GLENTEK INC.

SERVO DRIVES AND SERVO MOTORS

EtherCAT

Gamma EtherCAT Manual

1st Edition

www.glentek.com September 2022 – V.1.0

Preface

Glentek was founded in Glendora, California in 1964. One year later, we incorporated and moved to Santa Monica, California. In the beginning, our company designed, and manufactured servo drives and related electronics for the defense industry, including applications involving advanced tactical aircraft and ship-based radar systems. During the 1970's we began production on some of the first PWM servo drives for the growing CNC machinery market. Further growth resulted in another move in 1980, this time to our current location in El Segundo, California.

We expanded into the medical market, producing custom servo drives for both CAT Scan and PET scan machines. To provide the highest performance and reliability possible, we began designing and manufacturing a complete line of servo motors to accompany our servo drives in the mid 1980's, providing a complete package for OEMs. Our first foray in DSP-based drives and the development of MotionMaestro, our Windows-based set-up and tuning software, occurred in the 1990's. Since then, we have released new drives capable of communicating using the latest protocols and standards, first Synqnet, then CANopen, and now EtherCAT. Our latest drives also accept feedback from the latest devices, including analog sin/cos encoders and absolute serial encoders.

For more information and details on our products, visit the company's website at www.glentek.com.

Trademarks

EtherCAT® is registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.

Contents

Tables and Figures

1. Scope

EtherCAT is a real-time Industrial Ethernet technology originally developed by Beckhoff Automation. The EtherCAT protocol which is disclosed in the IEC standard IEC61158 is suitable for hard and soft real-time requirements in automation technology, in test and measurement and many other applications.

The focus during the development of EtherCAT was on short cycle times (≤ 100 µs), low jitter for accurate synchronization (≤ 1 µs) and low hardware costs.

EtherCAT was introduced in April 2003, and the EtherCAT Technology Group was founded in November 2003 - Meanwhile ETG has grown into the world's largest Industrial Ethernet and fieldbus organization. The ETG brings together manufacturers and users, which contribute to technical working groups to the advancement of the EtherCAT technology [1] .

Glentek's EtherCAT drives follow CiA402 drive profile that is mapped to EtherCAT is specified in IEC 61800-7-201 and IEC 61800-7-301. In the IEC Standard many objects and operation modes are defined as optional. This document intends to define a common behavior of an EtherCAT servo drive supporting the CiA402 drive profile.

For further technical questions, please visit our website at www.glentek.com or call us at (310) 322-3026

2. Terms and Abbreviations

AL- Application Layer
ADO- Address Offset
CoE- CAN application protocol over EtherCAT
CSP- Cyclic Synchronous Position
CSV- Cyclic Synchronous Velocity
CST- Cyclic Synchronous Torque
DC- Distributed Clocks
DL- Data Link Layer
ENI- EtherCAT Network Information
ESC- EtherCAT Slave Controller
ESI- EtherCAT Slave Information
ESM- EtherCAT State Machine
ETG- EtherCAT Technology Group
FCS- Frame Check Sequence
HM- Homing Mode
OD- Object Dictionary
PDO- Process data objects
RO- Read Only
RW- Read and Write
SDO- Service data objects
SM- SyncManager
WKC- Working Counter
VL- Velocity mode

3. Introduction

3.1. The EtherCAT protocol

The EtherCAT is a protocol that is based on Ethernet frame according to IEEE 802.3 as shown in Figure 3 - 1. The EtherCAT frame has a header for datagrams and one or more datagrams. The EtherCAT header has the information about the EtherCAT type. The datagram has the datagram header, data, and working counter (WKC). The datagram header indicates commands (read/write), address information, and data length. The working counter is incremented after each read/write access for checking EtherCAT network [1] .

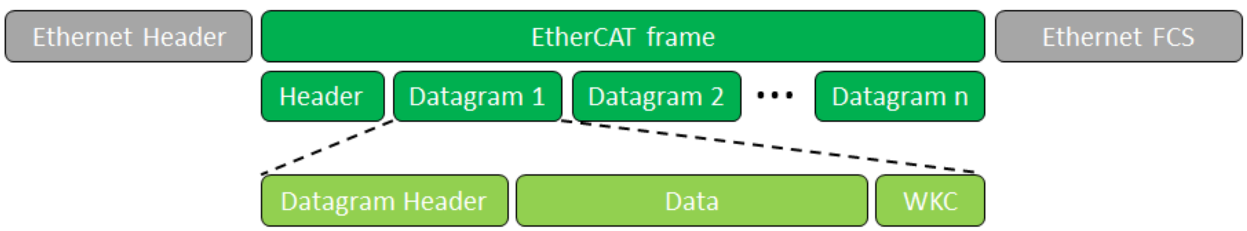


Figure 3 - 1. EtherCAT in a standard Ethernet frame (according to IEEE 802.3)

The cyclic and acyclic data can be extracted and inserted on the fly when the EtherCAT frame is transmitted on the network as shown in Figure 3 - 2 [2] .

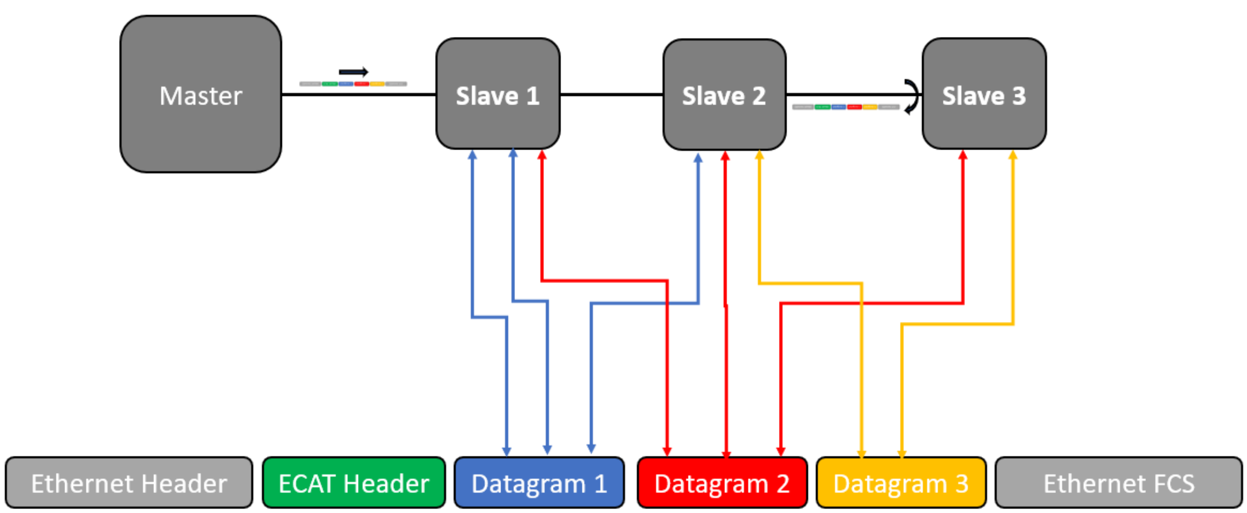


Figure 3 - 2. Inserting data on the fly

3.2. EtherCAT System Architecture

The basic EtherCAT system architecture is shown in Figure 3 - 3. The EtherCAT master uses a standard Ethernet port and network configuration information stored in the EtherCAT Network Information (ENI) file. The ENI is created based on the EtherCAT Slave Information (ESI) file which are provided by the vendors for each device. Slaves are connected via Ethernet. Any topology type is possible for EtherCAT networks. The ESI can be converted and downloaded into the EEPROM at slaves through the network [3] .

