.1. Receive PDO Communication Parameter

6.4.1.1. Objects 1400,1401,1402,1403

Index	Name	Object Type	Category
1400,1401,1402,1403	Receive PDO Parameter	Record	Mandatory

Sub-index	Description	Entry Category	Access	PDO Mapping	NVM [1]	Value Range	Default
0	Number of entries	Mandatory	RO	NO	NO	UNSIGNED8	2
1	COB-ID	Mandatory	RW	NO	YES	UNSIGNED32	1400h: 200h + Node-ID 1401h: 300h + Node-ID 1402h: 400h + Node-ID 1403h: 500h + Node-ID
2	Transmission Type [3]	Mandatory	RW	NO	YES	UNSIGNED8	FE h

A unique COB-ID (unique with respect to the entire CANopen network, not just the node) must be assigned to each PDO which will be used over the CAN network. It is recommended using the Predefined Connection Set where ever possible. It is the system designer's responsibility to ensure that all PDOs have a unique COB-ID. It is best to assign the COB-IDs in a logical order, with the most important PDOs assigned to the lowest COB-IDs.

Sub-Index	Bit	Description
0	0 to 7	This sub-index o contains the number of valid entries within the communication record.
1	0 to 10	COB-ID
	11 to 28	Always 0
	29	Always 0 = 11bit ID (CAN2.0A)
	30	0: RTR allowed on this PDO 1: RTR is Not allowed on this PDO
	31	0: PDO Exists / Is Valid 1: PDO Does not Exist / Is Not Valid
2	0 to 7	Description of Transmission Type [2] [3] [5]

Transmission Type	PDO Transmission				
	Cyclic	Acyclic	synchronous	asynchronous	RTR only
0		Valid	valid		
1 to 240	valid		valid		
241 to 251	reserved				
252	Not Used for Received PDO's				
253					
254, 255				valid	

*1) see object Store Parameters 1010h
*2) Transmission Type
*3) Changes valid only in PRE_OPERATIONAL state
*4) not used
*5) It is not allowed to change the value while the PDO exists (Bit 31) of sub-index 1 is 0) in the state PRE_ OPERATIONAL
*6) not used

6.4.2.Receive PDO Mapping Parameter

Index	Name	Object Type	Category
1600,1601,1602,1603	Receive PDO Mapping Parameter	Record	Mandatory

6.4.2.1. RPDO-1 (1600)

Sub-index	Description	Entry Category	Access	PDO Mapping	NVM 1)	Value Range	Default
0	Number of Mapped Objects	Mandatory	RW	NO	YES	0: deactivate 2) 1-8: active	2
1	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	6040 0010h 3)
2	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	6060 0008h 4)
3- 8	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	0000 0000

*3) Controlword: Index 6040h, Sub-Index 0, 16 bits in length
*4) Modes of Operation: Index 6060h, Sub-Index 0, 8 bits in length

6.4.2.2. RPDO-2 (1601)

Sub-index	Description	Entry Category	Access	PDO Mapping	NVM 1)	Value Range	Default
0	Number of Mapped Objects	Mandatory	RW	NO	YES	0: deactivate 2) 1-8: active	2
1	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	6040 0010h 5)
2	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	6071 0010h 6
3- 8	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	0000 0000

*5) Controlword: Index 6040h, Sub-Index 0, 16 bits in length
*6) Target Torque (Mode TQ): Index 6071h, Sub-Index 0, 16 bits in length

6.4.2.3. RPDO-3 (1602)

Sub-index	Description	Entry Category	Access	PDO Mapping	NVM 1)	Value Range	Default
0	Number of Mapped Objects	Mandatory	RW	NO	YES	0: deactivate 2) 1-8: active	2
1	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	6040 0010h 7)
2	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	607A 0020h 8)
3- 8	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	0000 0000

*7) Controlword: Index 6040h, Sub-Index 0, 16 bits in length
*8) Target Position (Mode PP): Index 607Ah, Sub-Index 0, 32 bits in length

6.4.2.4. RPDO-4 (1603)

Sub-index	Description	Entry Category	Access	PDO Mapping	NVM [1)]	Value Range	Default
0	Number of Mapped Objects	Mandatory	RW	NO	YES	0: deactivate [2)] 1-8: active	2
1	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	6040 0010h [9)]
2	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	60FF 0020h [10)]
3- 8	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	0000 0000

*9) Controlword: Index 6040h, Sub-Index 0, 16 bits in length
*10) Target Velocity (Mode PV): Index 60FFh, Sub-Index 0, 32 bits in length

6.5. Transmit PDO

Four Transmit PDO's (RPDO) are available. Their parameters are defined in the objects 1800h, 1801h, 1802h and 1803h. The assignment of the data to be received is defined in the mapping objects 1A00h, 1A01h, 1A02h and 1A03h.

6.5.1. Transmit PDO Communication Parameter

6.5.1.1. Objects 1800,1801,1802,1803

Index	Name	Object Type	Category
1800,1801,1802,1803	Transmit PDO Parameter	Record	Mandatory

Sub-index	Description	Entry Category	Access	PDO Mapping	NVM [1)]	Value Range	Default
0	Number of entries	Mandatory	RO	NO	NO	UNSIGNED8	5
1	COB-ID	Mandatory	RW	NO	NO	UNSIGNED32	1800h: 200h + Node-ID 1801h: 300h + Node-ID 1802h: 400h + Node-ID 1803h: 500h + Node-ID
2	Transmission Type [3) 6)]	Mandatory	RW	NO	YES	UNSIGNED8	FE h
3	Inhibit Time [2) 5)]	Optional	RW	NO	YES	UNSIGNED16	0
5	Event Time [2) 4)]	Optional	RW	NO	YES	UNSIGNED16	0

A unique COB-ID (unique with respect to the entire CANopen network, not just the node) must be assigned to each PDO which will be used over the CAN network. It is recommended using the Predefined Connection Set where ever possible. It is the system designer's responsibility to ensure that all PDOs have a unique COB-ID. It is best to assign the COB-IDs in a logical order, with the most important PDOs assigned to the lowest COB-IDs.

Sub-Index	Bit	Description
0	0 to 7	This sub-index o contains the number of valid entries within the communication record.
1	0 to 10	COB-ID
	11 to 28	Always 0
	29	Always 0 = 11 bit ID (CAN2.0A)
	30	0: RTR allowed on this PDO 1: RTR is Not allowed on this PDO
	31	0: PDO Exists / IS Valid 1: PDO Does Not Exist / Is Not valid
2	0 to 7	Description of Transmission Type [2) 3)]

		Transmission Type / PDO Transmission: Cyclic / acyclic / synchronous / asynchronous / RTR only 0: — / valid / valid / — / — 1 to 240: valid / — / valid / — / — 241 to 251: reserved 252: — / — / valid / — / valid 253: — / — / — / valid / valid 254: — / — / — / valid / — 255: — / — / — / valid / —
3	0 to 16	Inhibit Time [2) 3) 5)]
5	0 to 16	Event Time [2) 3) 4)]

*1) see object Store Parameters 1010h
*2) Transmission Type
*3) changes valid only in PRE_OPERATIONAL State
*4) Value will be ignored if Transmission Type less than 241 (SYNC enabled)
*5) It is not allowed to change the value while the PDO exists (Bit 31) of sub-index 1 is 0) in the state PRE_ OPERATIONAL

6.5.2. Transmit PDO Mapping Parameter

Index	Name	Object Type	Category
1A00,1A01,1A02,1A03	Transmit PDO Mapping Parameter	Record	Mandatory

6.5.2.1. TPDO-1 (1A00)

Sub-index	Description	Entry Category	Access	PDO Mapping	NVM [1)]	Value Range	Default
0	Number of Mapped Objects	Mandatory	RW	NO	YES	0: deactivate [2)] 1-8: active	2
1	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	6041 0010h [3)]
2	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	6061 0008h [4)]
3 - 8	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	0000 0000

*3) Statusword: Index 6041h, Sub-Index 0, 16 bits in length
*4) Actual Modes of Display: Index 6061h, Sub-Index 0, 8 bits in length

6.5.2.2. TPDO-2 (1A01)

Sub-index	Description	Entry Category	Access	PDO Mapping	NVM [1)]	Value Range	Default
0	Number of Mapped Objects	Mandatory	RW	NO	YES	0: deactivate [2)] 1-8: active	2
1	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	6041 0010h [5)]
2	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	6077 0010h [6]
3 - 8	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	0000 0000

*5) Statusword: Index 6041h, Sub-Index 0, 16 bits in length
*6) Torque actual value: Index 6077h, Sub-Index 0, 16 bits in length

6.5.2.3. TPDO-3 (1A02)

Sub-index	Description	Entry Category	Access	PDO Mapping	NVM [1)]	Value Range	Default
0	Number of Mapped Objects	Mandatory	RW	NO	YES	0: deactivate [2)] 1-8: active	2
1	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	6041 0010h [7)]
2	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	6064 0020h [8)]
3 - 8	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	0000 0000

*7) Statusword: Index 6041h, Sub-Index 0, 16 bits in length
*8) Position Actual: Index 6064h, Sub-Index 0, 32 bits in length

6.5.2.4. TPDO-4 (1A03)

Sub-index	Description	Entry Category	Access	PDO Mapping	NVM [1)]	Value Range	Default
0	Number of Mapped Objects	Mandatory	RW	NO	YES	0: deactivate [2)] 1-8: active	2
1	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	6041 0010h [9)]
2	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	606C 0020h [10)]
3 - 8	PDO Mapping	Conditional	RW	NO	YES	UNSIGNED32	0000 0000

*9) Statusword: Index 6041h, Sub-Index 0, 16 bits in length
*10) Velocity actual value: Index 606Ch, Sub-Index 0, 32 bits in length

*1) see object Store Parameters 1010h
*2) Defines the number of mapped Objects (default and maximum of 8)

7. Manufacture Objects

Index	sub index	Description	
2000	0	CANopen ID	
2001	0	CANopen Baud Rate	
2002	-	System Read	
2003	-	System Write	
2100	-	Control Access	
2101	-	Control Access Actual	
2109	0	Pac Command	
2110	-	Status CANopen	

Figure 41 - Dictionary of Manufacture Objects

7.1. System Read (2002)

System Read is an object that allows CANopen to internal registers.

Index	sub index	Description	Type	Default	Attr	NVM
2002	-	System Read	-		-	
	0	Number of Entries	UINT8	2	RO	
	1	GID	UINT32	GID_CANOPEN_PARAMETER_BAUD	RW	
	2	Data			RO	

Figure 42 - System Read object

To use the System write register the user must perform the following steps

(1) Write Global ID to 2002.1
(2) Read data at 2002.2

Refer to Tritex CANopen Getting Started for examples in using this object.

7.2. System Write (2003)

System Write is an object that allows CANopen to internal registers.

Index	sub index	Description	Type	Default	Attr	NVM
2003	-	System Write	-		-	
	0	Number of Entries	UINT8	2	RO	
	1	GID	UINT32	GID_CANOPEN_PARAMETER_BAUD	RW	
	2	Data			RW	

Figure 43 - System Write object

To use the System write register the user must perform the following steps

(1) Write GID to 2003.1
(2) Write data to 2003.2

Refer to Tritex CANopen Getting Started for examples in using this object.

7.3. Drive Command Access (2109)

Refer to Tritex CANopen Getting Started for examples in using this object.

Index	sub index	Description	Type	Default	Attr	NVM
2109	0	Drive Command Access using PACs	GID	0	WO	

Figure 44 - System Write object

8. Motion

The device profile for drives and motion control defines the functional behavior of controllers for servo drives, frequency inverters and stepper motors. The specification includes a finite state automaton (FSA). The state of the drive determines which commands are accepted and if high power is applied. States are changed by a *control-word* received from the host-controller can be initiated by internal events. The current state is indicated by the *Statusword*.

8.1. State Machine

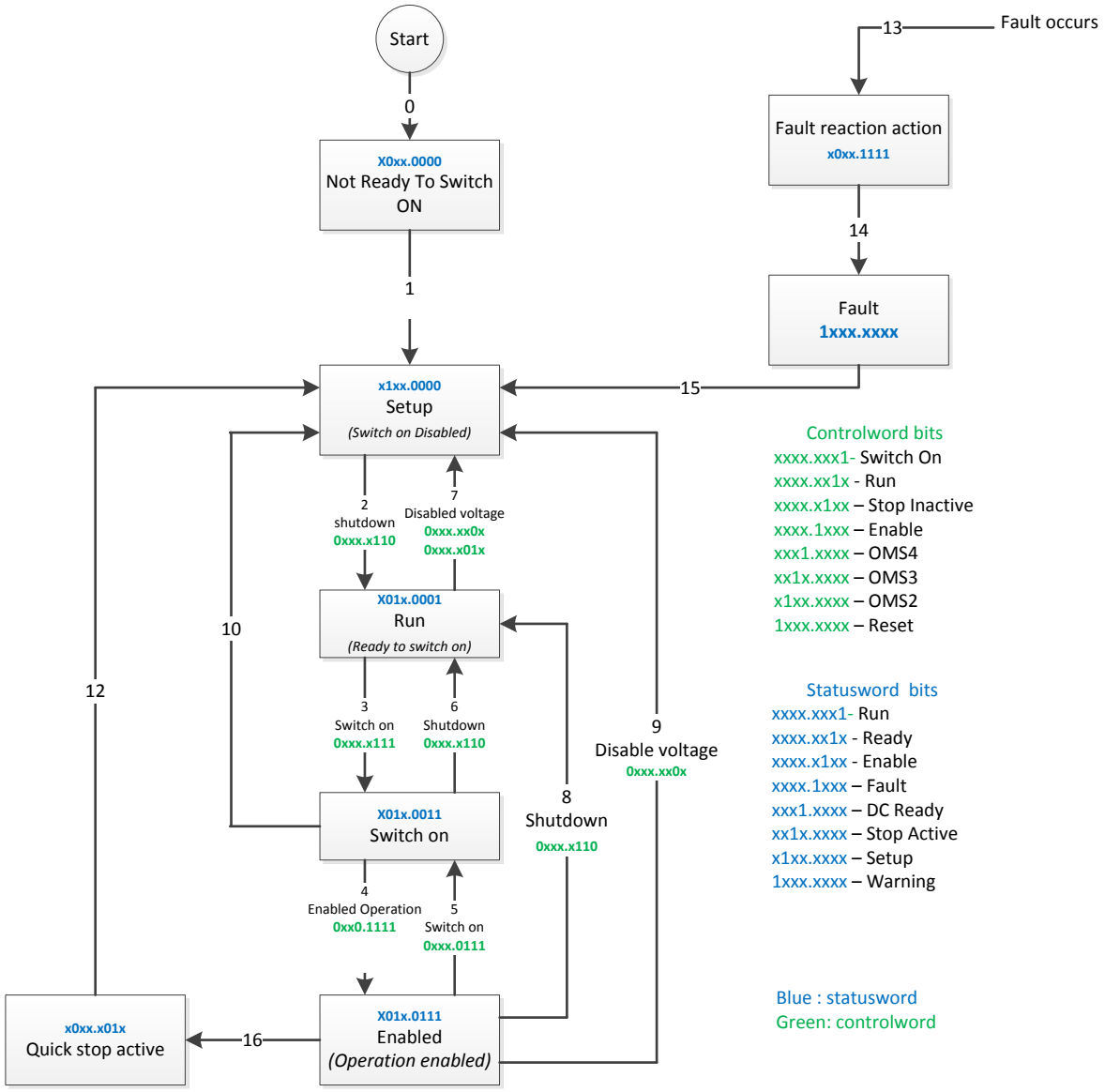


Figure 45 - State machine

NOTE! Steps could be bypass automatically at startup – reference Control Enable Actions for more information.

Transition	Events	Actions
0	Automatic transition after power-on or reset application	Drive device self-test and/or self initialization shall be performed.
1	Automatic transition Test If Options: A. auto switch-on and run at startup is enable go to "Switch On" state B. Auto-Enable on Start up – enable operational enable	Communication shall be activated.
2		
3		
4		
5		
6		
7		
8		
9		
10		
11		
12		
13		
14		
15	Fault reset command from control device	A reset of the fault condition is carried out, if no fault exists currently on the drive device; after leaving the Fault state, the Fault reset bit in the controlword shall be cleared by the control device. *(else if option Auto Reset Fault Off , clearing of Fault reset is not needed)*
16		

Figure 46 - Transition events and actions

Not Ready (*DS402: Not Ready to Switch On)* [(d)]

- Low level (logic) power has been applied to the drive.
- The drive is being initialized.
- The drive is not (and cannot be) enabled.

Setup *(DS402: Switch On Disabled)* [(d)]

- Drive Initialization is complete.
- The drive parameters have been set up
- Drive parameters may be changed.
- High (mains) voltage may not be applied to the drive, (e.g. for safety reasons).
- The drive is not (and cannot be) enabled.

Run (DS402: Ready to Switch On) [(d)]

- The drive parameters may be changed.
- The drive is not (and cannot be) enabled.

Ready (DS402: Switch On) [(d)]

- The drive parameters may be changed.
- The drive is not enabled.

Enabled (DS402: Operational Enabled) [(d)]

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

(This corresponds to normal operation of the drive.)

Stop Active (DS402: Quick Stop Active) [(d)]

- The Quick Stop function is being executed.
- The drive function is enabled and power is applied to the motor.
- The drive parameters may be changed.

Fault Reaction Active (DS402: Fault Reaction Active) [(d)]

- The drive parameters may be changed.
- A non-fatal fault has occurred in the drive.
- The Quick Stop function is being executed.

Fault (DS402: Fault) [(d)]

- The drive parameters may be changed.
- A fault has occurred in the drive.
- The drive is not (and cannot be) enabled.

8.2. Control

User configurable parameters refer to parameters for limits.

Description	Type	Norm	Acc.	NVM	GID	Modbus	CAN
Control Group							
controlword	UINT16		RW		PACs		6040.0[(M)]
Modes of operation	INT16[(b)]		RW		SYSTEM.MOTION.COMMAND. REQUESTEDMODE		6060.0[(M)]
Stop Method	INT16		RW		SYSTEM.CONTROL.COMMAND. STOPACTION		605A.0
Fault Reaction code	INT16		RW		SYSTEM.CONTROL.COMMAND. FAULTACTION		605E.0
Stop Ramp	UINT32	mrps/s	RW		SYSTEM.CONTROL.COMMAND.STOPDECEL		6085.0

Figure 47 - Control Group Register (Run time)

Initialize

Description	Type	Norm	Access	NVM	GID	Modbus	CAN
Options	UINT16		RW	*	SYSTEM.CONTROL.USERPARAM.OPTIONS		3100.0
Fault Action	INT16		RW	*	SYSTEM.CONTROL.USERPARAM.FAULTACTION		3101.0
Stop Action	INT16		RW	*	SYSTEM.CONTROL.USERPARAM.STOPACTION		3102.0
Enable Action	UINT16		RW	*	SYSTEM.CONTROL.USERPARAM.ENABLEACTION		4001.0
Stop Ramp	UINT32	mrps/s	RW	*	SYSTEM.CONTROL.USERPARAM.STOPDECEL		3103.0

Figure 48 – Control Group registers (Initialize)

8.2.1. Controlword (6040.0)

The controlword has a dual purpose, controlword the state machine of the drive and command the motion mode.

15 12	11	10	9	8	7	6 5 4	3	2	1	0
reserved	Break release	reserved	Mode specific	Halt	Fault reset	Mode specific	Enable	Quick Stop	Run	Switch on

MSB LSB

Figure 49 - Controlword

Drive Control bits							
Controlword							
15 -7	6	5	4	3	2	1	0
x … x	**Reset fault**	**x**	**x**	**Enable**	**Stop**	**Run**	**Switch on**

Figure 50 - Drive control bits

Bits	Drive Control	*DS402*[(d)]	Description
0	**Switch on**	Switch on	interlock

1	**Run**	*Enable voltage*	The DS402 FSA 'ready to switch on' state is waiting for the drive to be set to enable high level power. Since the drive doesn't control its own bus power, this state place for controllers that require an extra command interlock before accepting the ENABLE bit. Controllers that don't want the extra interlock may elect to force this control bit set at start-up. Refer to GID.SYSTEM.CONTROL.USERPARAM.OPTIONS
2	**Stop**	*Quick stop*	Command the drive to deactivate, base on Stop option To deactivate stop controlword bit 2 must be equal to 1. Refer to *GID. SYSTEM.CONTROL.USERPARAM.STOPACTION* (0x605A.0).
3	**Enable**	*Enable operation*	Commands drive into operational enable state.
6	**Reset Faults**	Fault reset	Reset faults on the rising edge.

Figure 51 - Drive control bits

Mode of Operation	Operation mode specific bits				
	9	8	6	5	4
Profile position (pp)	Change on Set point	Halt	Abs/rel	Change set Immediately	New set
Profile velocity (pv)	Reserved	Halt	Reserved	Reserved	Reserved
Profile torque (pt)	Reserved	Halt	Reserved	Reserved	Reserved
Homing (hm)		Halt			
Profile jog (pj)		Halt			

Figure 52 - Controlword, Mode specific bits

8.2.2.Mode of Operation (6060.0)

Drive Mode *(Mode of Operation)* [(d)] the function of the product depends on the selected modes of operation. It is not possible to operate the modes in parallel. The user must select a mode to operate in.

Supported modes are:

Value	Definition
-8	Internal mode - Stop
-7	Internal mode - Lock Voltage
-6	Internal mode - Lock Current
-5	Internal mode - PreHeat
-4	Index mode (im) indexes 0 – xx *(future)*
- 3	analog position mode *(future)*
- 2	Jog
-1	Hold Position
0	No mode
+1	Profile position (pp)
+3	Profile velocity (pv)
+4	Profile Torque (tq)
+6	Homing mode (hm)
+7	interpolated position mode (ip) *(future)*

Figure 53 - Mode Options

- The product allows the user to switch dynamically from operation mode to operation mode.

- The mode of operation is set or read using Mode of Operation (6060h) and Mode of Operation display (6061h).
- 6061h will always show current active mode state, when Enable operation is inactive current mode is no – mode.
- New mode is loaded from 6060h on the rising edge of operation mode bit.

8.2.3.Fault reaction (605E.0)

Indicates what action is performed when fault is detected in the PDS. The slow down ramp is the deceleration value of the used mode of operations.

Value	Definition	Description
0	Disable drive function	Immediately disable the drive, Motor is free to rotate (Drive status = switch on Disable)
+1	Slow down on slow down ramp	Use the active mode deceleration, if mode has no deceleration rate define the quick stop ramp is used. (Drive status = switch on Disable)
+2	Slow down on quick stop ramp	Use the active mode deceleration, then disable the drive (Drive status = switch on Disable)
+3	Slow down on current limit	Not supported (same as Disable drive function)
+4	Slow down on voltage limit	Not supported (same as Disable drive function)

Figure 54 - Fault reaction

NOTE!

Priority faults are events that the drive has detected and will shut download immediately without using Fault reaction configuration.

Events:

- Current exceed factory Fault Trip Current (SYSTEM.BUS.FACTORYPARAM.IFAULT)
- Voltage exceed High Voltage Trip (SYSTEM.BUS.FACTORYPARAM.VFAULT)
- When Enable and Voltage is below Low Voltage Trip (SYSTEM.BUS.FACTORYPARAM.VLOW)

Action : Break on(if available), bus Power off, set fault flag
State: DS402 - Fault Active State

Drive Trip setting are available on Exlar Expert software page 'Factory Parameters/Limits'

8.2.4.Stop methods (605A.0)

Stop methods *(Quick Stop Option Code)* [d] indicates what action is performed when the Deactivated stop function is executed. The slow down ramp is the deceleration value of the used mode of operations. *(DS402: Quick stop methods)*

Value	Stop Methods	Definition	Description
0	Disable Drive	Disable drive function	Immediately disable the drive 1. Active Mode = None (0) 2. PWM Disable 3. Enable Break 4. State = Run *(DS402: Ready to Switch On)*
+1	Slow Down Ramp -> Setup	Slow down on slow down ramp and transit into Switch On Disabled	Decelerate using motion mode's Deceleration (if available, else stop decelerate) and then disable drive
+2	Quick Stop Ramp -> Setup	Slow down on quick stop ramp and transit into Switch On Disabled	Decelerate using stop Deceleration then disable the drive
+3		Slow down on current limit and transit into Switch On Disabled	unimplemented - same as DISABLE (0)
+4		Slow down on voltage limit and transit into Switch On Disabled	unimplemented - same as DISABLE (0)
+5	Slow Down Ramp ->Stop Active	Slow down on slow down ramp and stay in Quick Stop Active	Decelerate using motion mode's Deceleration (if available, else stop Decelerate) and remain in Quick Stop Active
+6	Quick Stop Ramp -> Stop Active	Slow down on quick stop ramp and stay in Quick Stop Active	Decelerate using stop Deceleration rate and remain in Quick Stop Active
+7		Slow down on current limit and stay in Quick Stop Active	unimplemented - same as DISABLE (0)
+8		Slow down on voltage limit and stay in Quick Stop Active	unimplemented - same as DISABLE (0)

Figure 55 - Stop methods

8.2.5.Enable Actions

The following are System Setup Options, internally updated only after new power cycle.

SYSTEM.CONTROL.USERPARAM.OPTIONS[1]		
Value	Name	Description
0x0001	Auto Run	Sets RUN bit 2 *(Enable voltage)* [d] in controlword at startup.
0x0002	Auto Switch On	Sets SWITCH On bit 1 *(Enable voltage)* [d] in controlword at startup.
0x0004	Auto Enable	Sets ENABLE bit 3 *(Operational Enable)* [d] in controlword at startup.
0x0008	Auto Fault Reset Off	Automatically resets the Fault Reset bit in controlword after executing the fault reset action or if no faults are pending so that the 'Fault Reset Off' command is not required. This simplifies the interface to external controllers, requiring only the Fault On command.
0x2000	Requires Thermal Enable	Requires that THERMAL.WARNINGS.TEMPERATURE is inactive before the READY state can become active.
0x4000	Requires Current Enable	Requires that BUS.WARNINGS.IC is inactive before the READY state can become

		active.
0x8000	Requires Voltage Enable	Requires that BUS.WARNINGS.VOLTAGE is inactive before the READY state can become active.

Figure 56 - System Setup Options

(1) Found on the Tritex Expert System Setup option page

8.3. Monitor

Description	Type	Norm	Acc.	GID	Modbus	CANopen
Control Status Group						
status word[2]	UINT16	--	RO	SYSTEM.CONTROL.EVENTS.STATUS	403	6041.0[(M)]
Modes of operation display[1]	INT16[(b)]	--	RO	SYSTEM.MOTION.MONITOR. ACTIVEMODE	2048	6061.0[(M)]
Position[1]	UINT32	0.0001 rev	RO	SYSTEM.POST.MONITOR.POSITION	762	6064.0[(M)]
Velocity[1]	UINT32	mv	RO	SYSTEM.POST.MONITOR.VELOCITY	764	606C.0[(M)]
Current	UINT16	0.001 rated	RO	SYSTEM.BUS.MONITOR.CURRENT	586	6078.0[(M)]
Continuous Current[1]	UINT16	0.001 rated	RO	SYSTEM.BUS.MONITOR. CONTINUOUSCURRENT	587	- - - -
Power	UINT16	0.1 W	RO	SYSTEM.BUS.MONITOR. POWER	588	- - - -
Continuous Power[1]	UINT16	0.1 W	RO	SYSTEM.BUS.MONITOR.CONTINUOUS POWER	589	- - - -
Board Temperature[3]	UINT16	0.1 DEG C	RO	SYSTEM.THERMAL.MONITOR.PCB	1428	- - - -
Actuator Temperature[3]	UINT16	0.1 DEG C	RO	SYSTEM.THERMAL.MONITOR.MOTOR	1430	- - - -
Heat sink Temperature[3]	UINT16	0.1 DEG C	RO	SYSTEM.THERMAL.MONITOR. HEATSINK	1429	- - - -

Figure 57 - Control Group Status Registers

NOTE! Using the Exlar Tritex Drive software the above items can be found on
[1] "Status " Page
[2] "Status Drive" Page
[3] "Diagnostic" Page

8.3.1.Statusword (6041.0)

The Statusword provide the status of the PDS FSA.

Drive Status										
Statusword										
15 -10	9	8	7	6	5	4	3	2	1	0
x … x	**Remote**	**Homed**	**x**	**Setup**	**Stop Active**	**DC Bus Ready**	**Fault**	**Enabled**	**Ready**	**Run**

Figure 58 - Statusword

Bits	Drive control bits		Description
	Exlar	*DS402*[(a)]	
0	**Run**	*Ready to run*	Drive is in normal runtime operation mode and is ready

			to accept the power command
1	**Ready**	*Switch on*	
2	**Enabled**	*Operational enabled*	The drive is enabled and ready to command motion.
3	**Fault**	Fault active	Fault has occurred in the system and fault reaction has completed
4	**DC Bus Ready**	*Voltage enable*	Tritex voltage is greater than “Factory voltage low trip”. Refer to Tritex installation guide for more information. (Reference System.Bus.FactoryParam.Vlow,636)
5	**Stop Active**	*Quick stop actived*	Indicates the PDS is reacting to deactivate request. The final state is determined by Quick Stop option register. 0= Quick Stop Active 1= Inactive,
6	**Setup**	Switch on disabled	Drive is in SETUP mode and not ready for operation. Some commands are available only in SETUP mode. The SETUP bit is a 'convenience' event bit and is always the inverse of the RUN bit
8	**Homed**	*Manufacture defined*	Drive is homed
9	**Remote**	*Remote*	0 = indicate that the controlword is not processed 1 = indicate that the controlword is processed

Figure 59 - Drive status bits

Mode of Operation	Mode specific bits		
	13	12	10
	oms	oms	tr
Profile position (pp)	Following error	Set-point acknowledge	Final target reached
Profile velocity (pv)	x	x	Target velocity reached
Profile torque (pt)	x	x	Target torque reached
Homing (hm)	See Homing Mode		
Profile jog (pj)	See Jog Mode		

Figure 60 - Statusword, Mode specific bits

PDS 402 State[d]	Exlar Internal State	Bits in status word						
		6	5	3	2	1	0	
		setup	Stop active	fault	enable	ready	run	
		(sod)	*(qs)*	*(f)*	*(oe)*	*(so)*	*(rtso)*	*DS 402: Bit definition*
Not Ready to Switch On	**Not Ready**	0	X	0	0	0	0	
Switch On Disabled	**Setup**	1	X	X	0	0	0	
Ready to Switch On	**Run**	0	1	0	0	0	1	
Switch On	**Ready**	0	1	0	0	1	1	
Operational Enabled	**Enabled**	0	1	0	1	1	1	
Quick Stop Active	**Stop Active**	0	0	0	1	1	1	
Fault Reaction Active	**Fault Reaction Active**	0	X	1	1	1	1	
Fault	**Fault**	0	X	1	0	0	0	

Figure 61 – DS402 state machine states

8.3.2.Mode of operation (6061.0)

Object provides the actual active operation mode. Refer to Mode of Operation for support modes.

8.3.3.Position (6064.0)

This object represents the actual value of the position measurement

8.3.4.Velocity (606C.0)

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

8.3.5.Current (6078.0)

Actual value of the current, it corresponds to the current in the motor. The value shall be given per thousand of rated current.

8.3.6.Power

The output power [0.1 W] is simply calculated as torque [NM] x speed [rad/s] using the pre-calculated power scale factor. power [0.1W] = i [mRated] * velocity [mrps] * power_factor [0.1W/mRated-mrps]

8.3.7.Temperature

Display current Temperature

8.4. Motion Setup

Description	Type	Norm	Acc.	NVM	GID	Modbus	CAN
Stop Ramp[1]	UINT32	mrps/s	RW	*	SYSTEM.MOTION.USERPARAM. STOPDECEL		6085.0
Max Velocity[1]	UINT32	mrps	RW	*	SYSTEM.MOTION.USERPARAM. VMAX		607F.0
Motion profile type	UINT16		RW		SYSTEM.MOTION.COMMAND. PROFILE.TYPE		6086.0
Profile acceleration[2]	UINT32	mrps/s	RW	*	SYSTEM.MOTION.COMMAND. PROFILE.ACCEL		6083.0
Profile deceleration[2]	UINT32	mrps/s	RW	*	SYSTEM.MOTION.COMMAND. PROFILE.DECEL		6084.0
Max Following Error[1]	UINT32	0.0001 rev	RW	*	SYSTEM.MOTION.USERPARAM. WINDOWS.PERROR		6065.0
In Position Width[1]	UINT32	0.0001 rev	RW	*	SYSTEM.MOTION.USERPARAM. WINDOWS.PERROR		6067.0
Following Error Time Limit[1]	UINT16	ms	RW	*	SYSTEM.MOTION.USERPARAM. WINDOWS.PERRORTIME		6066.0
Time to Active in Position[1]	UINT16	ms	RW	*	SYSTEM.MOTION.USERPARAM. WINDOWS.PZEROTIME		6068.0
Velocity Window width[1]	UINT16	mrps	RW	*	SYSTEM.MOTION.USERPARAM. VTARGETWINDOW		606D.0
Velocity threshold[1]	UINT16	mrps	RW	*	SYSTEM.MOTION.USERPARAM. VZEROWINDOW		606F.0
Velocity window time[1]	UINT16	ms	RW	*	SYSTEM.MOTION.USERPARAM. WINDOWS.VTARGETTIME		606E.0
Velocity threshold time[1]	UINT16	ms	RW	*	SYSTEM.MOTION.USERPARAM. WINDOWS.VZEROTIME		6070.0
Target Current Window[1]	UINT16	0.001 rated	RW	*	SYSTEM.MOTION.USERPARAM. WINDOWS.ITARGET		4002.0
In Current Limit Time[1]	UINT16	ms	RW	*	SYSTEM.MOTION.USERPARAM. WINDOWS.ITARGETTIME		4003.0
Maximum rated current[3]	UINT32	mA	R-	*	SYSTEM.BUS.FACTORYPARAM.IRATED		6075.0
Max Current[1]	UINT16	0.001 rated	RW	*	SYSTEM.MOTION.USERPARAM. LIMITS.CURRENT		6073.0

Figure 62 - Motion Group (run time)

NOTE! Using the Exlar Tritex Drive software the above items can be found on the
[1] "Setup/System Setup/Motion Setup " Page
[2] "Motion/Profile Velocity" Page
[3] "Setup/Factory Parameters/Limits" Page

8.4.1. Quick stop deceleration (6085.0)

Stop Ramp *(Quick Stop Deceleration)* [(d)] indicates the configured deceleration used to stop the motor when the quick stop function is activated and the quick stop code object (605Ah) is set to 2 or 6. The quick stop deceleration is also used if the fault reaction code object (605Eh) is 2.

8.4.2. Max Profile Velocity (607F.0)

Max Velocity *(Max Profile Velocity)* [(d)] indicates the configured maximal allowed velocity in either direction during a profile position mode

8.4.3. Motion Profile type (6086.0)

Profile type used for velocity command generation by both the position (pp) and profile velocity (pv) motion modes.

Value	Definition	Description
0	LINEAR	Linear (trapezoidal) ramping
1	SINE	Sinusoidal (haversine) ramping (Not Supported)
2	JERK_FREE	No jerk limiting (Not Supported)
3	JERK_LIMITED	No jerk limiting (Not Supported)

Figure 63 - Motion Profile option

8.4.4. Profile acceleration (6083.0)

The DS402 defines Profile acceleration to indicate the acceleration rate for both the Profile position (PP) and Profile velocity (PV) mode.

NOTE! While in Position mode and the Independent data option is enabled, Position mode will use an autonomous acceleration value. Refer to Profile Position mode Options for more information.

8.4.5. Profile deceleration (6084.0)

The DS402 defines Profile deceleration to indicate the deceleration rate of the motion mode used for Profile position (PP) and Profile velocity (PV) mode.

NOTE! While in Position mode and the Independent data option is enabled, Position mode will use an autonomous deceleration value. Refer to Profile Position mode Options for more information.

8.4.6. Follow Error Window (6065.0)

Max Following Error (*Follow Error Window*) [(d)] Indicate the configured range of tolerated position values symmetrically to the position demand value. If the position actual value is out of the following error window, a

following error occurs. A following error may occur when a drive is blocked, unreachable profile le velocity occurs, or at wrong closed-loop coefficients. Following Error Time out

8.4.7. In Position window (6067.0)

In Position Width (*In Position window)* [(d)] defines the configured symmetrical range of accepted positions relatively to the target position. If the actual value of the position encoder is within the position window, this target position shall be regarded as reached. The target position shall be handled in the same manner as in the profile generator concerning limiting functions and transformation into internal machine units before it may be used with this function.

8.4.8. Following Error Time out (6066.0)

Following Error Time Limit (*Following Error Time out)* [(d)] indicates the configured time for a following error condition, after that the bit 13 of the statusword will be set to 1.

8.4.9. In Position window time (6068.0)

Time to Achieve In position *(In Position Window Time)* [(d)] indicates the configured time, during which the actual position within the position window is measured. The value is given in milliseconds.

The time required for position error to remain outside of the p_error_window

8.4.10. Velocity window (606D.0)

Velocity Window Width *(Velocity Window)* [(d)] indicate the configured velocity window, determines whether feedback is within target window.

8.4.11. Velocity Threshold (606F.0)

Determines the feedback value to indicate within zero window.

8.4.12. Velocity window time (606E.0)

Sets the hysteresis time for the in velocity window status event; the time is specified in milliseconds

8.4.13. Velocity threshold time (6070.0)

When the actual velocity is within the position window during the defined position window time which is given in multiples of milliseconds

8.4.14. Current Window (4002.0)

Sets the hysteresis window for the current window status events; units of measure is MRATED

8.4.15. Current Window time (4003.0)

Sets the hysteresis time for the in current window status event; the time is specified in milliseconds

8.4.16. Maximum Current (6073.0)

User configurable [1] **Peak Current** is the configured maximum permissible torque creating current in the motor. The value is given per thousand of rated current

[1]Reference Exlar Expert software Factory Parameters Peak Current limit

8.4.17. Motor rated current (6075.0)

This object indicates the [1] **Continuous Current rating** of the motor. All relative current data refers to this value.

[1]Reference Exlar Expert software Factory Parameters Continuous Current limit

8.5. Motion Warnings Settings

Motion warnings are user defined setting that will generate warning alerts. These could be used to alert user application of issues before factory defaults occur. Warnings can be found on Exlar Tritex Expert Software page "System Setup/Limits".

Description	Type	Norm	Acc.	NVM	GID	Modbus	CANopen
High Current Warning		% Rated Current	RW	*	SYSTEM.BUS.USERPARAM. IWARNINGHIGH	662	- - - - - -
Current Warning off Level		% Rated Current	RW	*	SYSTEM.BUS.USERPARAM. IWARNINGLOW	663	- - - - - -
High Bus Voltage warning		Volts	RW	*	SYSTEM.BUS.USERPARAM. VWARNINGHIGH	664	- - - - - -
Voltage warning off Level		Volts	RW	*	SYSTEM.BUS.USERPARAM. VWARNINGLOW	666	- - - - - -

Figure 64 – Bus Limits

Description	Type	Norm	Acc.	NVM	GID	Modbus	CANopen
High Board Temperature		0.1 DEG C	RW	*	SYSTEM.THERMAL.USERPARAM. PCB.HIGH	1466	- - - - - -
Low Board Temperature		0.1 DEG C	RW	*	SYSTEM.THERMAL.USERPARAM. PCB.LOW	1467	- - - - - -
High Heat sink Temperature		0.1 DEG C	RW	*	SYSTEM.THERMAL.USERPARAM.H EATSINK.HIGH	1468	- - - - - -
Low Heat sink Temperature		0.1 DEG C	RW	*	SYSTEM.THERMAL.USERPARAM.H EATSINK.LOW	1469	- - - - - -
[1]High Motor Temperature		0.1 DEG C	RW	*	SYSTEM.THERMAL.USERPARAM.M OTOR.HIGH	1470	- - - - - -
[1]Low Motor Temperature		0.1 DEG C	RW	*	SYSTEM.THERMAL.USERPARAM.M OTOR.LOW	1271	- - - - - -

Figure 65 - Thermal Warning Levels

[1] Hardware dependent, refer to hardware manuals if this function is available.

8.5.1. Current

The High Current Warning and Current Warning off Level identifies when the Current Warnings is activated and deactivated. Refer to Diagnostic Bus Warnings for more information.

```
IF (CONTINUOUS_CURRENT > HIGH_CURRENT_WARNING LEVEL)
        SET WARNING_CURRENT_HIGH            // Continuous current warning Active

 If (CONTINUOUS_CURRENT < CURRENT_WARNING_OFF_LEVEL)
        CLEAR WARNING_CURRENT_HIGH          // Clear Continuous current warning
```

NOTE! Continuous Current can be found on Exlar Expert Status Page (Modbus ID 587) and in a CANopen system object (6078.0)

8.5.2. Bus Voltage

The High Bus Voltage warning and Voltage warning off Level identifies when the Voltage Warnings is activated and deactivated. Refer to Diagnostic Bus Warnings for more information.

```
IF (VOLTAGE   > HIGH BUS VOLTAGE WARNING LEVEL)
        SET BUS_WARNING_VOLTAGE
        SET BUS_WARNING_VHIGH

IF (VOLTAGE < VOLTAGE WARNING OFF LEVEL)
        CLEAR BUS_WARNING_VOLTAGE
        CLEAR BUS_WARNING_VLOW
```

8.5.3. Temperatures

The High Temperature warning and Low Temperature identifies when the associate temperature Warnings is activated and deactivated

8.5.3.1. Board Temperature

The High Board Temperature warning and Low Board Temperature identifies when the temperature Warnings is activated and deactivated. Refer to Diagnostic Thermal Warnings for more information

User Parameters Warnings

```
IF (PCB_TEMPERATURE >    HIGH_BOARD_TEMPERATURE_WARNING LEVEL)
        SET THERMAL_WARNING_TEMPERATURE
        SET THERMAL_WARNING_TEMPERATURE_HIGH
        SET THERMAL_WARNING_PCB
        SET THERMAL_WARNING_PCB_HIGH

IF (PCB_TEMPERATURE   < LOW_BOARD_TEMPERATURE LEVEL)
        Set THERMAL_WARNING_TEMPERATURE
        Set THERMAL_WARNING_TEMPERATURE_LOW
```

Set THERMAL_WARNING_PCB
Set THERMAL_WARNING_PCB_LOW

These warnings are cleared automatically and set with the above logic

Factory Temperature Faults

IF (PCB_TEMPERATURE > FACTORY_PCB_ TEMPERATURE_LIMIT)
SET THERMAL_FAULT_TEMPERATURE
SET THERMAL_FAULT_PCB

Test for Stress levels

IF (PCB_TEMPERATURE > FACTORY_PCB_ MONITOR_STRESS_PCB)
SAVE PCB_TEMPERATURE

8.5.3.2. Heat sink Temperature

The High sink Temperature warning and Low sink Temperature identifies when the temperature Warnings is activated and deactivated. Refer to Diagnostic Thermal Warnings for more information

If (Heatsink Temperature > factory Parameters limits heat sink)
Set THERMAL_FAULT_TEMPERATURE
Set THERMAL_FAULT_HEATSINK

If (Heatsink Temperature > User parameter Heatsink high)
Set THERMAL_WARNING_TEMPERATURE
Set THERMAL_WARNING_TEMPERATURE_HIGH
Set THERMAL_WARNING_HEATSINK
Set THERMAL_WARNING_HEATSINK_HIGH

If (Heatsink Temperature < User parameter Heatsink low)
Set THERMAL_WARNING_TEMPERATURE
Set THERMAL_WARNING_TEMPERATURE_LOW
Set THERMAL_WARNING_HEATSINK
Set THERMAL_WARNING_HEATSINK_LOW;

These warnings are cleared automatically and set with the above logic

Factory Temperature Faults

If (Heatsink Temperature > factory Stress Heatsink)
Set THERMAL_STRESS_HEATSINK;

8.5.3.3. Motor Temperature

The High motor Temperature warning and Low motor Temperature identifies when the temperature Warnings is activated and deactivated. Refer to Diagnostic Thermal Warnings for more information

If (motor temperature > factory parameter limits motor)

Set THERMAL_FAULT_TEMPERATURE
Set THERMAL_FAULT_MOTOR;

If (motor temperature > user parameter motor high)
Set THERMAL_WARNING_TEMPERATURE
Set THERMAL_WARNING_TEMPERATURE_HIGH
Set THERMAL_WARNING_MOTOR
Set THERMAL_WARNING_MOTOR_HIGH

If (motor temperature < user parameter low)
Set THERMAL_WARNING_TEMPERATURE
Set THERMAL_WARNING_TEMPERATURE_LOW
Set THERMAL_WARNING_MOTOR
Set THERMAL_WARNING_MOTOR_LOW

These warnings are cleared automatically and set with the above logic

Factory Temperature Faults

If (motor temperature > factory thermal stress motor)
Set THERMAL_STRESS_MOTOR;

9. Homing mode (hm)

Description	Type	Norm	Acc.	NVM	GID	Modbus	CANopen
Control Group							
control word	UINT6		RW		See PAC's		6040.0
Modes of operation	INT16[(b)]		RW		SYSTEM.MOTION.COMMAND. REQUESTEDMODE		6060.0[(M)]
Quick Stop	UINT16		RW		SYSTEM.CONTROL.COMMAND. STOPDECEL		605A.0
Fault Reaction code	INT16		RW		SYSTEM.CONTROL.COMMAND. FAULTACTION		605E.0
Homing Method	INT16 [(b)]		RW		SYSTEM.MOTION.MODES.HOME. PARAM.METHOD		6098.0
Control Status Group							
status word	UINT16		RO		SYSTEM.CONTROL.EVENTS.STATUS		6041.0[(M)]
Modes of operation display	INT16[(b)]		RO		SYSTEM.MOTION.MONITOR. ACTIVEMODE		6061.0[(M)]
Motion Group							
(see Motion Group)							
Profile mode Group							
Mode Controls			RO		SYSTEM.MOTION.MODES.HOME. EVENTS.CONTROL		3500.0
Mode Status			RO		SYSTEM.MOTION.MODES.HOME. EVENTS.STATUS		3501.0
Options	HOME_ OPTION		RW	*	SYSTEM.MOTION.MODES.HOME. PARAM.OPTIONS		3502.0
Homing offset	UINT32	0.0001 Rev	RW	*	SYSTEM.MOTION.MODES.HOME. PARAM.OFFSET		607C.0
Homing acceleration	UINT32	mrps/ s	RW	*	SYSTEM.MOTION.MODES.HOME. PARAM.ACCEL		609A.0
Homing speed (Fast)	UINT32	mrps	RW	*	SYSTEM.MOTION.MODES.HOME. PARAM.VFAST		6099.1
Homing speed (slow)	UINT32	mrps	RW	*	SYSTEM.MOTION.MODES.HOME. PARAM.VSLOW		6099.2
Index offset	INT32	0.0001 Rev	RW	*	SYSTEM.MOTION.MODES.HOME. PARAM.INDEXOFFSET		3504.0
Current limit	UINT16	0.001 rated	RW	*	SYSTEM.MOTION.MODES.HOME. PARAM.CURRENTLIMIT		3505.0
Final Home Position	INT32	0.0001 Rev	RW	*	SYSTEM.POST.USERPARAM. HOMEOFFSET		- - - -

Figure 66 - Homing mode group registers (Run time)

9.1. Controlword – Homing

The mode specific control bits, that is accessible through DS402 controlword (6040.0).