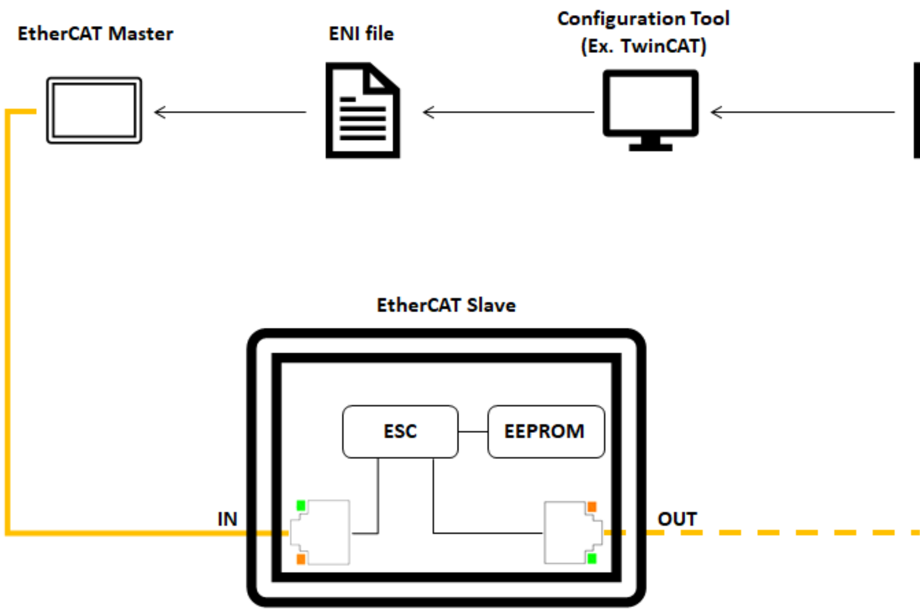


Figure 3 - 3. EtherCAT System Architecture

3.3. EtherCAT State Machine

The slave runs a state machine to indicate which functionalities are available. This EtherCAT State Machine (ESM) is shown in Figure 3 - 4. ESM requests are written by the master to the slave's AL Control register in the ESC. If the configuration for the requested state is valid, the slave acknowledges the state by setting the AL Status register. If not, the slave sets the error flag in the AL Status register and writes an error code to the AL Status Code register [3] .

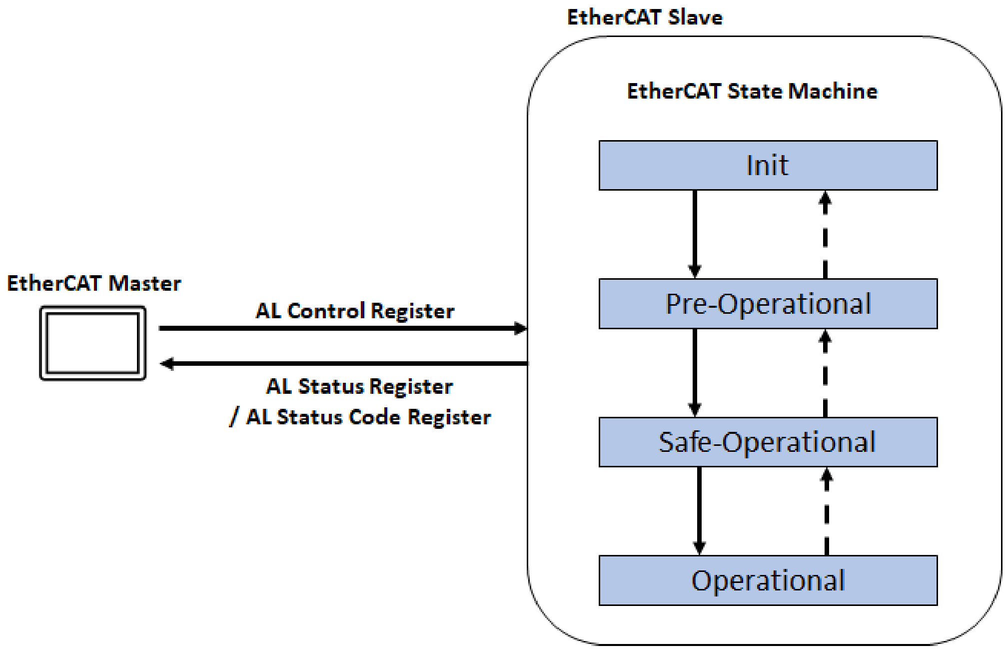


Figure 3 - 4. EtherCAT Slave State Machine

The states are described in Table 3 - 1.

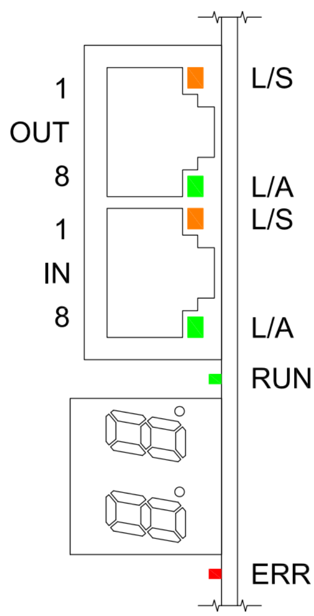
Table 3 - 1. EtherCAT State Machine Description

State	Available Functions
Init (INIT)	Init state. No communication on the application layer is available. The master has access only to the DL-information registers.
Pre-Operational (PREOP)	Pre-Operational state. Mailbox communication on the application layer available, but no process data communication available.
Safe-Operational (SAFEOP)	Safe-Operational state. Mailbox communication on the application layer, process (input) data communication available. In SAFEOP only inputs are evaluated; outputs are kept in 'safe' state.
Operational (OP)	Operational state. Process data inputs and outputs are valid.

4. Connecting to The Glentek EtherCAT Drive

4.1. EtherCAT Connections

Two EtherCAT ports on the front panel of Gamma are supported for IN port and OUT port. The straight connection (CAT5 or higher) is used and connected as daisy chain. The IN port is connected from Master side and the OUT port is connected to network side. The communication rate is up to 100 Mbit/s and the maximum length between nodes on the network is 100 [m] (=328[ft]). The port utilizes RJ45 type connector. The EtherCAT port of a Gamma drive is Illustrated in Figure 4 - 1. Table 4 - 1 shows the RJ45 Pin designations and Figure 4 - 2 shows the EtherCAT Connection Diagram.



LED	Indication	EtherCAT State
RUN (GREEN)	Off	Init
	Blinking	Pre-Operational
	Single Flash	Safe-Operational
	Flashes	Initialization or Bootstrap
	On	Operational
ERR (RED)	Off	No Error
	Blinking	Invalid Configuration
	Single Flash	Unsolicited State Change
	Double Flash	Application Watchdog Timeout
	On	PDI Watchdog Timeout
L/S (ORANGE)	Off	Not Connected
	On	Connected
L/A (GREEN)	Off	Not Connected
	On	Connected
	Flickering	Communication Active

Figure 4 - 1. The EtherCAT port of the Gamma Drive (RJ45)

EtherCAT Activity of Gamma drive is indicated by LEDs, which indicate the

- Current state of the state machine: INIT, PREOP, SAFEOP, OP (RUN LED)
- Error code (ERR LED)
- Link/Activity of the ports (L/A LED)
- Link/Status of the ports (L/S LED)

Table 4 - 1. RJ45 EtherCAT Communication Designations

Pin #	Signal	Name
1	TX+	Transmission Data +
2	TX-	Transmission Data -
3	RX+	Receive Data +
4	C1	Common for Tx (optional)
5	C2	Common for Rx (optional)
6	RX-	Receive Data -
7	NC	Not Connected
8	GND	Ground