15 9	8	7	6 5	4	3 0
(see 9.2)	**Halt**	(see 9.2)	(Reserved 0)	**Homing operation start**	(see 9.2)
MSB					LSB

Bit	Value	Definition
4	0	Do not start homing procedure
	1	Start or continue homing procedure
8	0	Enable bit 4
	1	Stop drive

Figure 67 - Homing controlword definition

At the rising edge of homing start (bit 4) and Halt is enabled (bit 8 = 0) homing previous homing reference is destroyed.

9.2. Statusword - Homing

The mode specific status bits, that is accessible through DS402 statusword (6041.0).

15 14	13	12	11	10	9	8	7 - 0
(see 3.2)	**Homing error**	Homing attained	(see 3.2)	**Target reached**	(see 3.2)	**Homed**	(see 3.2)
MSB							LSB

Bit 13	Bit 12	Bit 10	Bit 8	Description
0	0	0	0	Homing procedure is in progress
0	0	1	0	Homing procedure is interrupted or not started
0	1	0	0	Homing is attained, but target is not reached
0	1	1	0	Homing procedure is completed successfully
1	0	0	0	Homing error occurred, velocity is not 0
1	0	1	0	Homing error occurred, velocity is 0
1	1	x	0	Reserved
0	0	0	1	Home - absolute reference frame valid

9.3. Homing Offset (607C.0)

This object indicates the configured difference between the zero position for the application and the machine home position (found during homing). During homing, the machine home position is found and once the homing is completed, the zero position is offset from the home position by adding the home offset to the home position. All subsequent absolute moves will be taken relative to this new zero position. This is illustrated in Figure 30. If this object is not implemented, then the home offset will be regarded as zero. Negative values indicate the opposite direction.

9.4. Homing Method (6098.0)

The homing method object determines the method that will be used during homing, positive equalities to defined DS402 methods, negative are Exlar defined.

The allowed homing methods

Value	Name	Definition
-8	POS_SW_ILIMIT_INDEX	Homing on switch (not level sensitive) , current Limiting (Foldback) and index pulse with positive initial direction
-7	POS_SW_ILIMIT	Homing on switch (not level sensitive) and current Limiting (Foldback) with positive initial direction
-6	POS_ILIMIT_INDEX	Homing Current Limiting (Foldback) and index pulse with positive initial direction
-5	POS_ILIMIT	Homing Current Limiting (Foldback) with positive initial direction
-4	NEG_SW_ILIMIT_INDEX	Homing on switch (not level sensitive) , current Limiting (Foldback) and index pulse with negative initial direction
-3	NEG_SW_ILIMIT	Homing on switch (not level sensitive) and current Limiting (Foldback) with negative initial direction
-2	NEG_ILIMIT_INDEX	Homing Current Limiting (Foldback) and index pulse with negative initial direction
-1	NEG_ILIMIT	Homing Current Limiting (Foldback) with negative initial direction
0	None	No homing method assigned
1	NEG_LIMIT_ON_OFF_INDEX	Homing on negative limit switch and index pulse
2	POS_LIMIT_ON_OFF_INDEX	Homing on positive limit switch and index pulse
3	POS_SW_ON_OFF_INDEX	Homing on positive home switch (left) and index pulse
4	POS_SW_OFF_ON_INDEX	Homing on positive home switch (right) and index pulse
5	NEG_SW_ON_OFF_INDEX	Home on negative home switch (left) and index pulse
6	NEG_SW_OFF_ON_INDEX	Home on negative home switch (right) and index pulse
17	NEG_LIMIT_ON_OFF	Home on negative limit switch.
18	POS_LIMIT_ON_OFF	Home on positive limit switch.
19	POS_SW_ON_OFF	Home on positive home switch (inactive).
20	POS_SW_OFF_ON	Home on positive home switch (active).
21	NEG_SW_ON_OFF	Home on negative home switch (inactive).
22	NEG_SW_OFF_ON	Home on negative home switch (active).
33	NEG_INDEX	Home on index pulse with negative initial direction
34	POS_INDEX	Home on index pulse with positive initial direction.
35	CURRENT_POSITION	Home on current position.

Figure 68 - Homing methods

Position Feedback Device	Index Pulse is derive from
Analog Hall (sin/cos)	Once per rev index pulse set when rev angle wraps, reset when if no wrap and abs (rev_angle) < INDEX_PULSE_WIDTH. (sin cos zero)
Incremental Encoder	Index Pulse

Figure 69 - Home Indexing

9.4.1. Method -1: Homing Negative with Current Limit

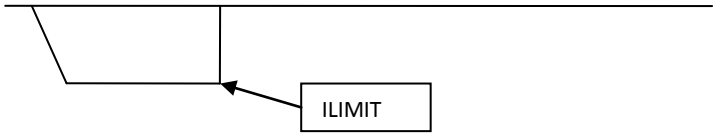


Figure 70 - NEG_ILIMIT (-1)

NEG_ILIMIT (-1)					
			Terminate sequence when		
	Direction	**Velocity**	(↑ of) **Sets**	(↓ of) **Reset**	**Level** (= 1)
1	Negative	Slow	ILIMIT (Home Current Limit)	- - - -	- - - -

9.4.2. Method -2: Homing Negative and Index pulse with Current Limit

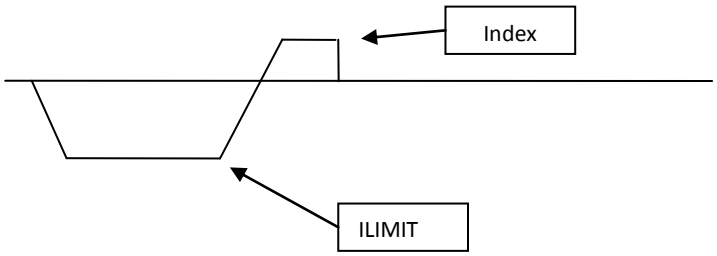


Figure 71 - NEG_ILIMIT_INDEX (-2)

NEG_ILIMIT_INDEX (-2)					
			Terminate sequence when		
	Direction	**Velocity**	(↑ of) **Sets**	(↓ of) **Reset**	**Level** (= 1)
1	Negative	Fast	ILIMIT (Home Current Limit)	- - - -	ILIMIT (Home Current Limit)
2	Positive	Slow	Index	- - - -	- - - -

9.4.3. Method -3: Homing Negative on switch with Current Limit

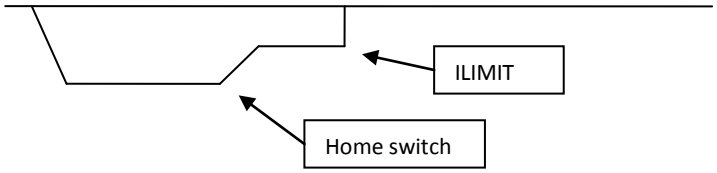


Figure 72 - NEG_SW_ILIMIT (-3)

NEG_SW_ILIMIT (-3)					
			Terminate sequence when		
	Direction	**Velocity**	(↑ of) **Sets**	(↓ of) **Reset**	**Level** (= 1)
1	Negative	Fast	Home Switch	- - - -	Home Switch
2	Negative	Slow	ILimit (Home Current Limit)	- - - -	ILimit (Home Current Limit)

9.4.4. Method -4: Homing Negative on switch and index pulse with Current Limit

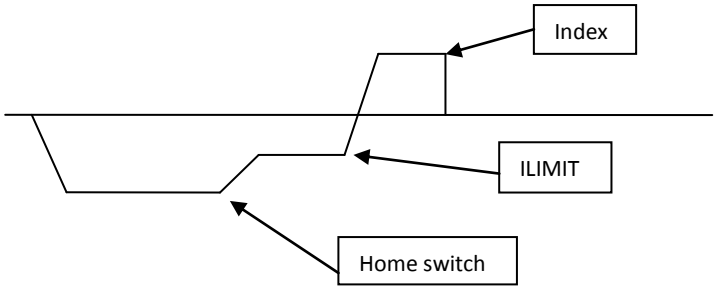


Figure 73 - NEG_SW_ILIMIT_INDEX (-4)

NEG_SW_ILIMIT_INDEX (-4)					
			Terminate sequence when		
	Direction	**Velocity**	(↑ of) **Sets**	(↓ of) **Reset**	**Level** (= 1)
1	Negative	Fast	Home Switch	- - - -	Home Switch
2	Negative	Slow	ILimit (Home Current Limit)	- - - -	ILimit (Home Current Limit)
3	Plus	Slow	Index	- - - -	- - - -

9.4.5. Method -5: Homing Positive with Current Limit

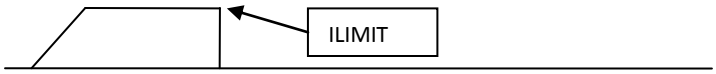


Figure 74 - Homing Positive with Current (-5)

POS_ILIMIT (-5)					
			Terminate sequence when		
	Direction	**Velocity**	(↑ of) **Sets**	(↓ of) **Reset**	**Level** (= 1)
1	Plus	Slow	ILimit (Home Current Limit)	- - - -	ILimit (Home Current Limit)

9.4.6. Method -6: Homing Positive and Index pulse with Current Limit

NOTE! Same as Method -2 but starts in the plus direction.

POS_ILIMIT_INDEX (-6)					
			Terminate sequence when		
	Direction	**Velocity**	(↑ of) **Sets**	(↓ of) **Reset**	**Level** (= 1)
1	Plus	Slow	ILimit (Home Current Limit)	- - - -	ILimit (Home Current Limit)
2	Negative	Slow	Index	- - - -	- - - -

Figure 75 - Homing Positive and Index with Current Limit (-6)

9.4.7. Method -7: Homing Positive on switch with Current Limit

NOTE! Same as Method -3 but starts in the plus direction.

POS_SW_ILIMIT (-7)					
			Terminate sequence when		
	Direction	**Velocity**	(↑ of) **Sets**	(↓ of) **Reset**	**Level** (= 1)
1	Plus	Fast	Home Switch	- - - -	Home Switch
2	Plus	Slow	ILimit (Home Current Limit)	- - - -	ILimit (Home Current Limit)

9.4.8. Method -8: Homing Positive on switch and Index pulse with Current Limit

NOTE! Same as Method -4 but starts in the plus direction.

POS_SW_ILIMIT (-8)					
			Terminate sequence when		
	Direction	**Velocity**	(↑ of) **Sets**	(↓ of) **Reset**	**Level** (= 1)
1	Plus	Fast	Home Switch	- - - -	Home Switch
2	Plus	Slow	ILimit (Home Current Limit)	- - - -	ILimit (Home Current Limit)
3	Negative	Slow	Index	- - - -	- - - -

9.4.9. Method +1: Homing Negative on limit switch and Index pulse

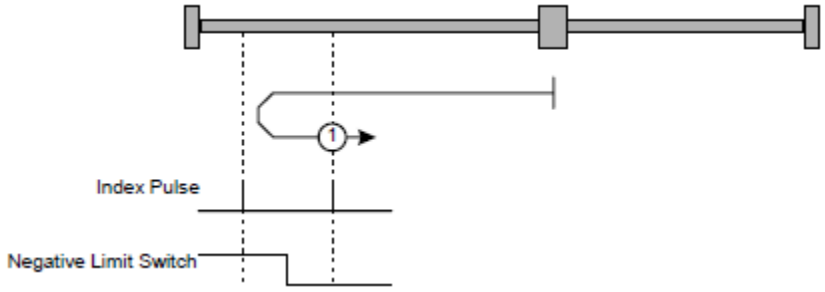


Figure 76 - Homing on negative limit switch and index pulse (+1)

NEG_LIMIT_ON_OFF_INDEX (+1)					
			Terminate sequence when		
	Direction	**Velocity**	(↑ of) **Sets**	(↓ of) **Reset**	**Level** (= 1)
1	Negative	Fast	Negative Limit Switch	- - - -	Negative Limit Switch
2	Plus	Slow	- - - -	Negative Limit Switch	- - - -
3	Plus	Slow	Index	- - - -	- - - -

The initial direction of movement shall be leftward if the negative limit switch is inactive (here: low). The home position is at the first index pulse to the right of the position where the negative limit switch becomes inactive.

9.4.10. Method +2: Homing on positive limit switch and Index pulse

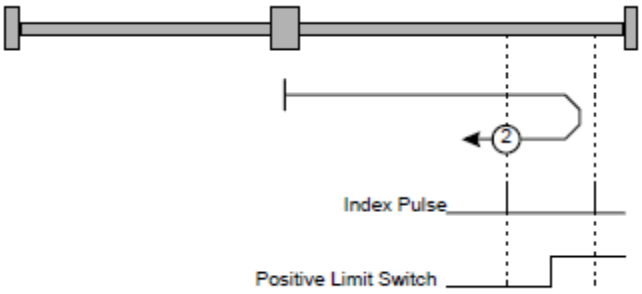


Figure 77 - Homing on positive limit switch and index pulse (+2)

POS_LIMIT_ON_OFF_INDEX (+2)					
			Terminate sequence when		
	Direction	**Velocity**	(↑ of) **Sets**	(↓ of) **Reset**	**Level** (= 1)
1	Plus	Fast	Positive Limit Switch	- - - -	Positive Limit Switch
2	Negative	Slow	- - - -	Positive Limit Switch	- - - -
3	Negative	Slow	Index	- - - -	- - - -

The initial direction of movement is rightward if the positive limit switch is inactive (here: low). The position of home is at the first index pulse to the left of the position where the positive limit switch becomes inactive.

9.4.11. Method +3 and +4: Homing on positive home switch and Index pulse

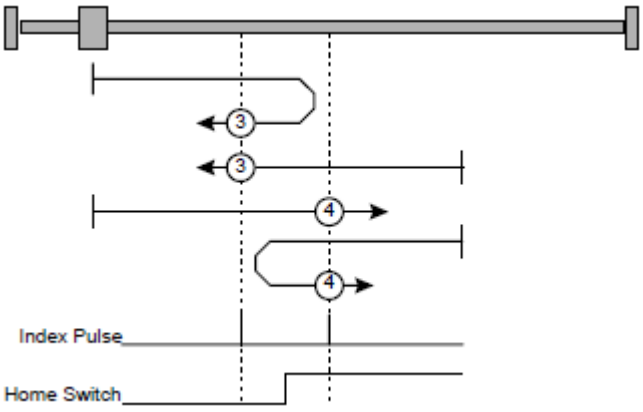


Figure 78 - Homing on positive home switch and index pulse (+3, +4)

The initial direction of movement shall be dependent on the state of the home switch. The home position shall be at the index pulse to either to the left or the right of the point where the home switch changes state. If the initial position is situated so that the direction of movement shall reverse during homing, the point at which the reversal takes place is anywhere after a change of state of the home switch.

POS_SW_OFF_ON_INDEX (+3)					
			Terminate sequence when		
	Direction	**Velocity**	(↑ of) **Sets**	(↓ of) **Reset**	**Level** (= 1)
1	Plus	Fast	Home Switch	- - - -	Home Switch
2	Negative	Slow	- - - -	Home Switch	- - - -
3	Negative	Slow	Index	- - - -	- - - -

POS_SW_OFF_ON_INDEX (+4)					
			Terminate sequence when		
	Direction	**Velocity**	(↑ of) **Sets**	(↓ of) **Reset**	**Level** (= 1)
1	Plus	Fast	- - - -	Home Switch	- - - -
2	Negative	Slow	Home Switch	- - - -	- - - -
3	Negative	Slow	Index	- - - -	- - - -

9.4.12. Method +5 and +6: Homing on Negative home switch and Index pulse

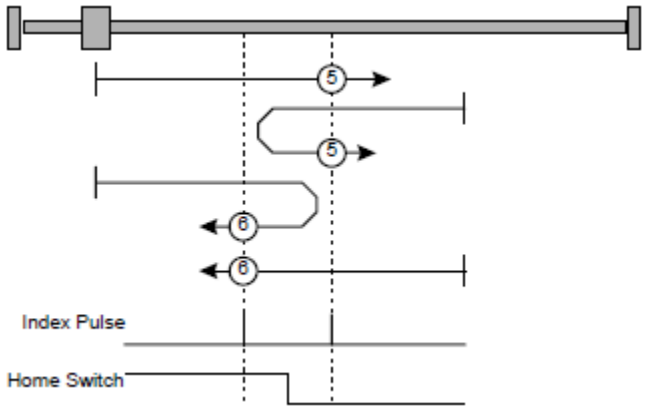


Figure 79 - Homing on negative home switch and index pulse (+5, +6)

The initial direction of movement shall be dependent on the state of the home switch. The home position shall be at the index pulse to either to the left or the right of the point where the home switch changes state. If the initial position is situated so that the direction of movement shall reverse during homing, the point at which the reversal takes place is anywhere after a change of state of the home switch

NEG_SW_ON_OFF_INDEX (+5)					
			Terminate sequence when		
	Direction	**Velocity**	(↑ of) **Sets**	(↓ of) **Reset**	**Level** (= 1)
1	Plus	Fast	Home Switch	- - - -	Home Switch
2	Negative	Slow	- - - -	Home Switch	- - - -
3	Negative	Slow	Index	- - - -	- - - -

NEG_SW_OFF_ON_INDEX (+6)					
			Terminate sequence when		
	Direction	**Velocity**	(↑ of) **Sets**	(↓ of) **Reset**	**Level** (= 1)
1	Plus	Fast		Home Switch	- - - -
2	Negative	Slow	Home Switch	- - - -	- - - -
3	Negative	Slow	Index	- - - -	- - - -

9.4.13. Method +7 to +14: Homing on home switch and Index pulse

These methods use a home switch, which is active over only a portion of the travel; in effect the switch has a 'momentary' action as the axis's position sweeps past the switch. Using the methods 7 to 10, the initial direction of movement shall be to the right, and using methods 11 to 14, the initial direction of movement shall be to the left except if the home switch is active at the start of the motion. In this case, the initial direction of motion shall be dependent on the edge being sought. The home position shall be at the index pulse on either side of the rising or falling edges of the home switch, as shown in Figure xx and Figure xx. If the initial direction of movement leads away from the home switch, the drive shall reverse on encountering the relevant limit switch

NOTE! Methods 7 through 14 are not supported.

9.4.14. Method +17 to +30: Homing without Index pulse

These methods are similar to methods 1 to 14 except that the home position is not dependent on the index pulse but only dependent on the relevant home or limit switch transitions. For example methods 19 and 20 are similar to methods 3 and 4 as shown in Figure

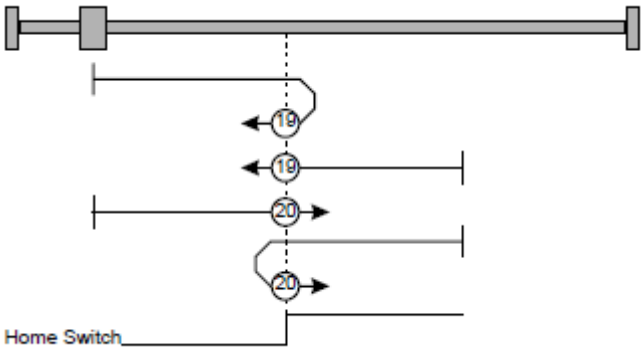


Figure 80 - Homing on positive home switch

NOTE! Methods 22 through 30 are not supported.

NEG_LIMIT_ON_OFF (+19)[1]					
			Terminate sequence when		
	Direction	**Velocity**	(↑ of) **Sets**	(↓ of) **Reset**	**Level** (= 1)
1	Plus	Fast	Home Switch	- - - -	Home Switch
2	Negative	Slow	- - - -	Home Switch	- - - -
3	Negative	Slow	Index	- - - -	- - - -

[1] Method 19 has the same sequence 1 and 2 as method 3 without Index.

9.4.15. Method +33 and +34: Homing on Index pulse

The direction of homing is negative or positive respectively. The home position shall be at the index pulse found in the selected direction as shown in Figure

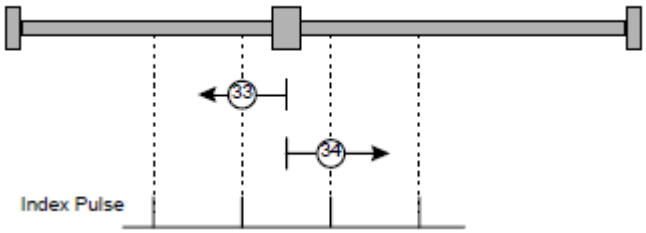


Figure 81 - Homing on index pulse

9.4.16. Method +35: Homing on Current position

The current position shall be taken to be the home position. This method does not require the drive device to be in operational enabled state.

9.5. Homing Switches

User assign physical input to switches (Home, Negative limit and Positive Limit), this can be accomplished through the Tritex Expert software on the Digital I/O Assignment page.

DS402 Description	Internal Generate	Physical Input
Home Switch	- - - -	Configurable Input
Negative Limit Switch	- - - -	Configurable Input
Positive Limit Switch	- - - -	Configurable Input
Index	MOTION.CONTROL.INDEX	- - - -
Home Current Limit	MOTION.CONTROL.IFOLDBACK	- - - -

Figure 82 – Homing switches

9.6. Homing Speed (6099)

The homing speeds object determines the fast (Fast velocity) and slow speeds (Slow velocity) that will be used during homing.

9.7. Homing Acceleration (609A.0)

The Home acceleration configures and deceleration to be used during homing operation.

9.8. Homing Offset (607C.0)

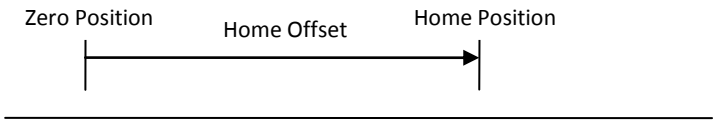


Figure 83 - Home Offset

The difference between the zero position for the application and the zero point of the machine; if a non-zero distance is specified, a final relative distance set-point is executed from the position found by the homing method using the fast velocity before establishing the home position.

9.9. Index Offset

If a non-zero distance is specified, a final relative distance SETPOINT is executed from the position found by the homing method using the fast velocity before establishing the home position.

For example if homing mode is (Method -2) and Index Offset is negative, drive will continue pass detecting Index by Index Offset using fast velocity before stopping at established home position. (See figure 83)

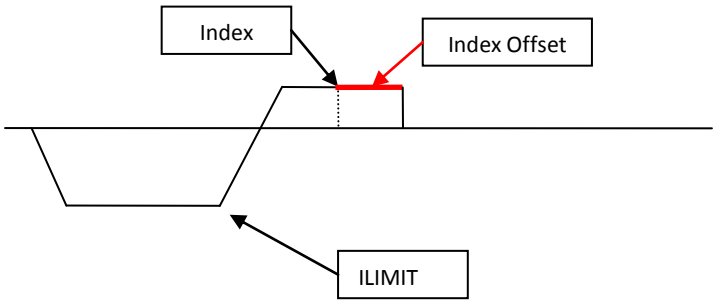


Figure 84 - Index Offset

9.10. Final Home Position

Position zero to the homed position in the absolute position reference frame. Upon completion of a Home or upon receipt of a DEFINEHOME command, the absolute position reference frame offset is adjusted so that the current absolute position is the HOME_OFFSET position (ABS_OFFSET = HOME_POSITION - R_POSITION).

9.11. Mode Controls

Motion Controls are commands are local to motion mode and are only valid when Mode is active.

Motion Mode Control		DS402 – Controlword (6040.0)	
Bit location	Name	Name	Bit
0x0010	HOME_CONTROL_START	CONTROL_CONTROL_OMS4	0x0010
0x0100	HOME_CONTROL_HALT	CONTROL_CONTROL_HALT	0x0100

OMS4 - START
If no positioning is in progress, the rising edge of START (OMS4) with HALT low starts the positioning of the axis.

HALT
Standard mode halts (PAUSE) control implementation.

0 - enable HOME_CONTROL_START
1 - Stop axis according to halt action code (0x605D)

9.12. Mode Status

When mode becomes active the Motion Profile status register will update the CANopen DS402 statusword.

Motion Mode Status		DS402 – Statusword (6041.0)	
Bit location	Name	Name	Bit
0x0001	Mode Active	- - - -	- - - -
0x0100	Homed	HOMED	0x0100
0x0400	Target Reached	TARGET_REACHED	0x0400
0x8000	Mode Halted	HALTED	0x8000
0x1000	Attained	OMS2	0x0100
0x2000	Mode Error	OMS1	0x0200

ACTIVE
Set when the HOME operational mode becomes the active operational mode and cleared when any other operational mode becomes active.

HOMED
Set when homing has completed successfully and the absolute reference frame is valid. Reset when the homing procedure is started (rising START with HALT low). Note that HOMED is NOT cleared when HOME becomes the active operational mode.

HALTED
Set when the HALT command is recognized by the trajectory generator. Target velocity is forced to zero internally; cleared on the falling edge of HALT or ACTIVE.

TARGET_REACHED
Set when the HOME operational mode becomes the active operational mode or when the HOME operational mode is exited and any other operational mode becomes active. Set when a HOME_STATUS_ERROR occurs and velocity is 0, and when the homing procedure completes successfully.

ERROR
Set when position error has remained outside of the position error window for more than the position error window time, cleared otherwise.

SETPOINT_ACTIVE
Set when any set-point remains to be processed and cleared on the rising edge of TARGET_REACHED when all set-points have been processed.

9.13. Options

Motion Mode Options		
Bit location	Name	Description
0x0001U	Auto start Home on Entering Homing mode	Level sensitive START and set START at power-on

9.14. Current Limit

Current Limit will reduce the amount of current allow during the while in Home mode, also is used to trigger homing sequence during homing methods that use ILIMIT to end motion sequence. For example in Homing mode -1 it is used to trigger end of Home move.

MOTION.CONTROL. CURRENT FOLDBACK becomes active when current Foldback limit is reached.

9.15. Application

9.15.1. Homing Commands

Homing Mode have option Command (PACs) available, that are available through the Tritex Drive setup software

Define Reference
The Reference command establishes a absolute reference frame with no change to the absolute position reference frame offset; positions remain continuous but the drive is considered 'homed' with no change to reported positions.

NOTE! Refer Tritex CANopen Getting Started Document for example of sending command to drive through the CANopen bus.

10. Profile Velocity mode (pv)

The profile velocity mode covers the following sub-functions:

Description	Type	Norm	Acc.	NVM	GID	Modbus	CAN
Control Group							
control word	UINT6		RW		See PAC's		6040.0(M)
Quick Stop	UINT16		RW		SYSTEM.CONTROL.COMMAND. STOPACTION		605A.0
Fault Reaction code	INT16		RW		SYSTEM.CONTROL.COMMAND. FAULTACTION		605E.0
Modes of operation	INT16(b)		RW		SYSTEM.MOTION.COMMAND. REQUESTEDMODE		6060.0(M)
Control Status Group							
status word	UINT16		RO		SYSTEM.CONTROL.EVENTS.STATUS		6041.0(M)
Modes of operation display	INT16(b)		RO		SYSTEM.MOTION.MONITOR. ACTIVEMODE		6061.0(M)

Description	Type	Norm	Acc.	NVM	GID	Modbus	CAN
Motion Group							
Profile acceleration	UINT32	mrps/s	RW		SYSTEM.MOTION.COMMAND.PROFILE.ACCEL		6083.0
Profile deceleration	UINT32	mrps/s	RW		SYSTEM.MOTION.COMMAND.PROFILE.DECEL		6084.0
Profile mode Group							
Mode Control	UINT16		RO				- - - -
Mode Status	UINT16		RO		SYSTEM.MOTION.MODES.PROFILEVELOCITY. EVENTS.EVENTS		- - - -
Velocity actual value	INT32	mrps	RO		SYSTEM.MOTION.CONTROL.VLOOP.FEEDBACK		606C.0(M)
target velocity	INT32	mrps	RW		SYSTEM.MOTION.MODES. PROFILEVELOCITY.COMMAND.TARGET		60FF.0(M)
Velocity Demand	INT32	mrps	RO		SYSTEM.MOTION.MODES. PROFILEVELOCITY.STATUS.COMMAND		606B.0
Current Limit (future)	UINT16	0.001 rated	RW		SYSTEM.MOTION.MODES.PV.COMMAND. CURRENTLIMIT		- - - -

Figure 85 -Profile velocity group registers (run time)

Description	Type	Norm	Acc.	NVM	GID	Modbus	CAN
Options	UINT16		RW	*	SYSTEM.MOTION.MODES.PV.PARAM.OPTIONS		- - - -
Current Limit	UINT16	0.001 rated	RW	*	SYSTEM.MOTION.MODES.PV.PARAM.CURRENT LIMIT		- - - -
Target velocity	INT32	mrps	RW	*	SYSTEM.MOTION.MODES.PV.PARAM. TARGET		- - - -

Figure 86 - Velocity Profile registers (Initialize)

10.1. Controlword - Velocity

The mode specific control bits, that is accessible through DS402 controlword (6040.0).

15 9	8	7 6	5 4	0
(see 9.2)	**Halt**	(see 9.2)	Reserved (0)	(see 9.2)
MSB				LSB

Bit	DS402	Value	Definition
8	Halt	0	Execute the motion
		1	Stop (pause) axis
14	MS14	0	
		1	Enables use of PV.COMMAND.CURRENT_LIMIT while PV operational mode is active

Figure 87 – Profile velocity control bits

10.2. Statusword - Velocity

The mode specific status bits, that is accessible through DS402 statusword (6041.0).

15 ... 14	8	12	11	10	9 0
(see 9.3)	Max slippage error	**speed**	(see 9.3)	**Target reached**	(see 93)
MSB					LSB

Bit	Name	Value	Definition
10	Target	0	Halt = 0; Target position not reached

	Reached		Halt=1; decelerating
		1	Halt=0; Target position reached Halt=1; decelerating
12	Speed	0	Speed is not equal to 0
		1	Speed is equal to 0 (within window for time)
13	Max Slippage error	0	**(not supported)**
		1	**(not supported)**

Figure 88 – Statusword velocity mode (vm)

10.3. Velocity Demand value (606B.0)

The output value of the trajectory generator

10.4. Target Velocity (60FF.0)

The Target Velocity is the input to the profile generator and the value is given in,

10.5. Mode Controls

Motion Controls are commands are local to motion mode and are only valid when Mode is active.

Motion Mode Control		DS402 – Controlword (6040.0)	
Bit location	Name	Name	Bit
0x0001	PV_CONTROL_MINUS	- - - -	- - - -
0x0100	PV_CONTROL_HALT	CONTROL_CONTROL_HALT	0x0100
0x4000	PV_CONTROL_ILIMIT	CONTROL_CONTROL_MS14	0x4000

MINUS
Sets the target velocity to the negative of the selected target.

ILIMIT
Enables use of PV.COMMAND.CURRENT_LIMIT while PV operational mode is active.

HALT
Sets the target velocity to zero, using the deceleration ramp in effect.

10.6. Mode Status

When mode becomes active the Motion Profile status register will update the CANopen DS402 statusword.

Motion Mode Status		DS402 – Statusword (6041.0)	
Bit location	Name	Name	Bit
0x0001	PV_STATUS_ACTIVE	- - - -	- - - -
0x0002	PV_STATUS_ACTIVE_PLUS	- - - -	- - - -
0x0004	PV_STATUS_ACTIVE_MINUS	- - - -	- - - -
0x0400	PV_STATUS_TARGET_REACHED	TARGET_REACHED	0x0400
0x1000	PV_STATUS_VZERO	OMS2	0x0100
0x2000	PV_STATUS_VTARGET_ERROR	OMS1	0x0200
0x8000	PV_STATUS_HALTED	MS2	0x8000

ACTIVE
Set when PV becomes the active operational mode and reset when any other operational mode becomes active.

ACTIVE_PLUS
Set when PV.STATUS.ACTIVE and PV.CONTROL.MINUS is inactive.

ACTIVE_MINUS
Set when PV.STATUS.ACTIVE and PV.CONTROL.MINUS is active.

TARGET_REACHED
Set when command velocity is equal to the active setpoint velocity. Reset on the rising or falling edge of ACTIVE.

VZERO_F
Set when feedback velocity has has not exceeded the user's zero velocity window for more than the zero velocity time. VZERO maps to MOTION_STATUS_VZERO_F.

FOLLOWING_ERROR
Set when velocity error has remained outside of the velocity error window for more than the velocity error window time, reset otherwise.

HALTED
Set when the HALT command is recognized by the trajectory generator. Target velocity is forced to zero internally. Reset on the falling edge of HALT or ACTIVE.

10.7. Options

(Not supported)

Motion Mode Options		
Bit location	Name	Description
0x0001U	PV_OPTIONS_ILIMIT	

10.8. Current Limit

(Not supported) Current Limit will reduce the amount of current to allow during the Profile mode.

11. Profile position mode (pp)

Supports two different way to apply target positions to the drive; the two modes are controlled by the timing of the bits for *new_setpoint* and *change_set_immediately* in the control word, and *setpoint_acknowledge* in the status word. These bits allow the setting up of a request response mechanism in order to prepare a set of setpoints while another set is still being processed in the drive unit.

- Single setpoints:
  If the change set immediately bit of the controlword is set to 1, a single set-point is expected by the drive. After reaching the *target_position*, the drive signals its status to a client and then receives a new setpoint. After reaching a *target_position*, the velocity is normally reduced to zero before starting a move to the next setpoint. (Refer to End velocity)

- Set of points:
  If the change set immediately bit of the controlword is set to 0, a set of set-points is expected by the drive. After reaching the *target_position*, the drive immediately processes the next *target position*, which results in a move where the velocity of the drive normally is not reduced to zero after achieving a setpoint.

Description	Type	Norm	Acc	NVM	GID	Modbus	CO
Control Group							
control word	UINT6		RW		See PAC's		6040.0(M)
Modes of operation	INT16(b)		RW		SYSTEM.MOTION.COMMAND. REQUESTEDMODE		6060.0(M)
Quick Stop	UINT16		RW		SYSTEM.CONTROL.COMMAND. STOPACTION		605A.0
Fault Reaction code	INT16		RW		SYSTEM.CONTROL.COMMAND. FAULTACTION		605E.0
Motion Profile Type	INT16		RW		SYSTEM.MOTION.COMMAND. PROFILE.TYPE		6086.0
Control Status Group							
Status word	UINT16		RO		SYSTEM.CONTROL.EVENTS. STATUS		6041.0(M)
Modes of operation display	INT16(b)		RO		SYSTEM.MOTION.MONITOR. ACTIVEMODE		6061.0(M)
Motion Group							
(see Motion Group)							
Profile mode Group							
Mode Control	UINT16		RO		SYSTEM.MOTION.MODES.PP. EVENTS.CONTROL		3600.0
Mode Status	UINT16		RO		SYSTEM.MOTION.MODES. PROFILEPOSITION.EVENTS.STAT US		3602.0
Mode Setup	UINT16		RW		SYSTEM.MOTION.MODES.PP. COMMAND.SETPOINT.OPTIONS		3603.0
Target Position	INT32	0.0001 Rev	RW		SYSTEM.MOTION.MODES.PP. COMMAND.SETPOINT.DISTANCE		607A.0(M)
Position actual value	INT32	0.0001 Rev	RO		SYSTEM.POST.MONITOR.POSITI ON		6064.0
Position Demand value	INT32	0.0001 Rev	RO		SYSTEM.MOTION.MODES.PP. MONITOR.SETPOINT.DISTANCE		6062.0
End Velocity	INT32	mrps	RW		SYSTEM.MOTION.MODES.PP. COMMAND.SETPOINT.VFINAL		6082.0
Current Limit (future)	UINT16	0.001 rated	RW		SYSTEM.MOTION.MODES.PP. COMMAND.SETPOINT.CURRENT LIMIT		3604.0
Position Acceleration(1)	UINT32	mrps/s	RW		SYSTEM.MOTION.MODES.PP. COMMAND.SETPOINT.ACCEL		3605.0
Position Deceleration(1)	UINT32	mrps/s	RW		SYSTEM.MOTION.MODES.PP. COMMAND.SETPOINT.DECEL		3606.0

Figure 89 - Profile Position group registers

(1) If independent data Option is active; Position Mode will use Profile acceleration, and deceleration values instead of global profile data when loading a new SETPOINT.

Description	Type	Norm	Acc.	NVM	GID	Modbus	CAN
Mode Setup	UINT16		RW	*	SYSTEM.MOTION.MODES.PP.PARAM.OPTIONS		3601.0
Max Buffer	UINT16		RW	*	SYSTEM.MOTION.MODES.PP.PARAM.NBUFFEREDSETPOINTS		3607.0

Figure 90 - Position Profile registers (Initialize)

11.1. Controlword – Profile Position

The mode specific control bits, that is accessible through DS402 controlword (6040.0).

15	14	13	12	11 10	9	8	7	6	5	4	3	0
(see 9.2)	**IL**	**SC**	**FC**	(see 9.2)	**COS**	**Halt**	(see 9.2)	**Abs/rel**	**CSI**	**NSP**	(see 9.2)	
MSB												LSB

Bit	Name	Value	Definition
14	I Limit	0	
		1	PP_CONTROL_ILIMIT ***(Not Supported)***
13	Smart Continue	0	
		1	PP_CONTROL_SMART_CONTINUE ***(Not Supported)***
12	Feedback Continue	0	
		1	PP_CONTROL_FBRELATIVE ***(Not Supported)***
8	Halt	0	Positioning will be executed or continued
		1	Axis will stopped
6	Abs/Rel	0	*Target position* is an absolute value
		1	*Target position is* a relative value

Figure 91 – Controlword bits for Profile position (pp) mode

Bits			Definition
9	**5**	**4**	
Change on Set-point (COS)	*Change set Immediately (CSI)*	*New set-point (NSP)*	*DS402*
Change on Set-Point	**Change set-Point**	**New Set-Point**	**Tritex**

0	0	0 -> 1	Positioning is completed (target reached) before the next one gets started
x	1	0 -> 1	Next positioning will start immediately
1	0	0 -> 1	Positioning with the current profile velocity up to the current set-point will be proceeded and then next positioning (see Figure xx and Figure xx) will be applied

PP_CONTROL_ILIMIT
(Manufacture defined bit) ***(Not supported)***

PP_CONTROL_SMART_CONTINUE
(Manufacture defined bit) When a new SETPOINT is to be buffered (not immediately executed) and the PP_CONTROL_CONTINUOUS flag is set the default action is to modify the END_VELOCITY of the previous (or active) set-point to its velocity so that it doesn't stop and targets the new SETPOINT's velocity and distance when it completes. The PP_OPTION_SMART_CONTINUE overrides this behavior to set the previous (or active) SETPOINT's END_VELOCITY to the lesser of the previous (or active) SETPOINT velocity and the new SETPOINT velocity.

PP_CONTROL_FBRELATIVE
(Manufacture defined bit) ***(Not supported)***

11.2. Statusword – Profile Position

The mode specific status bits, that is accessible through DS402 statusword (6041.0).

15	14	13	12	11	10	9	0
(see section 9.3)	*MS2*[(d)]	**Following Error**[(2)]	**Set-point acknowledge**	(see section 9.3)	**Target reached**	(see section 9.3)	
MSB							LSB

Bit	DS402[(d)]	Name	Value	Definition
10	*Target Reached*	Target Reached	0	Halt = 0; Target position not reached Halt = 1; decelerating
			1	Halt = 0; Target position reached Halt = 1; velocity is 0
12	*OMS2*	Set point Ack.	0	Previous set point already process
			1	Previous set point still in process, set point overwriting is accepted
13	*OMS1*	Following Error	0	No following error
			1	Following error[(2)]
14	*MS2*	Set Point NACK	0	
			1	SETPOINT NACK[(1)]

Figure 92 –Statusword for Profile position (pp) mode

(1)SETPOINT NACK
Set when a NEW_SETPOINT command was received but not validated due to either illegal set-point data or non-immediate set-point execution with no free set-point buffer. Cleared when a NEW_SETPOINT command received and validated and on the falling edge of ACTIVE. SETPOINT_NACK may be used as a 'set-point non-acknowledge' indicating that the NEW_SETPOINT command should be reissued.

(2)Following Error
Set when position error has remained outside of the position error window for more than the position error window time, cleared otherwise.

11.3. Position Demand Value (6062.0)

Position demand is the monitor position value.

11.4. Target Position (607A.0)

This object indicates the commanded position that the drive should move to in position profile mode using the current settings of motion control parameters such as velocity, acceleration, deceleration, motion profile type etc. The value of this object is interpreted as absolute or relative depending on the ‘abs/rel' flag in the controlword.

11.5. Profile Velocity (6081.0)

Indicate the configured velocity normally attained at the end of the acceleration ramp during a profiled motion and shall be valid for both directions of motion allowed velocity in either direction during a profiled motion

11.6. End Velocity (6082.0)

The end velocity defines the velocity which the drive must have on reaching the target position. Normally, the drive stops at the target position, i.e. the end velocity = 0. Below is an example what occurs if End Velocity is none zero.

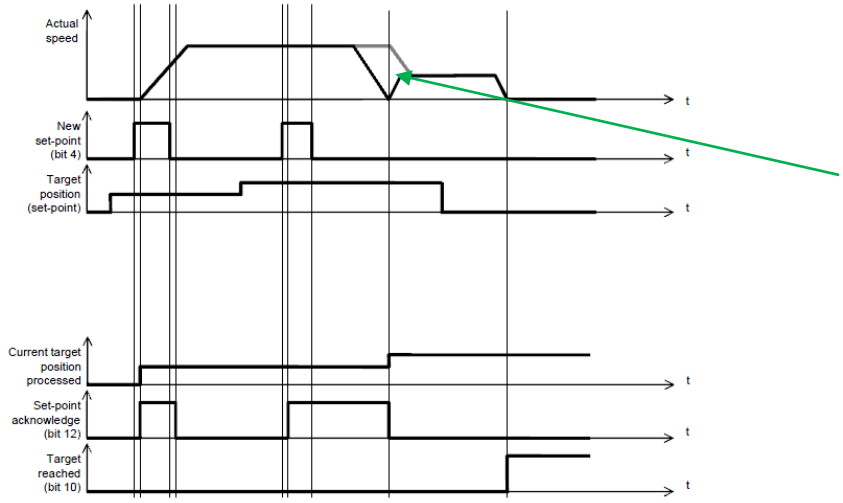


Figure 93 - End Velocity

11.7. Mode Controls

Motion Controls are commands are local to motion mode and are only valid when Mode is active.