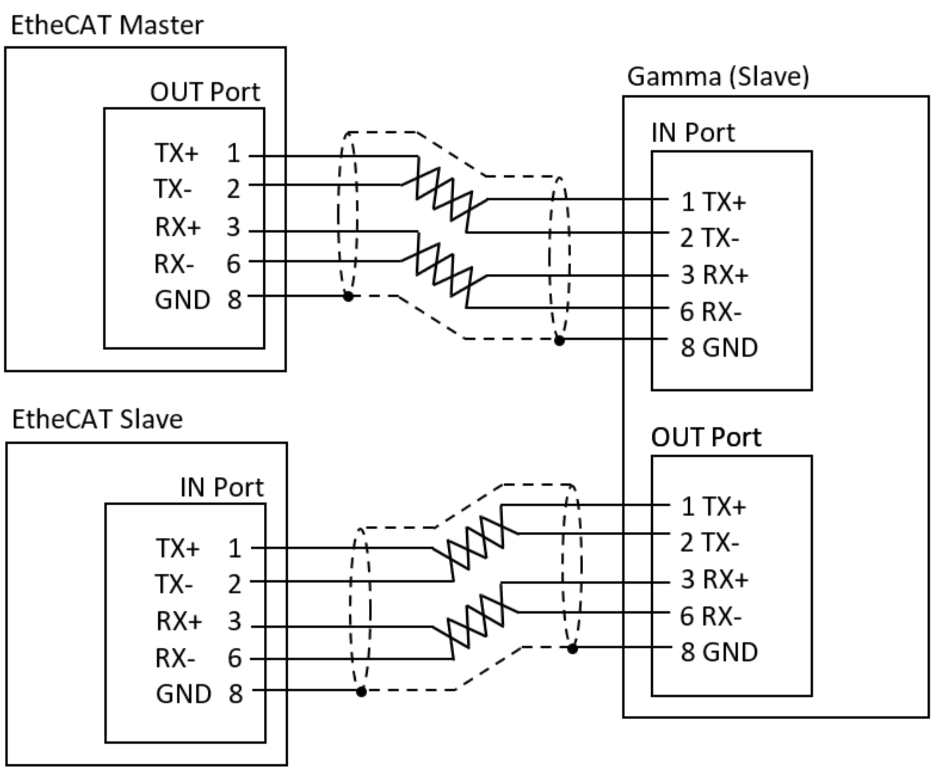


Figure 4 - 2. The EtherCAT Connection Diagram

4.2. Explicit Device Identification [7]

Normally EtherCAT slaves are automatically assigned addresses. Glentek's EtherCAT drive supports the explicit device identification for a special purpose as preventing against cable swapping.

• Prevention against cable swapping if at least two identical devices are used in one application it might be necessary to prevent the mix-up of these devices by cable swapping.
Example Scenario: Within a machining center there might be two identical drives to work in X and Y direction. To avoid that the drives receive wrong process data, for example after a device replacement, an explicit identification of the devices can be used.

The Device Identification value can be used optionally for unique addressing.

4.2.1. EtherCAT ID Selector

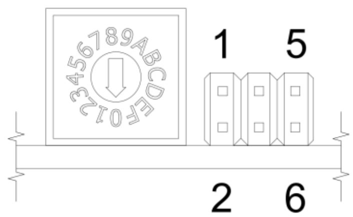


Figure 4 - 3. EtherCAT ID Selector (Rotary SW and Jumper)

The valid range of EtherCAT ID addresses extend from 1-127. The EtherCAT ID of the Gamma drive can be assigned via the 16-bit rotary switch and the address select jumpers located beside.

$$\text{EtherCAT ID} = \textit{Rotary Value} + 16 \times \textit{Jumper Position}$$

The Jumper Position uses the scheme outlined in Figure 4 - 3. For example, setting the Jumper Position = 2 and the Rotary Value = Eh will set the EtherCAT ID to be 46. Special care must be taken so that the chosen EtherCAT ID is unused by other slaves on the EtherCAT bus and not equal to zero. After power cycling, the gamma drive will assume the configured EtherCAT ID.

Table 4 - 2. EtherCAT ID Selection

Node-ID #	Rotary Value	Jumper Position
0-15	0h-Fh	0
16-31	0h-Fh	1
32-47	0h-Fh	2
48-63	0h-Fh	3
64-79	0h-Fh	4
80-95	0h-Fh	5
96-111	0h-Fh	6
112-127	0h-Fh	7

4.2.2. Requesting EtherCAT ID by Master

The EtherCAT Master can request the EtherCAT ID value by setting Bit 5 of AL Control register (0x0120.5 = ID Request) or setting the value of Explicit Device Identification (ADO 0x0134) by TwinCAT as Figure 4 - 4 after checking “Box 2 (9GEXX)\EtherCAT\Advanced Settings\General\ Behavior\Check Identification”

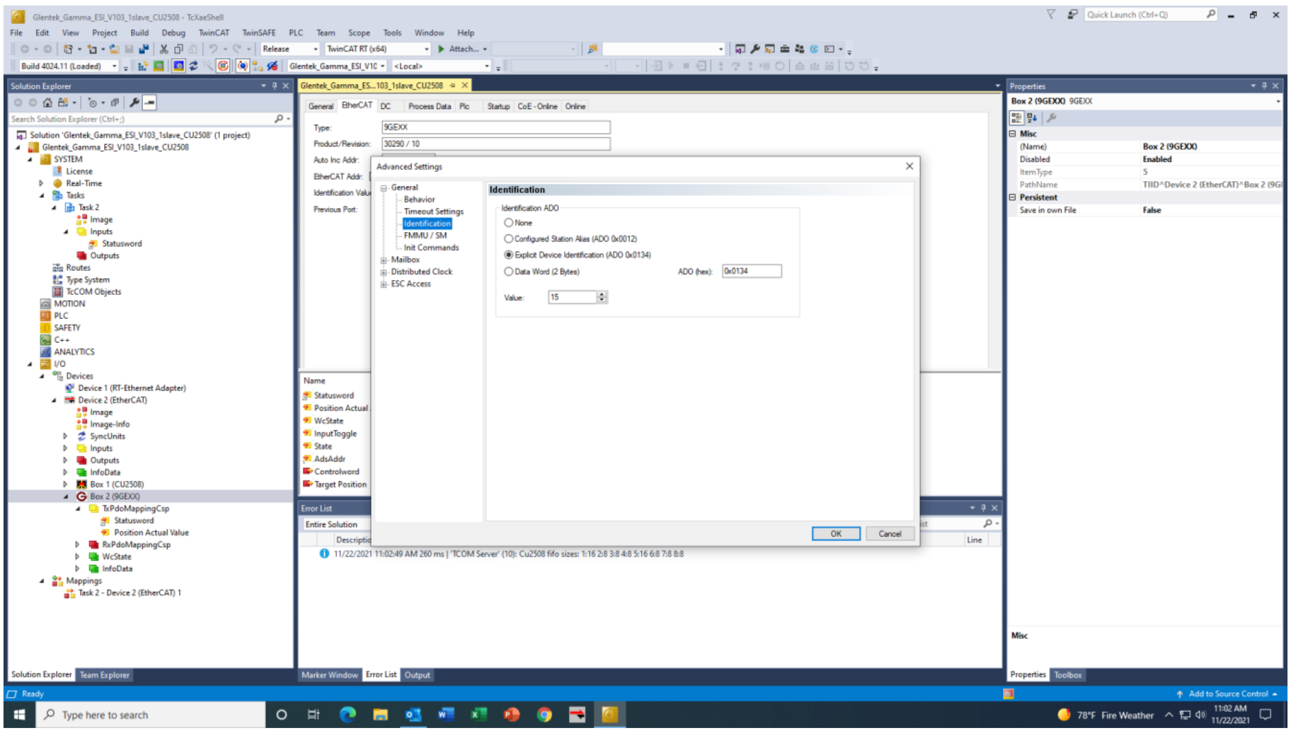


Figure 4 - 4. Set Explicit Device Identification by TwinCAT

5. Communication Objects

5.1. Object 1000h: Device Type

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
1000h	00h	Device Type	00020192h	No	Unsigned 32-bit	RO	No

This object gives information about the device type and its functionality.