Motion Mode Control		Exlar Drive Interface	DS402 – Controlword (6040.0)	
Bit location	Name		Name	Bit
0x0010	PP_CONTROL_NEW_SETPOINT	New Set Point	CONTROL_CONTROL_OMS4	0x0010
0x0020	PP_CONTROL_IMMEDIATE	Change Set Point	CONTROL_CONTROL_OMS3	0x0020
0x0100	PP_CONTROL_HALT	Halt	CONTROL_CONTROL_HALT	0x0100
0x0040	PP_CONTROL_RELATIVE	Relative Position	CONTROL_CONTROL_OMS2	0x0040
0x0200	PP_CONTROL_CONTINUOUS	Change on Set Point	CONTROL_CONTROL_OMS1	0x0200
0x1000	PP_CONTROL_FBRELATIVE	- - - - - - - -	CONTROL_CONTROL_MS12	0x1000
0x2000	PP_CONTROL_SMART_CONTINUE	- - - - - - - -	CONTROL_CONTROL_MS13	0x2000
0x4000	PP_CONTROL_ILIMIT	- - - - - - - -	CONTROL_CONTROL_MS14	0x4000

NEW_SETPOINT
If no positioning is in progress, the rising edge of bit 4 starts the positioning of the axis. If positioning is in progress, bits 5 and 9 are used to determine the appropriate action.

IMMEDIATE
If a rising edge of NEW_SETPOINT is received while positioning is in progress with IMMEDIATE high, a single set-point is expected and the next positioning target is set immediately. If IMMEDIATE is low, the CONTINUOUS bit determines the appropriate action.

HALT
Standard mode halts (PAUSE) control implementation.

0 - positioning executed or continued
1 - Stopped according to halt action code (0x605D)

RELATIVE

0 - Target position is an absolute value
1 - Target position is a relative value

CONTINUOUS
If a rising edge of bit 4 (start) is received while positioning is in progress with bit 5 low (non-immediate), bit 9 is used to determine the appropriate motion. If bit 9 is 0, the positioning in progress is completed to a stop (target reached) before the next positioning is started. If bit 9 is high, the current positioning is completed at the current profile velocity (no-stopping) and then the next positioning is started.

FBRELATIVE
(Future Option)

SMART_CONTINUE
When a new SETPOINT is to be buffered (not immediately executed) and the PP_CONTROL_CONTINUOUS flag is set the default action is to modify the END_VELOCITY of the previous (or active) set-point to its velocity so that it doesn't stop and targets the new SETPOINT's velocity and distance when it completes. The

PP_OPTION_SMART_CONTINUE overrides this behavior to set the previous (or active) SETPOINT's END_VELOCITY to the lesser of the previous (or active) SETPOINT velocity and the new SETPOINT velocity.

ILIMIT
(Future Option)

11.8. Mode Status

When mode becomes active the Motion Profile status register will update the CANopen DS402 statusword.

Motion Mode Status		DS402 – Statusword (6041.0)	
Bit location	Name	Name	Bit
0x0001	PP_STATUS_ACTIVE	- - - -	- - - -
0x0002	PP_STATUS_SETPOINT_ACTIVE	- - - -	- - - -
0x0004	PP_STATUS_AT_VELOCITY	- - - -	- - - -
0x0400	PP_STATUS_TARGET_REACHED	TARGET_REACHED	0x0400
0x0100	PP_STATUS_SETPOINT_ACK	OMS2	0x0100
0x0200	PP_STATUS_FOLLOWING_ERROR	OMS1	0x0200
0x4000	PP_STATUS_SETPOINT_NACK	MS2	0x4000
0x8000	PP_STATUS_HALTED	HALTED	0x8000

ACTIVE
Set when profile positioning mode becomes the active operational mode and cleared when any other operational mode becomes active.

SETPOINT_ACTIVE
Set when any set-point remains to be processed and cleared on the rising edge of TARGET_REACHED when all set-points have been processed.

AT_VELOCITY
Set when command velocity is equal to the active set-point velocity; cleared on the rising or falling edge of ACTIVE and the rising edge of SETPOINT_ACTIVE.

TARGET_REACHED
Set when all set-points have been processed and completed and cleared on the rising edge of ACTIVE or SETPOINT_ACTIVE.

SET_POINT_ACK
This event fires after the drive receives a NEW_SETPOINT and has accepted the new dataset for the set-point. It indicates that the set-point data may again be modified, if necessary. SET_POINT_ACKNOWLEDGE is reset to zero after the falling edge of NEW_SETPOINT to signal that the drive can accept new set-points.

SETPOINT_NACK
Set when a NEW_SETPOINT command was received but not validated due to either illegal set-point data or non-immediate set-point execution with no free set-point buffer. Cleared when a NEW_SETPOINT command received

and validated and on the falling edge of ACTIVE. SETPOINT_NACK may be used as a 'set-point non-acknowledge' indicating that the NEW_SETPOINT command should be reissued.

HALTED
Set when the HALT command is recognized by the trajectory generator. Target velocity is forced to zero internally; cleared on the falling edge of HALT or ACTIVE.

FOLLOWING_ERROR
Set when position error has remained outside of the position error window for more than the position error window time, cleared otherwise.

11.9. Mode Setup

Motion Mode Options		
Bit location	Name	Description
0x0001	FAULT_ON_NACK	fault on SETPOINT_NACK
0x0002	SMART_CONTINUE	next SETPOINT entered with lower velocity
0x0004	INDEPENDENT_DATA	SETPOINT ACCEL/DECEL vs. global
0x0010	Auto Reset NSP	auto reset CONTROL.NEW_SETPOINT when able
0x0020	Reset NSP ON Target	auto reset CONTROL.NEW_SETPOINT on target reached

FAULT ON NACK
If the controller dropped New Set-point command then issued a new NEW_SETPOINT command with IMMEDIATE set without waiting for SETPOINT_ACK to be removed. There is no buffer available in which to store the new Set-point data and a warning are issued. The warning is duplicated in status so that a controller that is monitoring status events can detect the error and recover by reissuing the NEW_SETPOINT command (when SETPOINT_ACK has been removed). The user has an option to fault when a SETPOINT_NACK is generated. If the fault option isn't selected, a warning will be generated.

SMART CONTINUE
When a new SETPOINT is to be buffered (not immediately executed) and the PP_CONTROL_CONTINUOUS flag is set the default action is to modify the END_VELOCITY of the previous (or active) set-point to its velocity so that it doesn't stop and targets the new SETPOINT's velocity and distance when it completes. The PP_OPTION_SMART_CONTINUE overrides this behavior to set the previous (or active) SETPOINT's END_VELOCITY to the lesser of the previous (or active) SETPOINT velocity and the new SETPOINT velocity.

INDEPENDENT DATA
The DS402 defines Profile acceleration/deceleration to function in both Profile position (PP) and Profile velocity (PV) mode. Enabling this allow an independent acceleration / deceleration register to be used for this mode.

NOTE! DS402 specification states that 'Set Point Acknowledge' will not release (transition from High-to-low) until 'New set-point' is release. RESET_NSP and

RESET NSP ON RESET are options to internally release 'New set-point' without Controller intervention.

Auto Reset NSP
Internally resets CONTROL.NEW_SETPOINT as soon as the drive is able to accept another SETPOINT

Reset NSP on Target
Internally resets CONTROL.NEW_SETPOINT (NSP) when STATUS.TARGET_REACHED becomes active.
Typically NSP is reset (return to zero) by controller when Status bit Set point acknowledge is set.

11.10. Current Limit

(Future Option) Current Limit will reduce the amount of current to allow during the Profile mode.

11.11. Maximum Buffer

Maximum number of set-point buffers, maximum allows valve is 8; set-point buffer array is load only after a reset condition has occurred. (Refer to Application note for more information)

11.12. Position Acceleration

While in Position mode and the Independent data option is enabled, Position mode will use an autonomous deceleration value. If independent data option is disabled – global acceleration will be used. Refer to Motion Group- Acceleration for more information.

11.13. Position Deceleration

While in Position mode and the Independent data option is enabled, Position mode will use an autonomous deceleration value. If independent data option is disabled – global deceleration will be used. Refer to Motion Group- Deceleration for more information.

11.14. Application notes

11.14.1. Set Points overview

11.14.1.1. Single set Points

If the bit change_set_immediately is "0"
After data is applied to the drive, a host signals that the data is valid by changing the bit *new_setpoint* to "1" in the Controlword (bit 4). The drive responds with *setpoint_acknowledge* set to "1" in the statusword (bit 12) after it has recognized and buffered the new valid data. Now the host can release *new_setpoint* and subsequently the drive will signal through *setpoint_acknowledge* = "0" its ability to accept new data again.

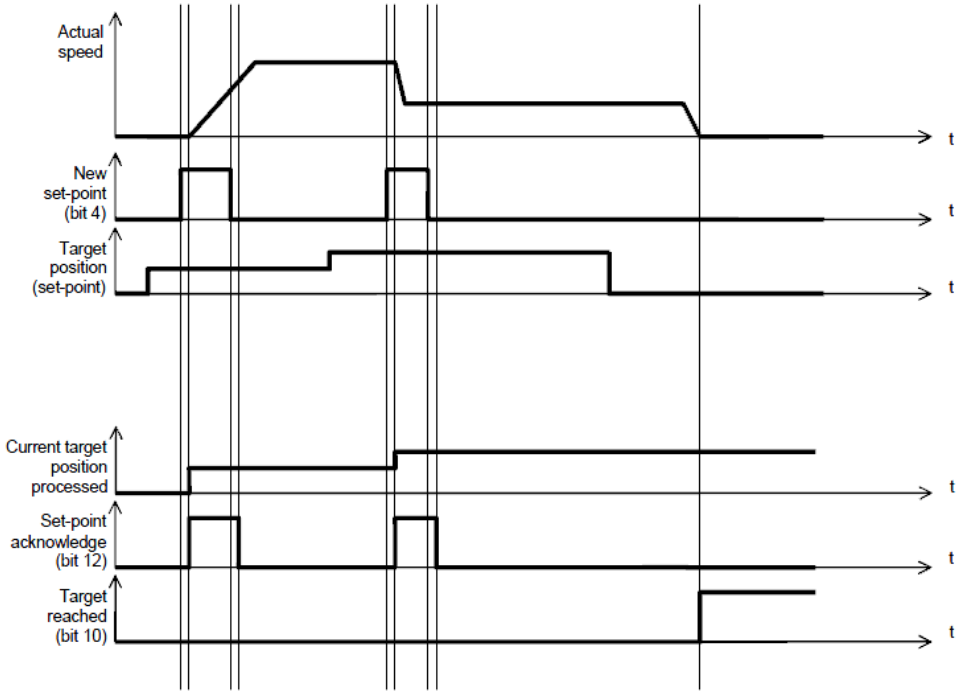


Figure 94 - PP Single Set Point

Single setpoint, if the bit change_set_immediately is "1"

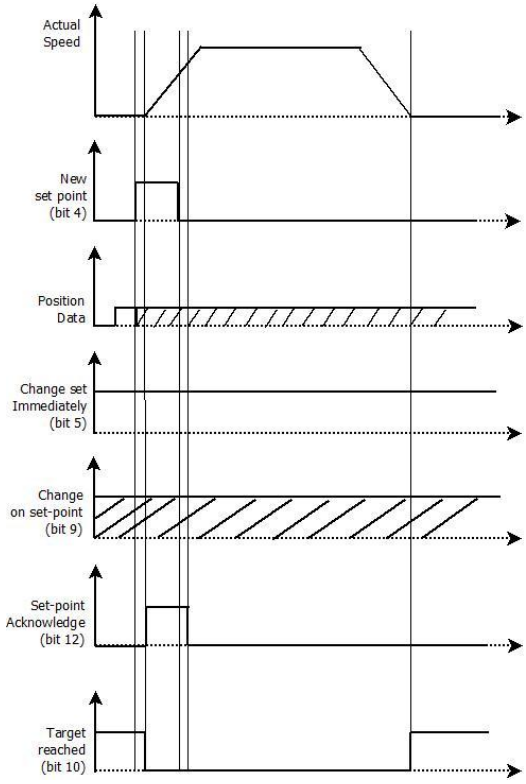


Figure 95 - Single set point

Position Data is capture of data needed to perform a move operation:

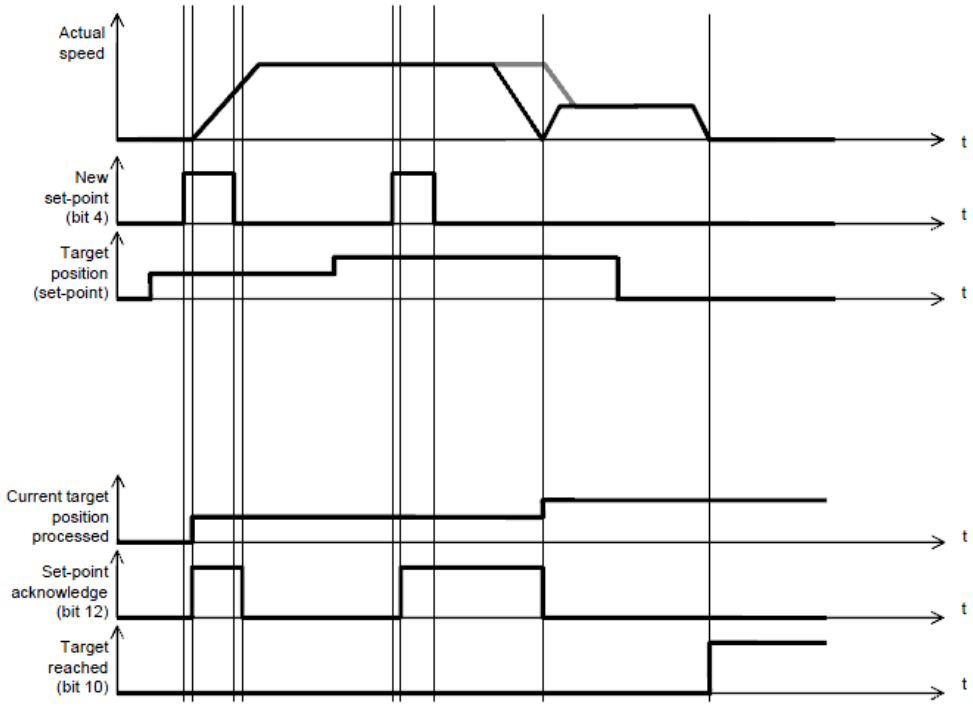
Name	Description	CANopen	GID
Mode setup			
Current limit	Current limit value for the position point		
Position or distance	absolute or relative move is dependent on bit 6 of controlword	607A.0	
Maximum velocity		607F.0	
Acceleration[(1)]			
Deceleration[(1)]			
End velocity		6082.0	

Figure 96 – Position mode data

[(1)] Source is dependent on the Mode Setup optional Independent Data bit. If the independent data is set, Position

11.14.1.2. Sets

When a set-point is in progress and a new set-point is validated by the new set-point (bit 4) in the controlword, the new set-point shall be processed only after the previous has been reached. The handshaking procedure shown in Figure xx is used for the set of set-points method. The additional grey line segment in the graph 'actual speed' shows the actual speed if the *change of set point* bit (bit 9) is set to 1. The number of set-points is dependent on Maximum buffer size.



With change_set_immediately set to "1", the hosts instruct the drive to apply a new setpoint immediately after reaching the previous one.

11.14.2. Additional

New set-points are buffered in the set-point list as long as free set-points are available in the drive device. If no set-point is in progress, the new set-point shall become active immediately. If a set-point is in progress, the new set-point shall be stored in the set-point buffer. If all set-point buffers are busy (*set-point acknowledge´* bit is 1), the reaction depends on the *change set immediately* bit. If the *change set immediately* bit is set to 1, the new set-point shall be processed immediately as single set-point. All previously loaded set-points will be discarded.

The target reached bit will remain 0 until all set-points are processed

- **Position data**
  Is the capture for the move operation, the data can be change before *new_setpoint* becomes active.

- **Moving to a single target position**
  The drive moves to the target position, whereby the velocity is reduced to zero. Reaching the target position is signaled by the bit for *target_reached* in the status word.

- **Moving to several target positions without an intermediate halt**
  After the target position has been reached, the drive moves immediately to the next target position. This requires that new set-points are signaled to the drive. This is done through a positive transition of the *new_setpoint* bit. In this case, the *setpoint_acknowledge* bit must not be active (=1) in the status word. The velocity is not reduced to zero when the first setpoint is reached.

12. Profile torque mode (pt)

The profile torque mode enables the processing of torque set-points and the associated current.

Description	Type	Norm	Acc.	NVM	GID	Modbus	CAN
Control Group							
Controlword	UINT16		RW		See PAC's		6040.0
Modes of operation	INT16[(b)]		RW		SYSTEM.MOTION.COMMAND. REQUESTEDMODE		6060.0
Quick Stop	UINT16		RW	*	SYSTEM.CONTROL.PARAM. STOPACTION		605A.0
Fault Reaction code	INT16		RW	*	SYSTEM.CONTROL.PARAM. FAULTACTION		605E.0
Control Status Group							
Status word	UINT16		RO		SYSTEM.CONTROL.EVENTS.STATUS		6041.0
Modes of operation display	INT16[(b)]		RO		SYSTEM. MOTION. STATUS. ACTIVEMODE		6061.0
Current Actual	UINT16	0.001 rated	RO		SYSTEM.BUS.MONITOR.CONTINUOUSCURRENT		6078.0

Run Time

Description	Type	Norm	Access	NVM	GID	Modbus	CAN
Profile mode Group							
Mode Controls	UINT16		RO		Pacs		3500.0
Mode Status	UINT16		RO		SYSTEM.MOTION.MODES. PROFILETORQUE.EVENTS.STATUS		3501.0
Target torque	INT16	0.001 rated	RW		SYSTEM.MOTION.PROFILE. TORQUE.COMMAND.TARGET		6071.0
Max Current	UINT16	0.001 rated	RW		SYSTEM.MOTION.USERPARAM. LIMITS.CURRENT		6073.0
Torque Demand	INT16	0.001 rated	RO		SYSTEM.MOTION.MODES. PROFILETORQUE.STATUS.COMMAND		6074.0
Torque actual value	INT16	0.001 rated	RO		SYSTEM.MOTION.ILOOP.TFEEDBACK		6077.0
Torque Slope	INT32	0.001 rated/s	RW		SYSTEM.MOTION.PROFILE. TORQUE.COMMAND.SLOPE		6087.0

Figure 97 - Torque Profile group registers (run time)

Initialize

Description	Type	Norm	Access	NVM	GID	Modbus	CAN
Target torque	INT16	0.001 rated	RW		SYSTEM.MOTION.MODES.PT.PARAM. TARGET		- - - -
Acceleration			RW		SYSTEM.MOTION.MODES.PT.PARAM.ACCEL		- - - -

Figure 98 - Torque Profile registers (Initialize)

[b] Internally registers are defined as INT16, CANopen DS402 specify as Byte.

12.1. Controlword – Profile Torque

The mode specific control bits, that is accessible through DS402 controlword (6040.0).

15 … 10	8	7 … 0
(see section 9.2)	**Halt**	(see section 9.2)
MSB		LSB

Bit	Name	Value	Definition
8	**Halt**	0	The motion will execute
		1	The Tritex will stop

Figure 99 –Statusword for Profile torque (pt) mode

12.2. Statusword – Profile Torque

The mode specific status bits, that is accessible through DS402 statusword (6041.0).

15	13	12	11	10	9	0
(see section 9.3)	reserved		(see section 9.3)	**Target reached**	(see section 9.3)	
MSB						LSB

Bit	Value	Definition
10	0	Halt (bit 8 in controlword) = 0; Target torque not reached Halt (bit 8 in controlword) = 1; Axis decelerates
	1	Halt (bit 8 in controlword) = 0; Target torque reached Halt (bit 8 in controlword) = 1; Velocity of axis is 0

Figure 100 –Statusword for Profile torque (pt) mode

12.3. Target Torque (6071.0)

This object indicates the configured input value for the torque controller in profile torque mode. The value is e given per thousand of rated torque

12.4. Motor rated current (6075.0)

This object indicates the [(1)] **Continuous Current rating** of the motor. All relative current data refers to this value.

[(1)]Reference Exlar Expert software Factory Parameters Continuous Current limit

12.5. Torque Actual Value (6077.0)

This object provides the actual value of the torque. It corresponds to the instantaneous torque in the motor. The value is in per thousand of rated torque.

12.6. Torque Slope (6087.0)

This object indicates the configured rate of change of torque. This is given in units of per thousand of rated torque per second.

12.7. Mode Controls

Motion Controls are commands are local to motion mode and are only valid when Mode is active.

Motion Mode Control		DS402 – Controlword (6040.0)	
Bit location	Name	Name	Bit
0x0001	PT.CONTROL.MINUS	- - - -	- - - -
0x0100	PT.CONTROL.HALT	CONTROL.CONTROL.HALT	0x0100

MINUS
Sets the target torque to the negative of the selected target.

HALT
Sets target torque to zero.

12.8. Mode Status

When mode becomes active the Motion Profile status register will update the CANopen DS402 statusword.

Motion Mode Status		DS402 – Statusword (6041.0)	
Bit location	Name	Name	Bit
0x0001	STATUS.ACTIVE	- - - -	- - - -
0x0002	STATUS.SETPOINT.ACTIVE	- - - -	- - - -
0x0004	STATUS.AT.VELOCITY	- - - -	- - - -
0x0400	STATUS.TARGET.REACHED	TARGET.REACHED	0x0400
0x8000	STATUS.HALTED	HALTED	0x8000

ACTIVE
Set when PT mode becomes the active operational mode and reset when any other operational mode becomes active.

ACTIVE.PLUS
Set when PT.STATUS.ACTIVE and PT.CONTROL.MINUS is inactive.

ACTIVE.MINUS
Set when PT.STATUS.ACTIVE and PT.CONTROL.MINUS is active.

TARGET.REACHED
Set when command torque is equal to the active set-point torque. Reset on the rising or falling edge of ACTIVE.

HALTED

Set when the HALT command is recognized by the trajectory generator. Target torque is forced to zero internally. Reset on the falling edge of HALT or ACTIVE.