Table 5 - 1. Definition of 1000h

Bit	Meaning
0-15 (16 bits)	Device Profile: 402 (0192h)
16-23 (8 bits)	Device Type: 02h indicating single servo drive
24-31 (8 bits)	Reserved

5.2. Object 1008h: Manufacturer Device Name

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
1008h	00h	Manufacturer Device Name	9GEXX	No	String	RO	No

The Manufacturer Device Name is given by a string value.

5.3. Object 1009h: Manufacturer Hardware Version

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
1009h	00h	Manufacturer Hardware Version	9000-7001-000	No	String	RO	No

The Manufacturer Hardware Version is given by a string value.

5.4. Object 100Ah: Manufacturer Software Version

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
100Ah	00h	Manufacturer Software Version	xxx	No	String	RO	No

The Manufacturer Software Version is given by a string value.

5.5. Object 1010h: Store Parameters

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
1010h	00h	Number of Entries	03h	No	Unsigned 8-bit	RO	No
1010h	01h	Save All Parameters	01h	No	Unsigned 32-bit	RW*	No
1010h	02h	Save Communication Parameters	01h	No	Unsigned 32-bit	RW*	No
1010h	03h	Save Application Parameters	01h	No	Unsigned 32-bit	RW*	No

* This object is only writable when ESM is in PreOP mode.

Current parameters are saved to EEPROM when a specific command is written to the appropriate subindex. The saved parameters are used to initialize the objects during power on the drive. The specific command is the reverse of "save", 65766173h. Other commands are not valid.

If 65766173h is written to subindex 1, all parameters are saved.

If 65766173h is written to subindex 2, the communication parameters are saved.

If 65766173h is written to subindex 3, the application parameters are saved.

After writing the command, '0' is returned during saving. If saving is succeeded, '1' is returned. If an incorrect command is input, SDO abort code (0x08000020) is returned. If saving is failed, SDO abort code (0x06060000) is returned.

5.6. Object 1011h: Restore Default Parameters

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
1011h	00h	Number of Entries	03h	No	Unsigned 8-bit	RO	No
1011h	01h	Restore All Default Parameters	01h	No	Unsigned 32-bit	RW*	No
1011h	02h	Restore Default Communication Parameters	01h	No	Unsigned 32-bit	RW*	No
1011h	03h	Restore Default Application Parameters	01h	No	Unsigned 32-bit	RW*	No

* This object is only writable when ESM is in PreOP mode.

Default parameters are restored and saved to EEPROM when a specific command is written to the appropriate subindex. The specific command is the reverse of “load”, 64616F6Ch. Other commands are not valid.

If 64616F6Ch is written to subindex 1, all default parameters are restored and saved.

If 64616F6Ch is written to subindex 2, default communication parameters are restored and saved.

If 64616F6Ch is written to subindex 3, default application parameters are restored and saved.

After writing the command, ‘0’ is returned during saving. If saving is succeeded, ‘1’ is returned. If an incorrect command is input, SDO abort code (0x08000020) is returned. If saving is failed, SDO abort code (0x06060000) is returned.

5.7. Object 1018h: Identity Object

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
1018h	00h	Number of Entries	04h	No	Unsigned 8-bit	RO	No
1018h	01h	Vendor ID	2C0h	No	Unsigned 32-bit	RO	No
1018h	02h	Product Code	90700100h	No	Unsigned 32-bit	RO	No
1018h	03h	Revision Number	xxxxxxxxh	No	Unsigned 32-bit	RO	No
1018h	04h	Serial Number	xxxxxxxxh	No	Unsigned 32-bit	RO	No

The Identity Object provides general identification information of the EtherCAT device.

Table 5 - 2. Definition of 1018h

Sub-index	Description
00h	Number of Entries: The number of Sub-Indices in the object.
01h	Vendor ID: Glentek Vendor ID 000002C0h
02h	Product Code: 90700100
03h	Revision Number
04h	Serial Number

5.8. Object 10F1h: Error Settings [4]

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
10F1h	00h	Error Settings	02h	No	Unsigned 8-bit	RO	No
10F1h	01h	Local Error Reaction	01h	No	Unsigned 32-bit	RW	No
10F1h	02h	Sync Error Counter Limit	04h	No	Unsigned 16-bit	RW	No

10F1:01h Local Error Reaction:

It describes which error reaction shall be used in case of an error. Set to 1 is disable SyncManager outputs from PDI side and go to ErrSafeOp.

10F1:02h Sync Error Counter Limit:

The SM Event Missed Error Counter increments every time by one when an SM Event was missed. If the internal Sync Error Counter exceeds the Sync Error Counter Limit a multiple error is detected and the slave changes its EtherCAT state to SAFEOP with AL Status Code 0x1A.

5.9. Object 10F8h: Timestamp Object

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
10F8h	00h	Actual Time Stamp	-	Yes	Unsigned 64-bit	RW	No

The Timestamp object holds the current local time.

5.10. Object 1600h: RxPDO0

Object	Sub-Index	Default	Data Range	PDO Map	Data Type	Access	Save to EEPROM
1600h	00h	09h	00h ~ 10h	No	Unsigned 8-bit	RW	Yes
1600h	01h	60400010h	-	No	Unsigned 32-bit	RW	Yes
1600h	02h	607A0020h	-	No	Unsigned 32-bit	RW	Yes
1600h	03h	60FF0020h	-	No	Unsigned 32-bit	RW	Yes
1600h	04h	60710010h	-	No	Unsigned 32-bit	RW	Yes
1600h	05h	60600008h	-	No	Unsigned 32-bit	RW	Yes
1600h	06h	00000008h	-	No	Unsigned 32-bit	RW	Yes
1600h	07h	60720010h	-	No	Unsigned 32-bit	RW	Yes
1600h	08h	60B80010h	-	No	Unsigned 32-bit	RW	Yes
1600h	09h	60FE0120h	-	No	Unsigned 32-bit	RW	Yes
1600h	0Ah	-	-	No	Unsigned 32-bit	RW	Yes
1600h	0Bh	-	-	No	Unsigned 32-bit	RW	Yes
1600h	0Ch	-	-	No	Unsigned 32-bit	RW	Yes
1600h	0Dh	-	-	No	Unsigned 32-bit	RW	Yes
1600h	0Eh	-	-	No	Unsigned 32-bit	RW	Yes
1600h	0Fh	-	-	No	Unsigned 32-bit	RW	Yes
1600h	10h	-	-	No	Unsigned 32-bit	RW	Yes

The default values of RxPDO0 are for the mode that is dynamically switchable among CSP, CSV, and CST mode of operation. This PDO can be dynamically mapped up to 16 objects. If an 8-bit object (such as 60600**08**h) is needed to map, an 8-bit padding (such as 00000**08**h) is needed.

5.11. Object 1601h: RxPDO1

Object	Sub-Index	Default	Data Range	PDO Map	Data Type	Access	Save to EEPROM
1601h	00h	02h	00h ~ 08h	No	Unsigned 8-bit	RW	Yes
1601h	01h	60400010h	-	No	Unsigned 32-bit	RW	Yes
1601h	02h	607A0020h	-	No	Unsigned 32-bit	RW	Yes
1601h	03h	-	-	No	Unsigned 32-bit	RW	Yes
1601h	04h	-	-	No	Unsigned 32-bit	RW	Yes
1601h	05h	-	-	No	Unsigned 32-bit	RW	Yes
1601h	06h	-	-	No	Unsigned 32-bit	RW	Yes
1601h	07h	-	-	No	Unsigned 32-bit	RW	Yes
1601h	08h	-	-	No	Unsigned 32-bit	RW	Yes

The default values of RxPDO1 are for the CSP mode of operation. This PDO can be dynamically mapped up to 8 objects.

5.12. Object 1602h: RxPDO2

Object	Sub-Index	Default	Data Range	PDO Map	Data Type	Access	Save to EEPROM
1602h	00h	02h	00h ~ 08h	No	Unsigned 8-bit	RW	Yes
1602h	01h	60400010h	-	No	Unsigned 32-bit	RW	Yes
1602h	02h	60FF0020h	-	No	Unsigned 32-bit	RW	Yes
1602h	03h	-	-	No	Unsigned 32-bit	RW	Yes
1602h	04h	-	-	No	Unsigned 32-bit	RW	Yes
1602h	05h	-	-	No	Unsigned 32-bit	RW	Yes
1602h	06h	-	-	No	Unsigned 32-bit	RW	Yes
1602h	07h	-	-	No	Unsigned 32-bit	RW	Yes
1602h	08h	-	-	No	Unsigned 32-bit	RW	Yes

The default values of RxPDO2 are for the CSV mode of operation. This PDO can be dynamically mapped up to 8 objects.

5.13. Object 1603h: RxPDO3

Object	Sub-Index	Default	Data Range	PDO Map	Data Type	Access	Save to EEPROM
1603h	00h	02h	00h ~ 08h	No	Unsigned 8-bit	RW	Yes
1603h	01h	60400010h	-	No	Unsigned 32-bit	RW	Yes
1603h	02h	60710010h	-	No	Unsigned 32-bit	RW	Yes
1603h	03h	-	-	No	Unsigned 32-bit	RW	Yes
1603h	04h	-	-	No	Unsigned 32-bit	RW	Yes
1603h	05h	-	-	No	Unsigned 32-bit	RW	Yes
1603h	06h	-	-	No	Unsigned 32-bit	RW	Yes
1603h	07h	-	-	No	Unsigned 32-bit	RW	Yes
1603h	08h	-	-	No	Unsigned 32-bit	RW	Yes

The default values of RxPDO3 are for the CST mode of operation. This PDO can be dynamically mapped up to 8 objects.

5.14. Object 1A00h: TxPDO0

Object	Sub-Index	Default	Data Range	PDO Map	Data Type	Access	Save to EEPROM
1A00h	00h	0Ah	00h ~ 10h	No	Unsigned 8-bit	RW	Yes
1A00h	01h	60410010h	-	No	Unsigned 32-bit	RW	Yes
1A00h	02h	60640020h	-	No	Unsigned 32-bit	RW	Yes
1A00h	03h	606C0020h	-	No	Unsigned 32-bit	RW	Yes
1A00h	04h	60770010h	-	No	Unsigned 32-bit	RW	Yes
1A00h	05h	60610008h	-	No	Unsigned 32-bit	RW	Yes
1A00h	06h	00000008h	-	No	Unsigned 32-bit	RW	Yes
1A00h	07h	60F40020h	-	No	Unsigned 32-bit	RW	Yes
1A00h	08h	60B90010h	-	No	Unsigned 32-bit	RW	Yes
1A00h	09h	60BA0020h	-	No	Unsigned 32-bit	RW	Yes
1A00h	0Ah	60FD0020h	-	No	Unsigned 32-bit	RW	Yes
1A00h	0Bh	-	-	No	Unsigned 32-bit	RW	Yes
1A00h	0Ch	-	-	No	Unsigned 32-bit	RW	Yes
1A00h	0Dh	-	-	No	Unsigned 32-bit	RW	Yes
1A00h	0Eh	-	-	No	Unsigned 32-bit	RW	Yes
1A00h	0Fh	-	-	No	Unsigned 32-bit	RW	Yes
1A00h	10h	-	-	No	Unsigned 32-bit	RW	Yes

The default values of TxPDO0 are for the mode that is dynamically switchable among CSP, CSV, and CST mode of operation. This PDO can be dynamically mapped up to 16 objects.

5.15. Object 1A01h: TxPDO1

Object	Sub-Index	Default	Data Range	PDO Map	Data Type	Access	Save to EEPROM
1A01h	00h	02h	00h ~ 08h	No	Unsigned 8-bit	RW	Yes
1A01h	01h	60410010h	-	No	Unsigned 32-bit	RW	Yes
1A01h	02h	60640020h	-	No	Unsigned 32-bit	RW	Yes
1A01h	03h	-	-	No	Unsigned 32-bit	RW	Yes
1A01h	04h	-	-	No	Unsigned 32-bit	RW	Yes
1A01h	05h	-	-	No	Unsigned 32-bit	RW	Yes
1A01h	06h	-	-	No	Unsigned 32-bit	RW	Yes
1A01h	07h	-	-	No	Unsigned 32-bit	RW	Yes
1A01h	08h	-	-	No	Unsigned 32-bit	RW	Yes

The default values of TxPDO1 are for the CSP mode of operation. This PDO can be dynamically mapped up to 8 objects.

5.16. Object 1A02h: TxPDO2

Object	Sub-Index	Default	Data Range	PDO Map	Data Type	Access	Save to EEPROM
1A02h	00h	02h	00h ~ 08h	No	Unsigned 8-bit	RW	Yes
1A02h	01h	60410010h	-	No	Unsigned 32-bit	RW	Yes
1A02h	02h	60640020h	-	No	Unsigned 32-bit	RW	Yes
1A02h	03h	-	-	No	Unsigned 32-bit	RW	Yes
1A02h	04h	-	-	No	Unsigned 32-bit	RW	Yes
1A02h	05h	-	-	No	Unsigned 32-bit	RW	Yes
1A02h	06h	-	-	No	Unsigned 32-bit	RW	Yes
1A02h	07h	-	-	No	Unsigned 32-bit	RW	Yes
1A02h	08h	-	-	No	Unsigned 32-bit	RW	Yes

The default values of TxPDO2 are for the CSV mode of operation. This PDO can be dynamically mapped up to 8 objects.

5.17. Object 1A03h: TxPDO3

Object	Sub-Index	Default	Data Range	PDO Map	Data Type	Access	Save to EEPROM
1A03h	00h	03h	00h ~ 08h	No	Unsigned 8-bit	RW	Yes
1A03h	01h	60410010h	-	No	Unsigned 32-bit	RW	Yes
1A03h	02h	60640020h	-	No	Unsigned 32-bit	RW	Yes
1A03h	03h	60770010h	-	No	Unsigned 32-bit	RW	Yes
1A03h	04h	-	-	No	Unsigned 32-bit	RW	Yes
1A03h	05h	-	-	No	Unsigned 32-bit	RW	Yes
1A03h	06h	-	-	No	Unsigned 32-bit	RW	Yes
1A03h	07h	-	-	No	Unsigned 32-bit	RW	Yes
1A03h	08h	-	-	No	Unsigned 32-bit	RW	Yes

The default values of TxPDO3 are for the CST mode of operation. This PDO can be dynamically mapped up to 8 objects.

5.18. Object 1C00h: Sync Manager Type

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
1C00h	00h	Number of Entries	04h	No	Unsigned 8-bit	RO	No
1C00h	01h	Sync-Manager Type Channel 1: Mailbox Write	01h	No	Unsigned 8-bit	RO	No
1C00h	02h	Sync-Manager Type Channel 2: Mailbox Read	02h	No	Unsigned 8-bit	RO	No
1C00h	03h	Sync-Manager Type Channel 3: Process Data Write (Outputs)	03h	No	Unsigned 8-bit	RO	No
1C00h	04h	Sync-Manager Type Channel 4: Process Data Read (Inputs)	04h	No	Unsigned 8-bit	RO	No

5.19. Object 1C12h: SM2 Assignment

Object	Sub-Index	Default	Data Range	PDO Map	Data Type	Access	Save to EEPROM
1C12h	00h	01h	00h ~ 01h	No	Unsigned 8-bit	RW	Yes
1C12h	01h	1601h	-	No	Unsigned 32-bit	RW	Yes

The object 1601h (for CSP mode) is assigned as default.