13. Profile Jog mode

Description	Type	Norm	Entry	NVM	GID	Modbus	CO
Control Group							
control word	UINT6		RW				6040.0
Modes of operation	INT16[(b)]		RW		SYSTEM.MOTION.COMMAND. REQUESTEDMODE		6060.0
Quick Stop	UINT16		RW		SYSTEM.CONTROL.COMMAND. STOPACTION		605A.0
Fault Reaction code	INT16		RW		SYSTEM.CONTROL.COMMAND. FAULTACTION		605E.0
Motion Profile Type	INT16		RW		SYSTEM.MOTION.COMMAND. PROFILE.TYPE		6086.0
Control Status Group							
Status word	UINT16		RO		SYSTEM.CONTROL.EVENTS.STATUS		6041.0
Modes of operation display	INT16[(b)]		RO		SYSTEM.MOTION.MONITOR. ACTIVEMODE		6061.0
Motion Group							
(see Motion Group)							
Profile mode Group							
Mode Controls	UINT16		RO		PACS		3900.0
Mode Status	UINT16		RO		SYSTEM.MOTION.MODES.JOG.EVENTS.STATUS		3901.0
Current Limit (future)	UINT16	0.001 rated	RW		SYSTEM.MOTION.MODES.JOG.COMMAND.CURRENTLIMIT		3909.0
Velocity Fast	INT32	mrps	RW		SYSTEM.MOTION.MODES.JOG.COMMAND.TARGET2		390A.0
Velocity Slow	INT32	mrps	RW		SYSTEM.MOTION.MODES.JOG.COMMAND.TARGET1		390B.0
Acceleration	UINT32	mrps /s	RW		SYSTEM.MOTION.MODES.JOG.COMMAND.ACCEL		390C.0
Deceleration	UINT32	mrps /s	RW		SYSTEM.MOTION.MODES.JOG.COMMAND.DECEL		390D.0

Figure 101 - Jog mode group registers (run time)

Initialize

Description	Type	Norm	Acc.	NVM	GID	Modbus	CO
Options	INT16		RW	*	SYSTEM.MOTION.MODES.JOG.PARAM. OPTIONS		3902.0
Profile	INT16		RW	*	SYSTEM.MOTION.MODES.JOG.PARAM. PROFILE		3903.0
Acceleration	INT32	mrps/s	RW	*	SYSTEM.MOTION.MODES.JOG.PARAM. ACCEL		3904.0
Deceleration	INT32	mrps/s	RW	*	SYSTEM.MOTION.MODES.JOG.PARAM. DECEL		3905.0
Velocity Slow	UINT32	mrps	RW	*	SYSTEM.MOTION.MODES.JOG.PARAM. TARGET1		3906.0
Velocity Fast	UINT32	mrps	RW	*	SYSTEM.MOTION.MODES.JOG.PARAM. TARGET2		3907.0
Current Limit	UINT16	0.001 rated	RW	*	SYSTEM.MOTION.MODES.JOG.PARAM. CURRENTLIMIT		3908.0

Figure 102 - Jog Profile registers (Initialize)

13.1. Controlword - Jog

The mode specific control bits, that is accessible through DS402 controlword (6040.0).

15	14	9 - 13	8	7 ... 0
(see 9.2)	**Limit Current**	(see 9.2)	**Halt**	(see 9.2)
MSB				LSB

Bit	Name	Value	Definition
8	**Halt**	0	The motion will execute
		1	The Tritex will stop
14	**Limit Current**	0	
		1	Limit Current

Figure 103 – Profile Jog control bits

13.2. Statusword - Jog

The mode specific status bits, that is accessible through DS402 statusword (6041.0).

15	14	13	12 - 11	10	9 0
Halted	(see 9.3)	**Following error**	(see 9.3)	**Target Reached**	(see 9.3)
MSB					LSB

Bit	Value	Definition
15	0	Not halted
	1	Halted
13	0	No following error
	1	Following Error
10	0	Target is not reached
	1	Target Reached

Figure 104 -Statusword for Profile Jog mode

13.3. Mode Controls

Motion Controls are commands are local to motion mode and are only valid when Mode is active.

Motion Mode Control		DS402 – Controlword (6040.0)	
Bit location	Name	Name	Bit
0x0001	G.CONTROL.MINUS	- - - -	- - - -
0x0002	JOG.CONTROL.TARGET2	- - - -	- - - -
0x0100	JOG.CONTROL.HALT	CONTROL.CONTROL.HALT	0x0100
0x4000	JOG.CONTROL.ILIMIT	CONTROL.CONTROL.MS14	0x4000

MINUS
Sets the target velocity to the negative of the selected target.

TARGET2
Sets the target velocity to Target2 or -Target2, depending on the state of

ILIMIT
(Future Option) Enables current limiting

HALT
Sets the target velocity to zero, using the deceleration ramp in effect.

13.4. Mode Status

When mode becomes active the Motion Profile status register will update the CANopen DS402 statusword.

Motion Mode Status		DS402 – Statusword (6041.0)	
Bit location	Name	Name	Bit
0x0001	STATUS.ACTIVE	- - - -	- - - -
0x0002	STATUS Active Plus	- - - -	- - - -
0x0004	STATUS Active Minus	- - - -	- - - -
0x0008	STATUS ACTIVE.TARGET1	- - - -	- - - -
0x0010	STATUS ACTIVE.TARGET2	- - - -	- - - -
0x0400	TARGET REACHED	TARGET.REACHED	0x0400
0x0200	STATUS FOLLOWING.ERROR	OMS1	0x0200
0x8000	STATUS HALTED	HALTED	0x8000

ACTIVE
Set when JOG becomes the active operational mode and reset when any other operational mode becomes active.

ACTIVE.PLUS
Set when JOG.STATUS.ACTIVE and JOG.CONTROL.MINUS is inactive.

ACTIVE.MINUS
Set when JOG.STATUS.ACTIVE and JOG.CONTROL.MINUS is active.

ACTIVE.TARGET1
Set when JOG.STATUS.ACTIVE and JOG.CONTROL.TARGET2 is inactive.

ACTIVE.TARGET2
Set when JOG.STATUS.ACTIVE and JOG.CONTROL.TARGET2 is active.

TARGET.REACHED
Set when command velocity is equal to the active set-point velocity. Reset on the rising or falling edge of ACTIVE.

HALTED
Set when the HALT command is recognized by the trajectory generator. Target velocity is forced to zero internally. Reset on the falling edge of HALT or ACTIVE.

FOLLOWING.ERROR
Set when position error has remained outside of the position error window for more than the position error window time, reset otherwise.

13.5. Options

Motion Mode Options		
Bit location	Name	Description
0x0001U	OPTIONS.ILIMIT	

13.6. Current Limit

(Future Option) Current Limit will reduce the amount of current to allow during the Profile mode.

13.7. Velocity Fast

The commanded velocity when Fast Velocity is enabled, when enable is dependent on Home Mode.

13.8. Velocity Slow

The commanded velocity when Slow Velocity is enabled, when enable is dependent on Home Mode

13.9. Acceleration

Acceleration is the acceleration rate for the Profile Jog mode.

NOTE! Acceleration is independent, has no connection to Motion Acceleration (6083.0)

13.10. Deceleration

Deceleration is the deceleration rate for the Profile Jog mode.

NOTE! Deceleration is independent, has no connection to Motion Deceleration (6084.0)

14. Digital IO

This object is the event status of Tritex Inputs and Outputs refer to 'Tritex Expert' manual in method of configuring IO action and events.

Description	Type	Attr	GID	Modbus	CANopen
Digital Inputs		RO	SYSTEM_DIGITAL_EVENTS_INPUTS		60FD.0
Digital Outputs		RO	SYSTEM_DIGITAL_EVENTS_OUTPUTS		60FE.0

14.1. Digital Inputs (60FD.0)

This object indicates the event status of user defined inputs, reference Tritex Expert software document in assigning Digital IO to Inputs.

Input	Bit map
0	0x0001
1	0x0002
2	0x0004
3	0x0008
4	0x0010
5	0x0020
6	0x0040
7	0x0080

Figure 105 – Digital Inputs

14.2. Digital Outputs (60FE.0)

This object indicates the event status of user defined outputs, reference Tritex Expert software document in assigning Digital IO to outputs.

Output	Bit map
0	0x0001
1	0x0002
2	0x0004
3	0x0008

Figure 106 - Digital Outputs

14.3. IO Assignment

Refer to Tritex Drive software for more information.

15. UI Modbus

The UI (User Interface) is the ideal method of configuring and monitoring the drive factory and user parameters. Refer to Tritex Expert software manual for more information.

15.1. RS485 settings

Below is an example of the RS485 interface settings.

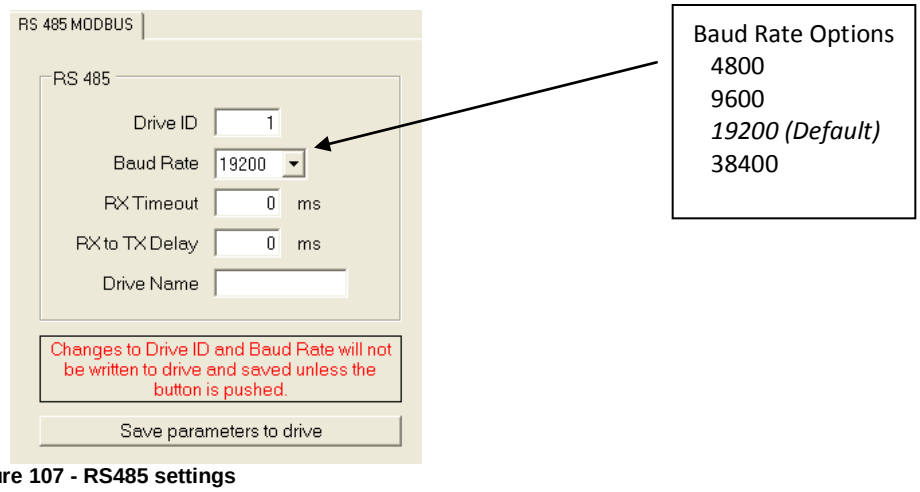


Figure 107 - RS485 settings

16. Scope

The Scope is a diagnostic tool allowing triggering and monitor of registers and events; is designed to be control through the UI Modbus interface. Refer to Tritex Expert software manual for more information.

17. Diagnostic

The following can be found on Tritex Expert Software page “Diagnostics”.

Warnings do not disable the drive, but could limit system operation. Some warnings are automatically cleared when warning condition is cleared others will remained active until reset occurred. Refer to each warning for more details.

Faults will halt the system base on the fault reaction. A Reset command will attempt to clear faults, but if the fault condition is still active, fault reaction will occur.

17.1. Drive Faults / Warnings

Description	Type	Attr	GID	Modbus	CANopen
Faults	UINT16	RO	EVENTS.FAULTS		3000.0(M)
Warnings	UINT16	RO	EVENTS.WARNINGS		3001.0(M)

Figure 108 – Drive status

Bit	Module
0x0001	System
0x0002	CANopen
0x0004	DIO
0x0008	UI (Modus)

Figure 109 - Drive Faults/Warnings bits

Faults and warnings are derive by individual components of the drive and are feed into drive fault and warning system (Figure 50). Some components have multiply subsystem modules, and these need to be examination determine actual fault/warning condition. Below is an example of determine active fault condition.

1. “Read Drive faults “
   Tritex Faults (Figure 112)
   Below example identifies System fault has occurred.

2. “Read Faults”
   System Faults (Figure 115)
   The System is comprised of several subsystems, to determine the actual error, separate read operations are needed. By examining register, it can be determine that a sub-module of system (Digital and Motion) has a fault condition.

3. “Read individual faults”
   A. **System Digital Faults**
      Load Parameters Error
   B. **System Motion Faults**
      Load TUNING.PARAM.LOAD Error

The below example (Figure 114) identifies two faults - default tuning parameters and Digital defaults parameters was loaded.

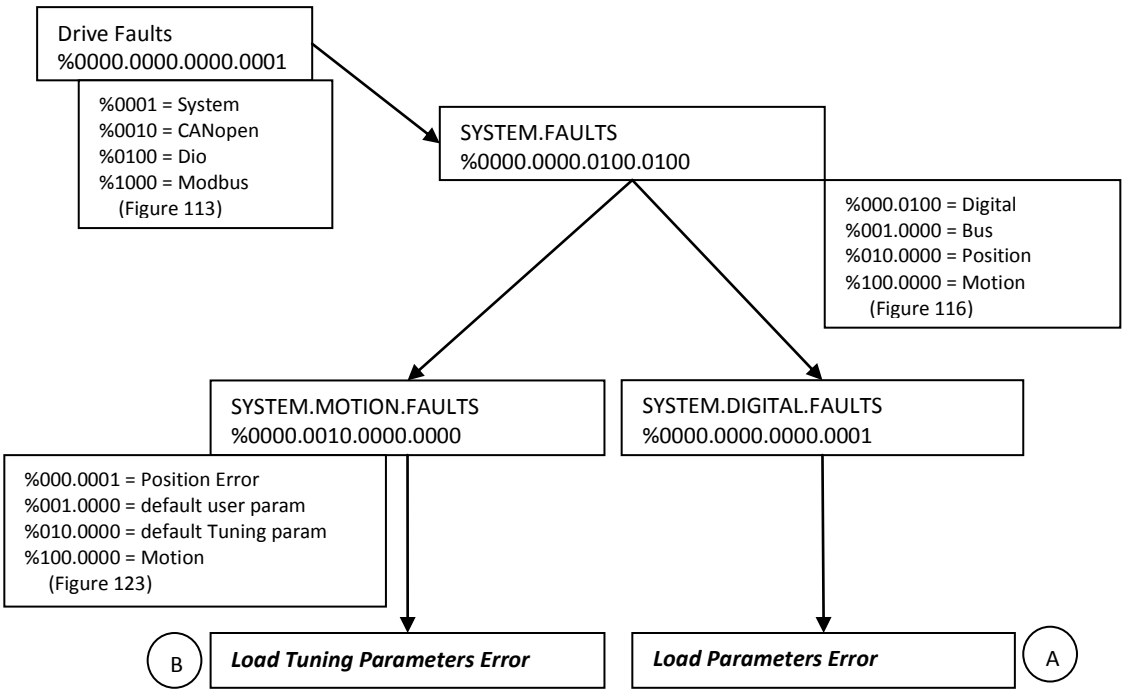


Figure 110 – Tritex Fault example

17.1.1. Motion System

The system Faults and Warnings identifies the overall status of the motion system (NVM, Position tracking, Motion etc).

Description	Type	Acc.	GID	Modbus	CANopen
Faults	UINT16	RO	SYSTEM.EVENTS.FAULTS		- - - - - -
Warnings	UINT16	RO	SYSTEM.EVENTS.WARNINGS		- - - - - -

Figure 111 – System Status

Bit	Module
0x0001	Non-Volatile Memory (NVM)
0x0002	Identification (ID)
0x0004	Digital IO
0x0008	Control
0x0010	Bus
0x0020	position-tracking (POST)
0x0040	Motion
0x0080	History Log
0x0100	Thermal

Figure 112 - System Faults/Warnings

17.1.1.1. Non-Volatile Memory

Description	Type	Attr	GID	Modbus	CANopen
Faults	UINT16	RO	SYSTEM.NVM.EVENTS.FAULTS		30C8.0(M)
Warnings	UINT16	RO	SYSTEM.NVM.EVENTS.WARNINGS		30C9.0(M)
Status	UINT16	RO	SYSTEM.NVM.EVENTS.STATUS		30CA.0(M)

Figure 113 – Non-Volatile State

Bit	Name	Description	Tip
0x0001	Hardware Read Write	Hardware error reading or writing	
0x0002	Firmware Revision	Incompatible NVM and firmware revision	
0x0008	Firmware Product	Incompatible NVM and product firmware	
0x0100	Block Load	User NVM block failed to load	

Figure 114 - NVM Faults

Bit	Name	Description	Tip
0x0001	Hardware Read Write	Updated to newer firmware revision	
0x0002	Firmware Revision	Downloaded to older firmware revision	
0x0100	Block Load	User NVM block failed to save	

Figure 115 - NVM Warnings

Bit	Name	Description	Tip
0x0001	Busy	NVM read or write active	
0x0002	Busy Read	NVM read active	
0x0004	Busy Write	NVM write active	

Figure 116 - NVM Status

17.1.1.2. Identification

Description	Type	Attr	GID	Modbus	CANopen
Faults	UINT16	RO	SYSTEM.ID.EVENTS.FAULTS		3064.0(M)
Warnings	UINT16	RO	SYSTEM.ID.EVENTS.WARNINGS		3065.0(M)

Figure 117 – Identification State

Bit	Name	Description	Tip
0x0001	Load User Parameters Error	Default User parameter loaded	
0x0002	Load Factor Parameters Error	Default Factory parameter loaded	

Figure 118 - ID Faults

Bit	Name	Description	Tip
0x0001	Load User Parameters Error	User parameter did not save	
0x0002	Load Factor Parameters Error	Factory parameter did not save	

Figure 119 – ID Warnings

17.1.1.3. Digital IO

Description	Type	Attr	GID	Modbus	CANopen
Faults	UINT16	RO	SYSTEM.DIGITAL.EVENTS.FAULTS		3258.0(M)
Warnings	UINT16	RO	SYSTEM.DIGITAL.EVENTS.WARNINGS		3259.0(M)

Figure 120 – Digital IO Stats

Bit	Name	Description	Tip
0x0001	Load User Parameters Error	Default User parameter loaded	

Figure 121 – Digital IO Faults

Bit	Name	Description	Tip
0x0001	Parameters Save	Parameters did not saved	
0x0002	Parameter Range	Parameter(s) out-of-Range	

Figure 122 – Digital IO - Warnings

17.1.1.4. Control

Description	Type	Attr	GID	Modbus	CANopen
Faults	UINT16	RO	SYSTEM.CONTROL.EVENTS.FAULTS		312C.0(M)
Warnings	UINT16	RO	SYSTEM.CONTROL.EVENTS.WARNINGS		312D.0(M)
Status	UINT16	RO	SYSTEM.CONTROL.EVENTS.STATUS		312E.0(M)

Figure 123 – Control State

Bit	Name	Description	Tip
0x0001	Load User Parameters Error	Default User parameter loaded	
0x0002	User PARAM Loaded	Default User parameter loaded	
0x0008	Internal	Internal fault	

Figure 124 – Controls Faults

Bit	Name	Description	Tip
0x0001	Parameters Save	Parameters did not save	
0x0002	Parameter Range	Parameter(s) out-of-Range	
0x2000	Unrecognized Fault Action	Unrecognized fault action	
0x4000	Unrecognized Stop	Unrecognized stop action	

	Action		
0x8000	Zero Deceleration	Stop deceleration is zero	

Figure 125 – Control - Faults/Warnings

Bit	Name	Description	Tip
0x0001	Run	Run mode active (ready to switch on power)	
0x0002	Ready	Ready (power switched on)	
0x0004	Enable	Operationally enabled	
0x0008	Fault	Fault active	
0x0010	DC bus Ready	High voltage applied	
0x0020	Stop Inactive	Stop inactive	
0x0040	Setup	Setup mode active (switch on disabled)	
0x0080	Warning	Warning active	
0x0100	Homed	Absolute reference frame valid (manufacturing specific 3)	
0x0200	Remote	Remote control/active (control word is processed)	
0x0400	Target Reached	Target reached / operation mode changed (mode specific)	
0x0800	Internal Limit	Internal limit active (mode specific)	
0x1000	OMS2	Operation mode specific 2	
0x2000	OMS1	Operation mode specific 1	
0x4000	MS2	Manufacturing specific 2 (unused)	
0x8000	Halted	Halted/paused (manufacturing specific 1)	

Figure 126 - Control Status

17.1.1.5. Bus

Description	Type	Attr	GID	Modbus	Index
Faults	UINT16	RO	SYSTEM.BUS.EVENTS.FAULTS		3190.0(M)
Warnings	UINT16	RO	SYSTEM.BUS.EVENTS.WARNINGS		3191.0(M)
Status	UINT16	RO	SYSTEM.BUS.EVENTS.STATUS		3192.0(M)

Figure 127 – Bus State

Bit	Name	Description	Tip
0x0001	Load User Parameters Error	Default User parameter loaded	
0x0002	Load Factor Parameter Error	Default Factory parameter loaded	
0x0010	Voltage Low	Bus voltage under low limit	If Drive state is Enabled and (voltage < Factory Parameter **Low voltage Trip** Limit)
0x0020	Voltage High	Bus voltage over high limit	If Monitor voltage > Factory Parameter **High voltage Trip** Limit
0x0040	Voltage	Bus voltage outside warning limits	If Voltage Low or Voltage High is set
0x0080	Hardware Current	Hardware current trip	Hardware control current trip has occurred
0x0100	Current High	High current fault	Monitor current > Factory Parameter **Fault Trip Current**
0x0200	Continuous Current	Continuous current fault	Continuous current > 1000 % of Rated Current
0x0400	Current	Current fault	Set when any of the Current faults are active

Figure 128 – Bus Faults

Bit	Name	Description	Tip
0x0001	Load User Parameters	User parameter did not save	
0x0002	Load factory Parameter	Factory parameter did not save	
0x0010	Voltage Low	Bus voltage under low warning limit	Motion Warnings Settings 8.5
0x0020	Voltage High	Bus voltage over high warning limit	Motion Warnings Settings 8.5
0x0040	Voltage	Bus voltage outside warning limits	Set if Voltage Low or Voltage High is set
0x0100	Current High	Continuous current warning	Motion Warnings Settings 8.5
0x0200	Rated Current	Continuous Current magnitude over rated	Set if (BUS.MONITOR.CURRENT > 1000) indicating that current being demanded is greater than rated current and will lead to an eventual continuous current fault if not reduced. Reset if (BUS.MONITOR.CURRENT <= 1000).

Figure 129 - Bus Warnings

Bit	Name	Description
0x0001	Shunt	Shunt active
0x0002	Shut Limit	Shunt out of available energy
0x8000	PWM Active	PWM's enabled

Figure 130 - Bus Status

17.1.1.6. Position Tracking

Description	Type	Attr	GID	Modbus	Index
Faults	UINT16	RO	SYSTEM.POST.EVENTS.FAULTS		31F4.0(M)
Warnings	UINT16	RO	SYSTEM.POST.EVENTS.WARNINGS		31F5.0(M)
Status	UINT16	RO	SYSTEM.POST.EVENTS.STATUS		31F6.0(M)

Figure 131 – Position Tracking State

Bit	Name	Description	Tip
0x0001	Load Factor Parameters Error	Default Factory parameter loaded	
0x0002	Load User Parameters Error	Default User parameter loaded	
0x2000	No Device	no tracking device selected	see Factory Parameters Position Feedback - Type
0x4000	Device Communication	internal communication lost to device	
0x8000	Tracking Error	Tracking error / loss of signal	

Figure 132 - Position Faults

Bit	Name	Description	Tip
0x0001	Load Factor Parameters	Factory parameter did not save	
0x0002	Load User Parameters	User parameter did not save	
0x4000	Signal	Signal degradation	
0x8000	Tracking Warning	Tracking error +/- 5 degrees mechanical or tracking rate > 60000 rpm	

Figure 133 - Position Warnings

Bit	Name	Description
0x0001	W	W (encoder)
0x0002	V	V (encoder)
0x0004	U	U (encoder)
0x2000	Index	Once per rev index pulse
0x4000	Homed	Absolute reference frame has been established
0x8000	Active	Feedback device is active (enabled)

Figure 134 - Position Track Status

17.1.1.7. Motion

Description	Type	Attr	GID	Modbus	Index
Faults	UINT16	RO	SYSTEM.MOTION.EVENTS.FAULTS		3384.0(M)

Warnings	UINT16	RO	SYSTEM.MOTION.EVENTS.WARNINGS		3385.0(M)
Status	UINT16	RO	SYSTEM.MOTION.EVENTS.STATUS		3386.0(M)
Control	UINT16	RO	SYSTEM.MOTION.EVENTS.CONTROL		- - - -

Figure 135 – Motion State

Bit	Name	Description	Tip
0x0001	Position Error Fault	Position error outside window for time	Reference System Setup **Following Error Time Limit**
0x0100	Load User Parameters Error	Default User parameters loaded	
0x0200	Load Tuning Parameters Error	Default Tuning parameters loaded	
0x0400	Load Factor Parameters Error	Default Factory parameters loaded	
0x8000	Operational Mode Fault	Operational mode fault	If (Position Profile Mode and Fault on Nack is set) Reference Position Profile – Mode Setup

Figure 136 - Motion - Faults

Bit	Name	Description	Tip
0x0001	Position Error	Position error outside window	Reference System Setup **Motion Max Following Error**
0x0100	Load User Parameters	User parameters did not save	
0x2000	Tuning Parameters Did not save	Tuning parameters did not save	

Figure 137 - Motion - Faults/Warnings

Bit	Name	Description	Tip
0x0002	Velocity Command Zero	Velocity command = 0	
0x0004	Velocity Target Zero	Velocity command = target	
0x0008	Velocity Feedback Zero	abs(velocity feedback) outside window for time	
0x0010	Velocity Target Feedback	abs(velocity target - velocity feedback) outside target window for time	
0x0100	Current Limit	Velocity loop saturated / limiting current command	
0x0200	Current Limit Positive	Velocity loop saturated / limiting (+) current command	
0x0400	Current Limit negative	Velocity loop saturated / limiting (-) current command	
0x0800	Current Foldback	Current command exceeds current Foldback limit	

Figure 138 - Motion Status

Bit	Name	Description	Tip
0x0001	Switch 1	generic switch 1	
0x0002	Switch 2	generic switch 2	
0x0004	Switch 3	generic switch 3	
0x0008	Switch 4	generic switch 4	
0x0010	Index	position tracking device index pulse command	
0x0020	Current Foldback	current FOLDBACK limit active	

Figure 139 – Motion Control

17.1.1.8. History Log

Description	Type	Attr	GID	Modbus	Index
Faults	UINT16	RO	SYSTEM.LOG.EVENTS.FAULTS		3320.0(M)
Warnings	UINT16	RO	SYSTEM.LOG.EVENTS.WARNINGS		3321.0(M)
Status	UINT16	RO	SYSTEM.THERMAL.EVENTS.STRESS		3322.0(M)

Figure 140 – History Log State

Bit	Name	Description	Tip
0x0001	STATUS_LOAD	Status log failed to load	
0x0002	STRESS_LOAD	Stress log failed to load	
0x0004	RECENT_LOAD	Recent fault log failed to load	
0x0008	MODULE_LOAD	Module fault log failed to load	
0x0010	SYSTEM_LOAD	System fault log failed to load	

Figure 141 - History Warning

Bit	Name	Description	Tip
0x0001	Updating NVM	Updating NVM	
0x0002	Logging	Logging (anything)	
0x0004	Logging Status	Logging to status log	
0x0008	Logging Stress	Logging to stress log	
0x0010	Logging Fault	Logging to fault logs	
0x0020	Resetting Counts	Resetting fault counts	
0x0040	Clearing Status	Clearing status log	
0x0080	Clearing Stress	Clearing stress log	
0x0100	Clearing Recent	Clearing recent fault log	
0x0200	Clearing Module	Clearing module fault log	
0x0400	Clearing System	Clearing system fault log	
0x0800	Pending	NVM update pending	
0x1000	Stress pending	tress NVM update pending	
0x2000	Recent Pending	Recent fault NVM update pending	
0x4000	Module Pending	Module fault NVM update pending	
0x8000	System Pending	System fault NVM update pending	

Figure 142 – History Log Status

17.1.1.9. Thermal

Description	Type	Attr	GID	Modbus	Index
Faults	UINT16	RO	SYSTEM.THERMAL.EVENTS.FAULTS		32BC.0(M)
Warnings	UINT16	RO	SYSTEM.THERMAL.EVENTS.WARNINGS		32BD.0(M)
Stress	UINT16	RO	SYSTEM.THERMAL.EVENTS.STRESS		32BE.0(M)

Figure 143 – Thermal State

Bit	Name	Description	Tip
0x0001	User Parameter Load	Default User parameters loaded	(1) If using Tritex Expert software – read parameters
0x0002	Factory Parameter Load	Default Factory parameters loaded	

			from drive then write parameters to drive. (2) If step 1 fail – contact service.
0x0010	Temperature	PCB, heat sink, or motor over factory thermal limit	Factory Parameters /Limits/
0x0020	PCB	PCB over factory thermal limit	Factory Parameters /Limits/ **Board Temperature Trip**
0x0040	Heat-sink	Heat sink over factory thermal limit *(Dependent on Hardware)*	Factory Parameters /Limits/ **Heat Sink Temperature Trip**
0x0080	Actuator	Motor over factory thermal limit *(Dependent on Hardware)*	Factory Parameters /Limits/ **Actuator Temperature Trip**

Figure 144 - Thermal Faults

Bit	Name	Description	Tip
0x0001	USER.PARAM.SAVE	User parameters failed to save	
0x0002	FACTORY.PARAM.SAVE	Factory parameter failed to save	
0x0010	TEMPERATURE	PCB, heat sink or motor temp. outside warning limits	Motion Warnings Settings 8.5.3
0x0020	TEMPERATURE.HIGH	PCB, heat sink or motor temp. above high warning limit	
0x0040	TEMPERATURE.LOW	Heat sink or motor temp. below low warning limit	
0x0080	PCB	PCB temp. outside warning limits	Motion Warnings Settings 8.5.3
0x0100	PCB.HIGH	PCB temp. above high warning limit	
0x0200	PCB.LOW	PCB temp. below low warning limit	
0x0400	HEATSINK	Heat sink temp. outside warning limits	Motion Warnings Settings 8.5.3
0x0800	HEATSINK.HIGH	Heat sink temp. above high warning limit	
0x1000	HEATSINK.LOW	Heat sink temperature below low warning limit	
0x2000	MOTOR	Motor TEMP. outside warning limits	Motion Warnings Settings 8.5.3
0x4000	MOTOR.HIGH	Motor TEMP. above high warning limit	
0x8000	MOTOR.LOW	Motor TEMP. below low warning limit	

Figure 145 - Thermal Warnings

Bit	Name	Description	Tip
0x0001	PCB	New maximum PCB temperature observed	
0x0002	Heat Sink	New maximum heat-sink temperature observed	
0x0004	Motor	New maximum motor temperature observed	

Figure 146 - Thermal Stress

17.1.2. CANopen

NOTE! If installed, below is the CANopen interface diagnostic information.

Description	Type	Attr	GID	Modbus	CANopen
Faults	UINT16	RO	CANOPEN.EVENTS.FAULTS		2110.1(M)
Warnings	UINT16	RO	CANOPEN.EVENTS.WARNINGS		2110.2(M)

Figure 147 - CANopen State

Bit	Name	Description	Tip
0x0001	User Parameters Loading fault	Default User parameters loaded	(1) If using Tritex Expert software – read parameters from drive then write parameters to drive. (2) If step 1 fail – contact service.
0x0002	Factory Parameters Loaded	Default Factory parameters loaded	
0x0010	Command Rights	Motion Command rights error	See Motion Control rights
0x0020	Heart Beat	Heartbeat fault	Lost Communication
0x0040	PDO Did not Process	PDO Data length fault	Check PDOs lengths
0x0080	Message Timeout	CAN transmit time out	
0x0100	CAN Bus-off- Active	CAN hardware bus off active	CAN interface is in Bus off condition will attempt to reset interface.

Figure 148 - CANopen Faults

Bit	Name	Description	Tip
0x0001	Save User Parameter Error	User parameters failed to save	
0x0002	Save Factory Parameter Error	Factory parameters failed to save	
0x0020	CAN Hardware Warning	CAN hardware CAN warning	
0x0040	CAN Hardware Overflow	CAN hardware overflow warning	
0x0080	CAN Bus Passive	CAN hardware bus passive	No active (not connected or Baud rate) CAN device on network. Cleared on next valid CAN message.
0x0100	TPDO1 Configuration	TPDO-1 Configuration warning	
0x0200	TPDO2 Configuration	TPDO-2 Configuration warning	
0x0400	TPDO3 Configuration	TPDO-3 Configuration warning	
0x0800	TPDO4 Configuration	TPDO-4 Configuration warning	
0x100	RPDO1 Configuration	RPDO-1 Configuration warning	
0x2000	RPDO2 Configuration	RPDO-1 Configuration warning	
0x4000	RPDO3 Configuration	RPDO-1 Configuration warning	
0x8000	RPDO4 Configuration	RPDO-1 Configuration warning	

Figure 149 - CANopen Warnings

17.1.3. UI (Modbus)

NOTE! If installed, below is the UI (Modbus interface diagnostic information.

17.1.4. DIO

NOTE! If installed, below is the Digital IO interface diagnostic information.

18. Motion Control Rights

NOTE! All interfaces (Modbus, Digital I/O, and CANopen) have a method to configure control access. If interface does not have access rights, command will not execute.

NOTE! Dependent on installed software module, Below example outline Modbus. CANopen and Digital I/O and Command Rights

Commands are divided into 11 groups (see below); each interface can be configured whether it will allow these group commands control or exclusives rights to command these functions.

Share Control vs. Exclusive Control
Sharing allows multiply controllers the capability of commanding commands while, exclusive rights only allow one control rights to the command group; these rights are determine at power up. If more than one interface request exclusive rights to commands, Interface with the highest priority will be granted the right.

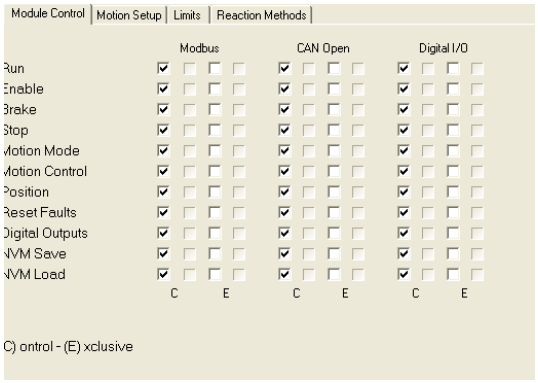


Figure 150 - Example Module Control Wedge

Bit	Group access Name	Example of Allowed Commands
0000	No access	Not allow to command Tritex
xxx1	KEY	key on/off, restart, etc
xxx2	ENABLE	bridge enabling/disabling
xxx4	Brake	brake release overriding
xxx8	Stop	stop/hold/pause/emergency
xx1x	Motion Mode	motion mode selection/initiation
xx2x	Motion CONTROL	motion mode control (start)
xx4x	POSITION	abs position change
xx8x	FAULT.RESET	fault resets

x1xx	DIGITAL.OUTPUTS	h/w digital outputs/leds on/off
x2xx	NVM.SAVE	NVM save
x4xx	NVM.LOAD	NVM load

Figure 151 - Group Access bits

18.1. Control Access Request

Each interface will request there access rights at power-up or reset and depended on there priority which rights they will be granted.

NOTE! Interface access is depended if interface is installed. Default configure is all interfaces have control access.

Description	Type	Attr	Priority	GID	Modbus	CANopen
Modbus Control	UINT16	RW	2	MODBUSUI.CMDCONTROL.CONTROL		- - - - - -
Modbus Exclusive	UINT16	RW	2	MODBUSUI.CMDCONTROL.EXCLUSIVE		- - - - - -
CANopen Control	UINT16	RW	1	CANOPEN.CMDCONTROL.CONTROL		2100.1
CANopen Exclusive	UINT16	RW	1	CANOPEN.CMDCONTROL. .EXCLUSIVE		2100.2
Digital Control	UINT16	RW	3	DIO.CMDCONTROL.CONTROL		- - - - - -
Digital Exclusive	UINT16	RW	3	DIO.CMDCONTROL.EXCLUSIVE		- - - - - -
Analog Control	UINT16	RW	TBD	***<Future>***		- - - - - -
Analog Exclusive	UINT16	RW	TBD	**<Future>**		- - - - - -

Figure 152 - Control Access Requested

18.2. Control Access Actual

Description	Type	Attr	GID	Modbus	CANopen
Modbus Control	UINT16	RO	MODBUSUI.PARAM.CMDCONTROL.CONTROL		- - - - - -
Modbus Exclusive	UINT16	RO	MODBUSUI.PARAM.CMDCONTROL.EXCLUSIVE		- - - - - -
Digital Control	UINT16	RO	DIO.PARAM.CMDCONTROL.CONTROL		- - - - - -
Digital Exclusive	UINT16	RO	DIO.PARAM.CMDCONTROL.EXCLUSIVE		- - - - - -
CANopen Control	UINT16	RO	CANOPEN.PARAM.CMDCONTROL.CONTROL		2101.1
CANopen Exclusive	UINT16	RO	CANOPEN.PARAM.CMDCONTROL.EXCLUSIVE		2101.2
Analog Control	UINT16	RO	***<Future>***		- - - - - -
Analog Exclusive	UINT16	RO	***<Future>***		- - - - - -

Figure 153 - Control Access Actual

18.3. Override

The override function is a command to override current rights, and grant all rights to the UI (Modbus) interface.

NOTE! Contract support for more information.

19. Appendix

19.1. GID

GID or Global Identification is the method used by the Tritex drive to map internal variables. These GID's are then cross reference to installed protocols. For example a GID of 0x3C000000 represents the System warnings and is identify as "SYSTEM.MOTION.EVENTS.FAULTS" cross reference to Modbus ID " 1900 " and CANopen ID "Index 3384, subindex 0".

Name	Global ID	Value	Modbus	CANopen
System Warnings	SYSTEM.MOTION.EVENTS.FAULTS	0x3C000000	1900	3384.0

Figure 154 - Global ID vs. Interface ID Example

19.2. PAC

Programmable Access Commands (PAC's) are functions that perform operations in the system. For example the below PAC function will define current absolute position as home, while home mode is active. Refer to Interface section (UI Modbus, and CANopen) for methods of sending PACs to drive. Note, if interface does not have control rights command will not execute.

PAC Description	Pac Command	Value	Control Rights Needed
Define Home	SYSTEM.POST.COMMANDS.DEFINEHOME.CONTROL	0x37200000	Position

Figure 155 - PAC List Example

19.3. User Units

Each network interface (Modbus, CANopen) have an independent option to define user defined units. This is accomplished by assigning scaling (numerator and denominator) to a variable. These conversions are transparent to the user. CANopen defines the scaling factor use base on the data type (position, velocity or acceleration), while Modbus allows any scaling factors to be use.

Internal units = (User units) * (Numerator / Denominator) Write operation
User units = (internal units) * (Denominator / Numerator) Read operation

NOTE! Thirty-two bit conversion is used and rounding error could occur between read and write values.

19.3.1. UI Modbus

Typically user will use Modbus address to read/write, by using the GID addressing system a user can scale data to/from internal units. This conversion is transparent to Modbus interface. Modbus allows any scale factory to be used to scale a variable; the following example demonstrates scaling of Target position.

Refer to Modbus user guided for more information to access GID through Modbus.

Description	Units	Att	GID	Modbus
Scale Factor 1 – Numerator	UINT32	RW	MODBUSUI.PARAM.CONVERT.0.MULTIPLIER	6930
Scale Factor 1 – Denominator	UINT32	RW	MODBUSUI.PARAM.CONVERT.0.DIVISOR	6932
Scale Factor 2 – Numerator	UINT32	RW	MODBUSUI.PARAM.CONVERT.1.MULTIPLIER	6934
Scale Factor 2 – Denominator	UINT32	RW	MODBUSUI.PARAM.CONVERT.1.DIVISOR	6936
Scale Factor 3 – Numerator	UINT32	RW	MODBUSUI.PARAM.CONVERT.2.MULTIPLIER	6938
Scale Factor 3 – Denominator	UINT32	RW	MODBUSUI.PARAM.CONVERT.2.DIVISOR	6940
Scale Factor 4 – Numerator	UINT32	RW	MODBUSUI.PARAM.CONVERT.3.MULTIPLIER	6942
Scale Factor 4 – Denominator	UINT32	RW	MODBUSUI.PARAM.CONVERT.3.DIVISOR	6944
Scale Factor 5 – Numerator	UINT32	RW	MODBUSUI.PARAM.CONVERT.4.MULTIPLIER	6946
Scale Factor 5 – Denominator	UINT32	RW	MODBUSUI.PARAM.CONVERT.4.DIVISOR	6948
Scale Factor 6 – Numerator	UINT32	RW	MODBUSUI.PARAM.CONVERT.5.MULTIPLIER	6950
Scale Factor 6 – Denominator	UINT32	RW	MODBUSUI.PARAM.CONVERT.5.DIVISOR	6952
Scale Factor 7 – Numerator	UINT32	RW	MODBUSUI.PARAM.CONVERT.6.MULTIPLIER	6954
Scale Factor 7 – Denominator	UINT32	RW	MODBUSUI.PARAM.CONVERT.6.DIVISOR	6956

Figure 156 – User Units Modbus

NOTE! Default Numerator and Denominator are 1

Name	GID	Modbus	CO
Target Position (no Scaling)	SYSTEM.MOTION.MODES.PP.COMMAND.SETPOINT.DISTANCE + 0	2406	607A.0
Target Position (Scale Factor 1)	SYSTEM.MOTION.MODES.PP.COMMAND.SETPOINT.DISTANCE + 2	- - - -	- - - -
Target Position (Scale Factor 2)	SYSTEM.MOTION.MODES.PP.COMMAND.SETPOINT.DISTANCE + 4	- - - -	- - - -
Target Position (Scale Factor 3)	SYSTEM.MOTION.MODES.PP.COMMAND.SETPOINT.DISTANCE + 6	- - - -	- - - -
Target Position (Scale Factor 4)	SYSTEM.MOTION.MODES.PP.COMMAND.SETPOINT.DISTANCE + 8	- - - -	- - - -
Target Position (Scale Factor 5)	SYSTEM.MOTION.MODES.PP.COMMAND.SETPOINT.DISTANCE + 10	- - - -	- - - -
Target Position (Scale Factor 6)	SYSTEM.MOTION.MODES.PP.COMMAND.SETPOINT.DISTANCE + 12	- - - -	- - - -
Target Position (Scale Factor 7)	SYSTEM.MOTION.MODES.PP.COMMAND.SETPOINT.DISTANCE + 14	- - - -	- - - -

Figure 157 - Example Modbus using units

NOTE! There is no direct Modbus address, of the scaled value. For example 2406 is the Modbus address of Target Position without scaling. Using scaling 1, user must use Modbus GID access portal to read/write data.

19.3.2. CANopen

CANopen associates a scale factory to a group of registers, for example scale factory 1 is always assigned to Position measurements. This conversion is transparent to CANopen interface.

Name	Units	Att	Description	GID	Modbus	CO
Scale Factor 1 – Numerator	UINT32	RW	Position	CANOPEN.PARAM.CONVERT.0.MULTIPLIER	7560	2102.1
Scale Factor 1 – Denominator	UINT32	RW		CANOPEN.PARAM.CONVERT.0.DIVISOR	7562	2102.2
Scale Factor 2 – Numerator	UINT32	RW	Velocity	CANOPEN.PARAM.CONVERT.1.MULTIPLIER	7564	2103.1
Scale Factor 2 – Denominator	UINT32	RW		CANOPEN.PARAM.CONVERT.1.DIVISOR	7566	2103.2
Scale Factor 3 – Numerator	UINT32	RW	Acceleration	CANOPEN.PARAM.CONVERT.2.MULTIPLIER	7568	2104.1
Scale Factor 3 – Denominator	UINT32	RW		CANOPEN.PARAM.CONVERT.2.DIVISOR	7570	2104.2

Figure 158 – User Units CANopen

NOTE! Numerator and Denominator defaults are 1

19.4. System Layout

To gain a better understanding of the system, below are highlights of the system organization.

- During start-up initialization, all non-volatile memory data is copied to *RAM* where it is made available to interfaces.
- The writing of parameter register data, however, modifies only the current values in *RAM*. Future power-ups will again initialize the *RAM* parameter blocks to their original values retained in non-volatile memory. The copying (saving) of parameter *RAM* blocks to non-volatile memory is carried out only when specified through direct control commands
- User parameters are stored as a block in non-volatile memory. The block contains a *CRC* (*Cyclic Redundancy Checksum*) word to guarantee data integrity. At power-up, the user parameter block is validated and copied to its runtime location in *RAM* where all parameters are available for both reading and writing.
- The drive firmware is design in a module method, allowing each module to act independent but controlled from superior control. This allows the system to send commands to active module motion module (Jog or Position, etc), without sacrificing process time.
- Modules (Motion, Controls) could have two sets of registers Run time and Initialize registers. Initialize registers are used only at initialization of the drive or in some cases the Initialize of motion mode. Run time registers are the current variables that the module will use its operation.
- The below figure illustrates how controlword (6040h) interface to active Motion mode Profile Velocity; reference Profile Velocity for more information. When clearing or setting a Controlword bit it feeds sends the updated bit pattern to the active motion mode, while the active mode updates the statusword of its current state. Do to the limitation of DS402, not all motion control bits are supported by the controlword directly. Reference each mode to determine which functions are available.