5.20. Object 1C13h: SM3 Assignment

Object	Sub-Index	Default	Data Range	PDO Map	Data Type	Access	Save to EEPROM
1C13h	00h	01h	00h ~ 01h	No	Unsigned 8-bit	RW	Yes
1C13h	01h	1A01h	-	No	Unsigned 32-bit	RW	Yes

The object 1A01h (for CSP mode) is assigned as default.

5.21. Object 1C32h: SM2 Output Parameters

Non-DC Mode (Freerun):

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
1C32h	00h	Number of Entries	20h	No	Unsigned 8-bit	RO	No
1C32h	01h	Synchronization Type	0h	No	Unsigned 16-bit	RW	No
1C32h	02h	Cycle Time	Local Cycle Time	No	Unsigned 32-bit	RO	No
1C32h	03h	Shift Time	0h	No	Unsigned 32-bit	RW	No
1C32h	04h	Synchronization Types Supported	Bit 0: Free Run Supported	No	Unsigned 16-bit	RO	No
1C32h	05h	Minimum Cycle Time	000249F0h	No	Unsigned 32-bit	RO	No
1C32h	06h	Calc and Copy Time	0h	No	Unsigned 32-bit	RO	No
1C32h	08h	Get Cycle Time	0h	No	Unsigned 16-bit	RW	No
1C32h	09h	Delay Time	0h	No	Unsigned 32-bit	RO	No
1C32h	0Ah	Sync0 Cycle Time	Same as 1C32:02h	No	Unsigned 32-bit	RW	No
1C32h	0Bh	SM Event Missed Counter	0h	No	Unsigned 16-bit	RO	No
1C32h	0Ch	Cycle Exceeded Counter	0h	No	Unsigned 16-bit	RO	No
1C32h	20h	Sync Error	FALSE	No	BOOL	RO	No

DC Mode (Sync0 Synchronization):

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
1C32h	00h	Number of Entries	20h	No	Unsigned 8-bit	RO	No
1C32h	01h	Synchronization Type	0002h	No	Unsigned 16-bit	RW	No
1C32h	02h	Cycle Time	Same value as Sync0 Cycle Time (register 0x09A3:0x09A0)	No	Unsigned 32-bit	RO	No
1C32h	03h	Shift Time	0h	No	Unsigned 32-bit	RW	No
1C32h	04h	Synchronization Types Supported	Bit 4:2: DC Type supported, 001b = DC Sync0	No	Unsigned 16-bit	RO	No
1C32h	05h	Minimum Cycle Time	000249F0h	No	Unsigned 32-bit	RO	No
1C32h	06h	Calc and Copy Time	00007530h	No	Unsigned 32-bit	RO	No
1C32h	08h	Get Cycle Time	0h	No	Unsigned 16-bit	RW	No
1C32h	09h	Delay Time	0h	No	Unsigned 32-bit	RO	No
1C32h	0Ah	Sync0 Cycle Time	Same as 1C32:02h	No	Unsigned 32-bit	RW	No
1C32h	0Bh	SM Event Missed Counter	0h	No	Unsigned 16-bit	RO	No
1C32h	0Ch	Cycle Exceeded Counter	0h	No	Unsigned 16-bit	RO	No
1C32h	20h	Sync Error	FALSE	No	BOOL	RO	No

5.22. Object 1C33h: SM3 Input Parameters

Non-DC Mode (Freerun):

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
1C33h	00h	Number of Entries	20h	No	Unsigned 8-bit	RO	No
1C33h	01h	Synchronization Type	0h	No	Unsigned 16-bit	RW	No
1C33h	02h	Cycle Time	Same as 1C32:02h	No	Unsigned 32-bit	RO	No
1C33h	03h	Shift Time	0h	No	Unsigned 32-bit	RW	Yes
1C33h	04h	Synchronization Types Supported	Same as 1C32:04h	No	Unsigned 16-bit	RO	No
1C33h	05h	Minimum Cycle Time	Same as 1C32:05h	No	Unsigned 32-bit	RO	No
1C33h	06h	Calc and Copy Time	0h	No	Unsigned 32-bit	RO	No
1C33h	08h	Get Cycle Time	0h	No	Unsigned 16-bit	RW	No
1C33h	09h	Delay Time	0h	No	Unsigned 32-bit	RO	No
1C33h	0Ah	Sync0 Cycle Time	Same as 1C33:02h	No	Unsigned 32-bit	RW	No
1C33h	0Bh	SM Event Missed Counter	0h	No	Unsigned 16-bit	RO	No
1C33h	0Ch	Cycle Exceeded Counter	0h	No	Unsigned 16-bit	RO	No
1C33h	20h	Sync Error	FALSE	No	BOOL	RO	No

DC Mode (Sync0 Synchronization):

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
1C33h	00h	Number of Entries	20h	No	Unsigned 8-bit	RO	No
1C33h	01h	Synchronization Type	0h	No	Unsigned 16-bit	RW	No
1C33h	02h	Cycle Time	Same as 1C32:02h	No	Unsigned 32-bit	RO	No
1C33h	03h	Shift Time	1D4C0h	No	Unsigned 32-bit	RW	Yes
1C33h	04h	Synchronization Types Supported	Same as 1C32:04h	No	Unsigned 16-bit	RO	No
1C33h	05h	Minimum Cycle Time	Same as 1C32:05h	No	Unsigned 32-bit	RO	No
1C33h	06h	Calc and Copy Time	00007530h	No	Unsigned 32-bit	RO	No
1C33h	08h	Get Cycle Time	0h	No	Unsigned 16-bit	RW	No
1C33h	09h	Delay Time	0h	No	Unsigned 32-bit	RO	No
1C33h	0Ah	Sync0 Cycle Time	Same as 1C33:02h	No	Unsigned 32-bit	RW	No
1C33h	0Bh	SM Event Missed Counter	0h	No	Unsigned 16-bit	RO	No
1C33h	0Ch	Cycle Exceeded Counter	0h	No	Unsigned 16-bit	RO	No
1C33h	20h	Sync Error	FALSE	No	BOOL	RO	No

6. CiA402 Drive Control and Objects

6.1. General

The drive is operated by local signals and by the Controlword sent by the control device via the network. The state of the drive is reported by the Statusword produced by the drive device. The drive State Machine is also controlled by error detection signals.

The Drive State Machine defines the status and the possible control sequence of the drive. A single state represents a special internal or external behavior. The state of the drive determines which commands are accepted. For example, it is only possible to start a point-to-point move when the drive is in the operation enabled state.

6.2. Drive State Machine

Figure 6 - 1 shows the Drive State Machine with respect to control of the power electronics as a result of user commands and internal drive faults.