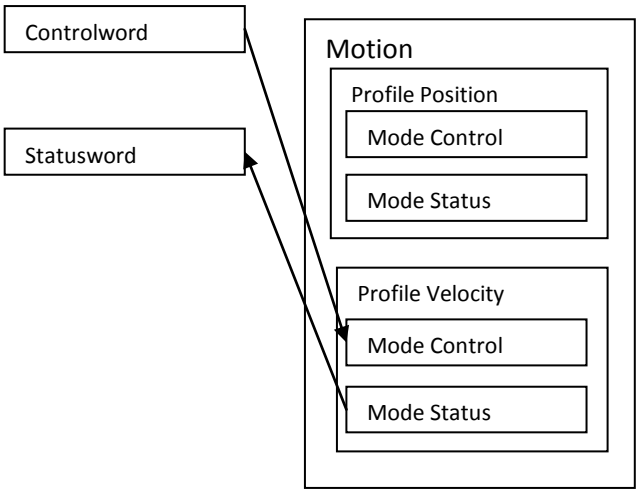


Figure 159 - Control/Status Layout

19.5. Notations

0.001 rated	Percentage of Rated (GID=SYSTEM.BUS.FACTORYPARAM.IRATED) *(CANopen -6075.0, Motor Rated Current)*
mrps/s	Acceleration Internal units
mrps	Velocity Internal units
0.0001 rev	Distance internal units
ms	Milliseconds

Figure 160 - Units

(b)	When read/write from Modbus data length is word, while from CANopen it is a byte.
(M)	CANopen Object is mappable
(d)	CANopen Profile –Motion DS402 specification name

Figure 161 – Superscript

20. CANopen Object Dictionary

20.1. Communication – DS301

Index	Description	Type	Att	Map	GID
1000.0	Device Type	UINT32	RO		CANOPEN.IDENTITY.DEVICETYPE
1001.0	Error Register	UINT8	RO		CANOPEN.STATUS.ERRORREGISTER
1005.0	COB-ID Sync	UINT32	RW		CANOPEN.PARAM.SYNCCOBID
100C.0	Guard Time	UINT16	RW		CANOPEN.PARAM.NODEGUARDTIME
100D.0	Life Time Factor	UINT8	RW		CANOPEN.PARAM.LIFETIMEFACTOR
1010	Save Parameters	-	-		-
1010.1	All	UINT32	RW		COMMANDS.SAVEUSER *(generates internal Pac)*
1010.2	CANopen	UINT32	RW		CANOPEN.PARAM *(generates internal Pac)*
1014.0	COB-ID EMCY	UINT32	RW		CANOPEN.PARAM.EMERGENCYCOBID
1015.0	Inhibit time EMCY	UINT16	RW		CANOPEN.PARAM.EMERGENCYINHIBITTIMER
1017.0	Produce Heartbeat timer	UINT16	RW		CANOPEN.PARAM.HEARTBEATPRODUCETIME
1018	Identity	-	-		-
1018.1	Vendor ID	UINT32	RO		CANOPEN.IDENTITY.VENDORID
1018.2	Product Code	UINT16	RO		SYSTEM.ID.ID.PRODUCTID
1400	RPDO1 COMM. parameter	-	-		
1400.1	COB-ID	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.0.COMM.COBID
1400.2	Trans type	UINT8	RW		CANOPEN.PARAM.PDORECEIVE.0.COMM.TRANMISSIONTYPE
1401	RPDO2 COMM. parameter	-	-		-
1401.1	COB-ID	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.1.COMM.COBID
1401.2	Trans type	UINT8	RW		CANOPEN.PARAM.PDORECEIVE.1.COMM.TRANMISSIONTYPE
1402	RPDO3 COMM. Parameter	-	-		-
1402.1	COB-ID	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.2.COMM.COBID
1402.2	Trans type	UINT8	RW		CANOPEN.PARAM.PDORECEIVE.2.COMM.TRANMISSIONTYPE
1403	RPDO4 COMM parameter	-	-		
1403.1	COB-ID	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.3.COMM.COBID
1403.3	Trans type	UINT8	RW		CANOPEN.PARAM.PDORECEIVE.3.COMM.TRANMISSIONTYPE
1600	RPDO1 mapping parameter	-	-		
1600.0	Num of Entries	UINT8	RW		CANOPEN.PARAM.PDORECEIVE.0.MAPPING.ENTRIES
1600.1	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.0.MAPPING.MAP.0
1600.2	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.0.MAPPING.MAP.1
1600.3	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.0.MAPPING.MAP.2
1600.4	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.0.MAPPING.MAP.3
1600.5	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.0.MAPPING.MAP.4
1600.6	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.0.MAPPING.MAP.5
1600.7	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.0.MAPPING.MAP.6
1600.8	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.0.MAPPING.MAP.7
1601	RPDO2 mapping parameter	-	-		
1601.0	Num of Entries	UINT8	RW		CANOPEN.PARAM.PDORECEIVE.1.MAPPING.ENTRIES
1601.1	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.1.MAPPING.MAP.0
1601.2	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.1.MAPPING.MAP.1
1601.3	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.1.MAPPING.MAP.2
1601.4	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.1.MAPPING.MAP.3
1601.5	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.1.MAPPING.MAP.4
1601.6	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.1.MAPPING.MAP.5
1601.7	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.1.MAPPING.MAP.6
1601.8	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.1.MAPPING.MAP.7

Index	Description	Type	Attr	Map	GID
1602	RPDO3 mapping parameter	-	-		-
1602.0	Num of Entries	UINT8	RW		CANOPEN.PARAM.PDORECEIVE.2.MAPPING.ENTRIES
1602.1	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.2.MAPPING.MAP.0
1602.2	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.2.MAPPING.MAP.1
1602.3	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.2.MAPPING.MAP.2
1602.4	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.2.MAPPING.MAP.3
1602.5	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.2.MAPPING.MAP.4
1602.6	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.2.MAPPING.MAP.5
1602.7	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.2.MAPPING.MAP.6
1602.8	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.2.MAPPING.MAP.7
1603	RPDO4 mapping parameter	-	-		-
1603.0	Num of Entries	UINT8	RW		CANOPEN.PARAM.PDORECEIVE.3.MAPPING.ENTRIES
1603.1	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.3.MAPPING.MAP.0
1603.2	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.3.MAPPING.MAP.1
1603.3	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.3.MAPPING.MAP.2
1603.4	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.3.MAPPING.MAP.3
1603.5	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.3.MAPPING.MAP.4
1603.6	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.3.MAPPING.MAP.5
1603.7	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.3.MAPPING.MAP.6
1603.8	parameter	UINT32	RW		CANOPEN.PARAM.PDORECEIVE.3.MAPPING.MAP.7
1800	TPDO1 COMM-parameter	-	-		-
1800.1	COB-ID	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.0.COMM.COBID
1800.2	Trans type	UINT8	RW		CANOPEN.PARAM.PDOTRANSMIT.0.COMM.TRANMISSIONTYPE
1800.3	Inhibit timer	UINT16	RW		CANOPEN.PARAM.PDOTRANSMIT.0.COMM.INHIBITTIME
1800.4	Event Timer	UINT16	RW		CANOPEN.PARAM.PDOTRANSMIT.0.COMM.EVENTTIMER
1801	TPDO2 COMM-parameter	-	-		-
1801.1	COB-ID	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.1.COMM.COBID
1801.2	Trans type	UINT8	RW		CANOPEN.PARAM.DOTRANSMIT.1.COMM.TRANMISSIONTYPE
1801.3	Inhibit timer	UINT16	RW		CANOPEN.PARAM.PDOTRANSMIT.1.COMM.INHIBITTIME
1801.4	Event Timer	UINT16	RW		CANOPEN.PARAM.PDOTRANSMIT.1.COMM.EVENTTIMER
1802	TPDO3 COMM-parameter	-	-		-
1802.1	COB-ID	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.2.COMM.COBID
1802.2	Trans type	UINT8	RW		CANOPEN.PARAM.PDOTRANSMIT.2.COMM.TRANMISSIONTYPE
1802.3	Inhibit timer	UINT16	RW		CANOPEN.PARAM.PDOTRANSMIT.2.COMM.INHIBITTIME
1802.4	Event Timer	UINT16	RW		CANOPEN.PARAM.PDOTRANSMIT.2.COMM.EVENTTIMER
1803	TPDO4 COMM-parameter	-	-		-
1803.1	COB-ID	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.3.COMM.COBID
1803.2	Trans type	UINT8	RW		CANOPEN.PARAM.PDOTRANSMIT.3.COMM.TRANMISSIONTYPE
1803.3	Inhibit timer	UINT16	RW		CANOPEN.PARAM.PDOTRANSMIT.3.COMM.INHIBITTIME
1803.4	Event Timer	UINT16	RW		CANOPEN.PARAM.PDOTRANSMIT.3.COMM.EVENTTIMER
1A00	TPDO1 mapping parameter	-	-		-
1A00.0	Number of Entries	UINT8	RW		CANOPEN.PARAM.PDOTRANSMIT.0.MAPPING.ENTRIES
1A00.1	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.0.MAPPING.MAP.0
1A00.2	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.0.MAPPING.MAP.1
1A00.3	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.0.MAPPING.MAP.2
1A00.4	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.0.MAPPING.MAP.3
1A00.5	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.0.MAPPING.MAP.4
1A00.6	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.0.MAPPING.MAP.5
1A00.7	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.0.MAPPING.MAP.6
1A00.8	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.0.MAPPING.MAP.7
1A01	TPDO2 mapping parameter	-	-		-
1A01.0	Num of Entries	UINT8	RW		CANOPEN.PARAM.PDOTRANSMIT.1.MAPPING.ENTRIES
1A01.1	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.1.MAPPING.MAP.0

1A01.2	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.1.MAPPING.MAP.1
1A01.3	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.1.MAPPING.MAP.2
1A01.4	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.1.MAPPING.MAP.3
1A01.5	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.1.MAPPING.MAP.4
1A01.6	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.1.MAPPING.MAP.5
1A01.7	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.1.MAPPING.MAP.6
1A01.8	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.1.MAPPING.MAP.7
1A02	TPDO3 mapping parameter		-		-
1A02.0	Num of Entries	UINT8	RW		CANOPEN.PARAM.PDOTRANSMIT.2.MAPPING.ENTRIES
1A02.1	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.2.MAPPING.MAP.0
1A02.2	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.2.MAPPING.MAP.1
1A02.3	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.2.MAPPING.MAP.2
1A02.4	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.2.MAPPING.MAP.3
1A02.5	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.2.MAPPING.MAP.4
1A02.6	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.2.MAPPING.MAP.5
1A02.7	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.2.MAPPING.MAP.6
1A02.8	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.2.MAPPING.MAP.7
1A03	TPDO4 mapping parameter	-	-		-
1A03.0	Num of Entries	UINT8	RW		CANOPEN.PARAM.PDOTRANSMIT.3.MAPPING.ENTRIES
1A03.1	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.3.MAPPING.MAP.0
1A03.2	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.3.MAPPING.MAP.1
1A03.3	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.3.MAPPING.MAP.2
1A03.4	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.3.MAPPING.MAP.3
1A03.5	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.3.MAPPING.MAP.4
1A03.6	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.3.MAPPING.MAP.5
1A03.7	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.3.MAPPING.MAP.6
1A03.8	parameter	UINT32	RW		CANOPEN.PARAM.PDOTRANSMIT.3.MAPPING.MAP.7

Figure 162 - CANopen communication Objects

20.2. Manufacture

Index	Description	Type		MAP	GID
2000.0	CAN Node ID	UINT8	RW	- - -	CANOPEN.PARAM.ID
2001.0	CAN Baud rate	UINT16	RW	- - -	CANOPEN.PARAM.BAUD
2002.1	System Access Read- GID	UINT32	RW	- - -	CANOPEN.STATUS.GIDREAD
2002.2	System Access Read- Data	UINT32	RW	- - -	
2003.1	System Access Write - GID	UINT32	RW	- - -	CANOPEN.STATUS.GIDWRITE
2003.2	System Access Write - Data	UINT32	RW	- - -	
2100.1	Control Access Request-Control	UINT16	RW	- - -	CANOPEN.PARAM.CMDCONTROL.CONTROL
2100.2	Control Access Request Exclusive	UINT16	RW	- - -	CANOPEN.PARAM.CMDCONTROL.EXCLUSIVE
2101.1	Control Access Actual Control	UINT16	RW	- - -	CANOPEN.CMDCONTROL.EXCLUSIVE
2101.1	Control Access Actual Exclusive	UINT16	RW	- - -	CANOPEN.CMDCONTROL.EXCLUSIVE
2102.1	Scale Factor 1 – Multiply(s)	UINT32	RW	- - -	GID.CANOPEN.PARAM.CONVERT.0.MULTIPLIER
2102.2	Scale Factor 1 – Divisor(s)	UINT32	RW	- - -	GID.CANOPEN.PARAM.CONVERT.0.DIVISOR
2103.1	Scale Factor 2 – Multiply(s)	UINT32	RW	- - -	GID.CANOPEN.PARAM.CONVERT.1.MULTIPLIER
2103.2	Scale Factor 2 – Divisor(s)	UINT32	RW	- - -	GID.CANOPEN.PARAM.CONVERT.1.DIVISOR
2104.1	Scale Factor 3 – Multiply(s)	UINT32	RW	- - -	GID.CANOPEN.PARAM.CONVERT.2.MULTIPLIER
2104.2	Scale Factor 3 – Divisor(s)	UINT32	RW	- - -	GID.CANOPEN.PARAM.CONVERT.2.DIVISOR
2109.0	Drive Command Access	UINT32	RW	- - -	CANOPEN.CMDCOMMAND.GID
2110.1	CANopen Status - Faults	UINT16	RO	(M)	CANOPEN.EVENTS.FAULTS
2110.2	CANopen Status - Warnings	UINT16	RO	(M)	CANOPEN.EVENTS.WARNINGS
3000.0	Drive Faults	UINT16	RO	(M)	EVENTS.FAULTS
3001.0	Drive Warnings	UINT16	RO	(M)	EVENTS.WARNINGS
3032.0	System – Faults	UINT16	RO	(M)	SYSTEM.EVENTS.FAULTS
3033.0	System -Warnings	UINT16	RO	(M)	SYSTEM.EVENTS.WARNINGS
3064.0	ID - Faults	UINT16	RO	(M)	SYSTEM.ID.EVENTS.FAULTS
3065.0	ID -Warnings	UINT16	RO	(M)	SYSTEM.ID.EVENTS.WARNINGS
3258.0	Digital IO - Faults	UINT16	RO	(M)	SYSTEM.DIGITAL.EVENTS.FAULTS
3259.0	Digital IO -Warnings	UINT16	RO	(M)	SYSTEM.DIGITAL.EVENTS.WARNINGS
312C.0	Control - Faults	UINT16	RO	(M)	SYSTEM.CONTROL.EVENTS.FAULTS
312D.0	Control -Warnings	UINT16	RO	(M)	SYSTEM.CONTROL.EVENTS.WARNINGS
3190.0	Bus - Faults	UINT16	RO	(M)	SYSTEM.BUS.EVENTS.FAULTS
3191.0	Bus -Warnings	UINT16	RO	(M)	SYSTEM.BUS.EVENTS.WARNINGS
3192.0	Bus - Status	UINT16	RO	(M)	SYSTEM.BUS.EVENTS.STATUS
31F4.0	Position Tracking-Faults	UINT16	RO	(M)	SYSTEM.POST.EVENTS.FAULTS
31F5.0	Position Tracking-Warnings	UINT16	RO	(M)	SYSTEM.POST.EVENTS.WARNINGS
31F6.0	Position Tracking- Status	UINT16	RO	(M)	SYSTEM.POST.EVENTS.STATUS
3384.0	Motion - Faults	UINT16	RO	(M)	SYSTEM.MOTION.EVENTS.FAULTS
3385.0	Motion -Warnings	UINT16	RO	(M)	SYSTEM.MOTION.EVENTS.WARNINGS
3386.0	Motion - Status	UINT16	RO	(M)	SYSTEM.MOTION.EVENTS.STATUS
3320.0	History Log - Faults	UINT16	RO	(M)	SYSTEM.LOG.EVENTS.FAULTS
3321.0	History Log - -Warnings	UINT16	RO	(M)	SYSTEM.LOG.EVENTS.WARNINGS
3322.0	History Log - Status	UINT16	RO	(M)	SYSTEM.LOG.EVENTS.STATUS
32BC.0	Thermal - Faults	UINT16	RO	(M)	SYSTEM.THERMAL.EVENTS.FAULTS
32BD.0	Thermal -Warnings	UINT16	RO	(M)	SYSTEM.THERMAL.EVENTS.WARNINGS
32BE.0	Thermal - Stress	UINT16	RO	(M)	SYSTEM.THERMAL.EVENTS.STRESS

Index	Description	Type		MAP	GID
30C8.0	NVM - Faults	UINT16	RO	(M)	SYSTEM.NVM.EVENTS.FAULTS
30C9.0	NVM - Warnings	UINT16	RO	(M)	SYSTEM.NVM.EVENTS.WARNINGS
30CA.0	NVM - Status	UINT16	RO	(M)	SYSTEM.NVM.EVENTS.STATUS
4001.0	Enable Actions	UINT16	RO	- - -	SYSTEM.CONTROL.USERPARAM.ENABLEACTION
4002.0	Current Window	UINT16	RW	- - -	SYSTEM.MOTION.USERPARAM.WINDOWS.ITARGET
4003.0	Current Window time	UINT16	RW	- - -	SYSTEM.MOTION.USERPARAM.WINDOWS.ITARGETTIME

Figure 163 - CANopen Manufacture Objects

20.3. Motion Profile - DS402

Index	Description[a]	Type	Att.	Map	Scale[s] Factor	GID
6040.0	Controlword	UINT16	RW	(M)	- - -	(PAC's)
6041.0	Statusword	UINT16	RO	(M)	- - -	SYSTEM.CONTROL.EVENTS.STATUS
605A.0	Quick Stop Code	INT16	RW		- - -	SYSTEM.CONTROL.COMMAND.STOPACTION
605E.0	Fault reaction Code	INT16	RW		- - -	SYSTEM.CONTROL.COMMAND.FAULTACTION
6060.0	Mode of Operation	INT8	RW	(M)	- - -	SYSTEM.MOTION.COMMAND.REQUESTEDMODE
6061.0	Mode of Operation Actual	INT8	RO	(M)	- - -	SYSTEM.MOTION.MONITOR.ACTIVEMODE
6062.0	Position Demand value	INT32	RO	(M)	1	SYSTEM.MOTION.MODES.PP.SETPOINTDEBUG.DISTANCE
6064.0	Position Actual Value	INT32	RO	(M)	1	SYSTEM.POST.MONITOR.POSITION
6065.0	Following Error Window	INT32	RW		1	SYSTEM.MOTION.USERPARAM.WINDOWS.PERROR
6066.0	Following Error Time out	UINT16	RW		- - -	SYSTEM.MOTION.USERPARAM.WINDOWS.PERRORTIME
6067.0	Position Window	INT32	RW		1	SYSTEM.MOTION.USERPARAM.WINDOWS.PZERO
6068.0	Position Window Time	UINT16	RW		- - -	SYSTEM.MOTION.USERPARAM.WINDOWS.PZEROTIME
606B.0	Velocity Demand Value	INT32	RO	(M)	2	SYSTEM.MOTION.MODES.PV.MONITOR.TARGET
606C.0	Velocity actual value	INT32	RW	(M)	2	SYSTEM.POST.MONITOR.VELOCITY
606D.0	Velocity Window	UINT16	RW		2	SYSTEM.MOTION.USERPARAM.WINDOWS.VTARGET
606E.0	Velocity Window Time	UINT16	RW		- - -	SYSTEM.MOTION.USERPARAM.WINDOWS.VTARGETTIME
606F.0	Velocity Threshold	INT16	RW		2	SYSTEM.MOTION.USERPARAM.WINDOWS.VZERO
6070.0	Velocity Threshold Time	UINT16	RW	(M)	- - -	SYSTEM.MOTION.USERPARAM.WINDOWS.VTARGETTIME
6071.0	Target Torque	INT16	RW		- - -	SYSTEM.MOTION.MODES.PT.COMMAND.TARGET
6073.0	Max Current	UINT16	RW		- - -	SYSTEM.MOTION.USERPARAM.LIMITS.CURRENT
6074.0	Torque Demand Value	INT16	RO		- - -	SYSTEM.MOTION.MODES.PT.MONITOR.COMMAND
6075.0	Motor Rated Current	UINT32	R-		- - -	SYSTEM.BUS.FACTORYPARAM.IRATED
6077.0	Torque Actual Value	INT16	RO	(M)	- - -	SYSTEM.MOTION.CONTROL.ILOOP.TFEEDBACK
6078.0	Current Actual Value	INT16	RO		- - -	SYSTEM.BUS.MONITOR.CURRENT
607C.0	Home offset	UIN32		(M)	1	SYSTEM.MOTION.MODES.HOME.PARAM.OFFSET
607A.0	Target Position	UINT32	RW	(M)	1	SYSTEM.MOTION.MODES.PP.COMMAND. SETPOINT.DISTANCE
607F.0	Max Profile Velocity	INT16	RW		2	SYSTEM.MOTION.USERPARAM.LIMITS.VELOCITY
6081.0	Profile Velocity	UINT32	RW	(M)	2	SYSTEM.MOTION.MODES.PP.COMMAND. SETPOINT.VTARGET
6082.0	End Velocity	UINT32	RW	(M)	2	SYSTEM.MOTION.MODES.PP.COMMAND.SETPOINT.VFIN AL
6083.0	Profile Acceleration	UINT32	RW	(M)	3	SYSTEM.MOTION.COMMAND.PROFILE.ACCEL
6084.0	Profile Declaration	UINT32	RW	(M)	3	SYSTEM.MOTION.COMMAND.PROFILE.DECEL
6085.0	Quick stop deceleration	UINT32	RW		3	SYSTEM.CONTROL.COMMAND.STOPDECEL
6086.0	Motion Profile Type	INT16	RW		- - -	SYSTEM.MOTION.COMMAND.PROFILE.TYPE
6087.0	Target Slope	UINT32	RW	(M)	3	SYSTEM.MOTION.MODES.PT.COMMAND.ACCEL
6098.0	Homing Method	INT8	RW	(M)	- - -	SYSTEM.MOTION.MODES.HOME.PARAM.METHOD
6099.0	Homing Speed	-	-	-	- - -	-
6099.1	During search for switch	INT32	RW	(M)	2	SYSTEM.MOTION.MODES.HOME.PARAM.VFAST
6099.2	During search for zero	INT32	RW	(M)	2	SYSTEM.MOTION.MODES.HOME.PARAM.VSLOW
609A.0	Homing Acceleration	UINT32	RW		3	SYSTEM.MOTION.MODES.HOME.PARAM.ACCEL
60FD.0	Digital inputs	UINT16	RO	(M)	- - -	SYSTEM.DIGITAL.EVENTS.INPUTS
60FE.0	Digital outputs	UINT16	RO	(M)	- - -	SYSTEM.DIGITAL.EVENTS.OUTPUTS
60FF.0	Target velocity	INT32	RW	(M)	2	SYSTEM.MOTION.MODES.PV.COMMAND.TARGET

Figure 164 - CANopen Profile DS402 Objects

[s] User defined units (reference section 18.3 for more information)

21. Figures