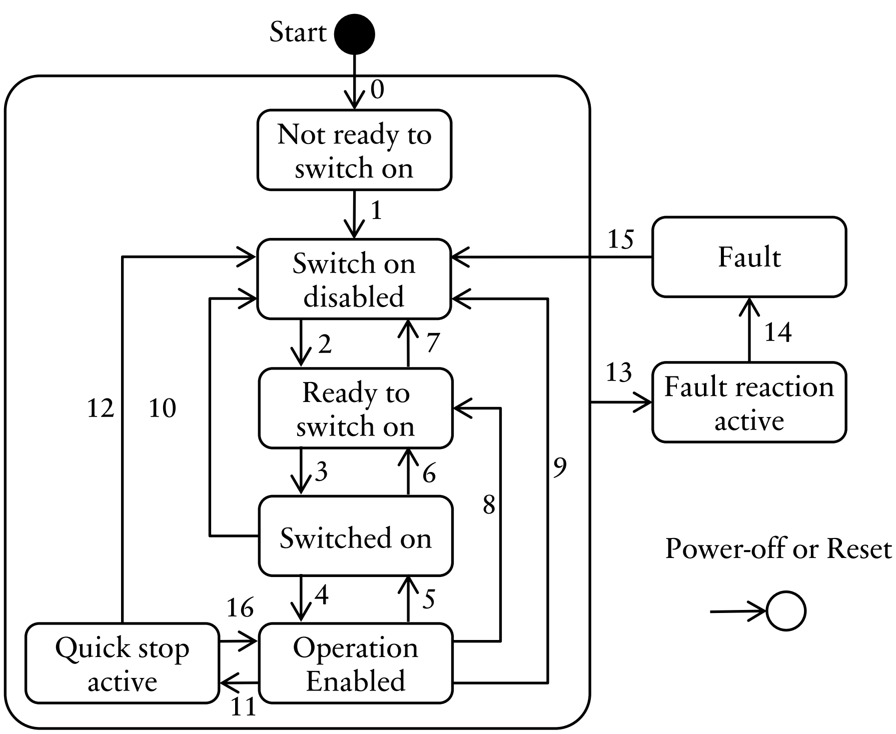


Figure 6 - 1. Drive State Machine

Table 6 - 1 describes the drive functions in each state. If in the *quick stop active* state, the quick stop option code is set to 5, 6, 7 or 8, the drive device will not leave the state but will transit to the *operation enabled* state with the *enabled operation* command. Object 6040h: Controlword controls State Machine state.

Table 6 - 1. State Machine State Functions

Drive State	Function Description
Not Ready to Switch on	Logic Power is applied to the Drive. Drive is initialized. Brake should be active if present. Drive is disabled. High Level Power may be applied.
Switch On Disabled	Drive Default parameters are loaded. Drive parameters may be changed. Drive is disabled. High Level Power may be applied. Drive has no fault. Brake should be active if present.
Ready to Switch On	Drive parameters may be changed. Drive is disabled. High Level Power may be applied. Drive has no fault. Brake should be active if present.
Switched On	Drive parameters may be changed. Drive is disabled. High Level Power is applied. Drive has no fault. Brake should be active if present. Drive is Ready for operation. No hardware Inhibit is active.
Operation Enabled	Limited Drive parameters may be changed depends on the operation mode. High Level Power is applied. Drive is enabled, and outputs power to the motor. Drive has no fault. Brake is released.
Quick Stop Active	Limited Drive parameters may be changed. High Level Power is applied. Drive is enabled, and outputs power to the motor. Drive has no fault. Brake is released. The transition State after Quick Stop completed depends on the 'Halt Option Code' 605Dh.
Fault Reaction Active	Faults have been detected. Drive reacts to the faults based on 'Fault reaction option code' 605Eh and fault type. Drive is disabled immediately when a Fatal fault is detected, and the brake should be active if present.
Fault	Drive parameters may be changed. Drive is disabled. High Level Power may be applied depends on the application. Drive has faults. Brake should be active if present.

If a state transition is requested, the related actions is processed completely before transitioning to a new state. The drive provides option to control the contractor for the mains may switch the high-level power. If the high-level power is switched-off, the motor is free to rotate if not braked. If a brake is present, the high-level power is switched off after a delay time in order to allow the brake to engage. No energy is being supply to the motor when the drive is disabled. Drive function enabled implies that energy is being supplied to the motor. Target or set-point values, such as torque, velocity, position, are only processed while the drive function is enabled.

If a fault is detected in the drive device, there is a transition to the *fault reaction active* state. Drive is automatically disabled in Fault State. The *fault* state is only left by using a fault reset command, after the fault is removed. In case of a fatal error, the drive device is no longer able to control the motor, remove the high power immediately.

6.3. Control and Status Objects

6.3.1. Object 6040h: Controlword

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
6040h	00h	Controlword	0h	Yes	Unsigned 16-bit	RW	No

This object indicates the received command controlling the Drive State Machine. It is structed as defined in Figure 6 - 2. The commands are coded as given in Table 6 - 2.

15 11	10	9	8	7	6 4	3	2	1	0
Reserved	*r*	*oms*	*h*	*fr*	*oms*	*eo*	*qs*	*ev*	*so*

MSB LSB

Reserved; **r** = reserved; **oms** = operation mode specific; **h** = halt; **fr** = fault reset; **eo** = enabled operation; **qs** = quick stop; **ev** = enable voltage; **so** = switch on

Figure 6 - 2. Controlword Value Definition

Table 6 - 2. Controlword Command Coding

Command	Bits of the Controlword					Transitions
	Bit 7	Bit 3	Bit 2	Bit 1	Bit 0	
Shutdown	0	X	1	1	0	2, 6, 8
Switch on	0	0	1	1	1	3
Switch on + Enable operation	0	1	1	1	1	3 + 4*
Disable voltage	0	X	X	0	X	7, 9, 10, 12
Quick stop	0	X	0	1	X	7, 10, 11
Disable operation	0	0	1	1	1	5
Enable operation	0	1	1	1	1	4, 16
Fault reset	↑	X	X	X	X	15
*Automatic transition to Enable operation state after executing Switch on state functionality						

Bits 9, 6, 5, and 4 of the Controlword are operation mode specific. The halt function (bit 8) behavior is also operation mode specific. If bit 8 is set to 1, the commanded motion is interrupted, the drive behaves as defined in the halt option code. After releasing the halt function, the commanded motion will resume if possible. Bit 10 is reserved for further use and is set to 0. Bits 11-15 are reserved.

6.3.2. Object 6041h: Statusword

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
6041h	00h	Statusword	0h	Yes	Unsigned 16-bit	RO	No

This object provides the status of the Drive. It is structed as defined in Figure 6 - 3. The commands are coded as given in Table 6 - 3.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		*oms*		*ila*	*tr*	*rm*	*ms*	*w*	*sod*	*qs*	*ve*	*f*	*oe*	*so*	*rtso*
MSB															LSB

Reserved; **oms** = operation mode specific; **ila** = Internal limit active; **tr** = target reached; **rm** = remote; **w** = warning; **sod** = switch on disable; **qs** = quick stop; **ve** = voltage enabled; **f** = fault; **oe** = operation enabled; **so** = switch on; **rtso** = ready to switch on

Figure 6 - 3. Statusword Value Definition

Table 6 - 3. Statusword Command Coding

Statusword	Drive State Machine state
xxxx xxxx x0xx 0000b	Not ready to switch on
xxxx xxxx x1xx 0000b	Switch on disabled
xxxx xxxx x01x 0001b	Ready to switch on
xxxx xxxx x01x 0011b	Switched on
xxxx xxxx x01x 0111b	Operation enabled
xxxx xxxx x00x 0111b	Quick stop active
xxxx xxxx x0xx 1111b	Fault reaction active
xxxx xxxx x0xx 1000b	Fault

If Bit 4 (voltage enabled) of the Statusword is 1 then high voltage is being applied to the drive. If Bit 5 (quick stop) of the Statusword is 0 then the drive is reacting on a quick stop request. If Bit 7 (warning) is 1 then there is a warning condition present. A warning is not an error nor a fault. An example of a warning can be a temperature limit exceeded or a job being refused. The status of the Drive State Machine state doesn't change. The cause of the warning is given in Object 603Fh: Error Code.

If Bit 9 (remote) is 1 then the Controlword is processed. If it is 0 (local) then the Controlword is not processed. The drive provides actual values, and it may accept communication objects for configuration data transmission for other parameter objects. If Bit 10 (target reached) is 1 then the drive has reached, it's set point. The set point is operation mode specific. Bit 10 is always set to 1 when the operation mode is changed. The change of a target value by software alters this bit. If quick stop option code is 5, 6, 7, or 8, Bit 10 is set to 1 when the quick stop operation is finished, and the drive is halted.

If Bit 11 (internal limit active) of the Statusword is 1, then an internal limit is active. An example of an internal limit is the position range limit. Bit 13 and Bit 12 of the Statusword are operation mode specific. Bit 14 and Bit 15 are reserved.

6.3.3. Object 605Bh: Shutdown Option Code

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
605Bh	00h	Shutdown Option Code	0h	No	Integer 16-bit	RW	No

This object defines what action is performed if there is a transition from Operation Enabled state to Ready to Ready to Switch On state.

Table 6 - 4. 605Bh Value Range Definition

Value	Definition
-32768 to -1	Reserved
0	Disable Drive function (switch-off drive power)
1 to 32768	Reserved

6.3.4. Object 605Ch: Disable Operation Option Code

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
605Ch	00h	Disable Operation Option Code	0h	No	Integer 16-bit	RW	No

This object defines what action is performed if there is a transition from Operation Enabled state to Switched on state.

Table 6 - 5. 605Ch Value Range Definition

Value	Definition
-32768 to -1	Reserved
0	Disable Drive function (switch-off drive power)
1 to 32768	Reserved

6.4. Operation Objects

The drive behavior depends on the active mode of operation. The control device writes to Object 6060h: Modes of Operation to indicate the actual activated operation mode. Controlword, Statusword and set-point objects are used mode-specific to avoid inconsistencies and erroneous behavior.

The following modes of operation are described in this manual:

- Cyclic sync position mode
- Cyclic sync velocity mode
- Cyclic sync torque mode

With the exception of 'Homing mode', the listed modes of operation deal with set-points.

6.4.1. Object 6060h: Modes of Operation

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
6060h	00h	Modes of Operation	08h	Yes	Integer 8-bit	RW	Yes

This object sets the operation mode. The object shows only the value of the requested operation mode. The actual operation mode of the drive is reflected in Object 6061h: Modes of Operation Display. Table 6 - 6 specifies the value definition.

Table 6 - 6. 6060h Value Range Definition

Value	Definition
-128to -1	Reserved
0	No mode change assigned
+1	Profile position mode (pp) – not supported
+2	Velocity mode (vl) – not supported
+3	Profile velocity mode (pv) – not supported
+4	Torque profile mode (tq) – not supported
+5	Reserved (r)
+6	Homing mode (hm)
+7	Interpolated position mode (ip) – not supported
+8	Cyclic sync position mode (csp)
+9	Cyclic sync velocity mode (csv)
+10	Cyclic sync torque mode (cst)
+11 to +127	Reserved

6.4.2. Object 6061h: Modes of Operation Display

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
6061h	00h	Modes of Operation Display	08h	Yes	Integer 8-bit	RO	No

This object provides the actual operation mode. Table 6 - 6 specifies the value definition.

6.4.3. Object 6502h: Supported Drive Modes

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
6502h	00h	Supported Drive Modes	03A0h	No	Unsigned 32-bit	RO	No

This object provides the information on the supported drive modes. Figure 6 - 4 specifies the value definition. The drive currently supports csp, csv, cst, and hm modes.

31 … 10	9	8	7	6	5	4	3	2	1	0
Reserved	*cst*	*csv*	*csp*	*ip*	*hm*	*r*	*tq*	*pv*	*vl*	*pp*
MSB										LSB

Legend: See Table 6 - 6

Figure 6 - 4. Object 6502h Value Definition

For bit 9 (cst), bit 8 (csv), bit 7 (csp), bit 6 (ip), bit 5 (hm), bit 3 (tq), bit 2 (pv), bit 1 (vl), and bit 0 (pp) a 1 means that the mode is supported and a 0 means that the mode is not supported. Bit 10 through bit 31 are set to 0, reserved.

6.4.4. Object 605Ah: Quick Stop Option Code

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
605Ah	00h	Quick Stop Option Code	02h	No	Integer 16-bit	RW	Yes

This object indicates what action is performed when the quick stop function is executed. The slow down ramp is the deceleration value of the used mode operations. Table 6 - 7 specifies the value definition.

Table 6 - 7. 605Ah Value Range Definition

Value	Definition
-32,786 to -1	Reserved
0	Disabled drive function
+1	Slow down on slow down ramp and transit into Switch on disabled
+2	Slow down on quick stop ramp and transit into Switch on disabled
+3	Slow down on current limit and transit into Switch on disabled
+4	Reserved
+5	Slow down on slow down ramp and stay in Quick stop active
+6	Slow down on quick stop ramp and stay in Quick stop active
+7	Slow down on current limit and stay in Quick stop active
+8	Reserved
+9 to +32,767	Reserved

6.4.5. Object 605Dh: Halt Option Code

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
605Dh	00h	Halt Option Code	01h	No	Integer 16-bit	RW	Yes

This object indicates what action is performed when the halt function is executed. The slow down ramp is the deceleration value of the used mode of operations Table 6 - 8 specifies the value definition.

Table 6 - 8. 605Dh Value Range Definition

Value	Definition
-32,768 to -1	Reserved
0	Reserved
+1	Slow down on slow down ramp and stay in Operation enabled
+2	Slow down on quick stop ramp and stay in Operation enabled
+3	Slow down on current limit and stay in Operation enabled
+4 to +32,767	Reserved

6.4.6. Object 605Eh: Fault Reaction Option Code

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
605Eh	00h	Fault Reaction Option Code	0h	No	Integer 16-bit	RW	No

This object defines the action performed when a fault is detected in the drive.

Table 6 - 9. 605Eh Value Range Definition

Value	Definition
-32768 to -1	Reserved
0	Disable Drive function (switch-off drive power)
1 to 32768	Reserved

6.4.7. Object 603Fh: Error Code

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
603Fh	00h	Error Code	-	No	Unsigned 16-bit	RO	No

This object provides the last error code occurred in the drive.

Table 6 - 10. 603Fh Error Code Description

Error Code	Error	Meaning
2310h	LSECB (Low Speed Electronic Circuit Breaker)	The drive output current has exceeded the time + current constraints programmed into the drive
2320h	HSECB (High Speed Electronic Circuit Breaker)	The drive has detected a short circuit on one or more of the output terminals
3210h	Bus Over Volt	The Input Bus Voltage to drive has exceeded the allowed max voltage
3220h	Bus Under volt	The Input Bus Voltage is below the minimum required
4310h	Drive Over Temp	The drive has exceeded the drive maximum allowed temperature
5530h	EEPROM	EEPROM checksum fault
7122h	Commutation Init	Hall angles do not match encoder counter angle. No Halls: Phase finding routine failed
7305h	Encoder	The drive has detected a fault on one or more of the encoder inputs
8611h	Position Following Error	Position Mode Only – Difference between commanded position and actual position is excessive
FF05h	Hall	The drive has detected an invalid Hall state on the Hall inputs from the motor
FF06h	Motor Over Temp	The drive has detected a motor over temp signal on the *Motor Overtemp* Input or on the I/O Connector
FF0Ah	Over Speed	The Motor speed has exceeded the specified threshold

7. Homing Mode

7.1. Overview

For the operation of positioning drives, an exact knowledge of the absolute position is normally required. In homing mode, the drive seeks the home position to gain the exact knowledge of the absolute position. The different homing methods will be discussed in this chapter. Positive and negative limit switches at the end of travel, a home switch in mid-travel, and the index pulses from an incremental encoder are common feedbacks detected and utilized during homing.

7.2. Functional Description

Figure 7 - 1: Homing Mode Function shows the define input and output objects. The user may specify the speeds, acceleration, and the method of homing. The object home offset allows the user to set the zero position away from the home position; see the definition of Object 607Ch Home offset. There is no output data except for those bits in the Statusword. The bits return the status, the result of the homing process, and the demand to the position control loops.

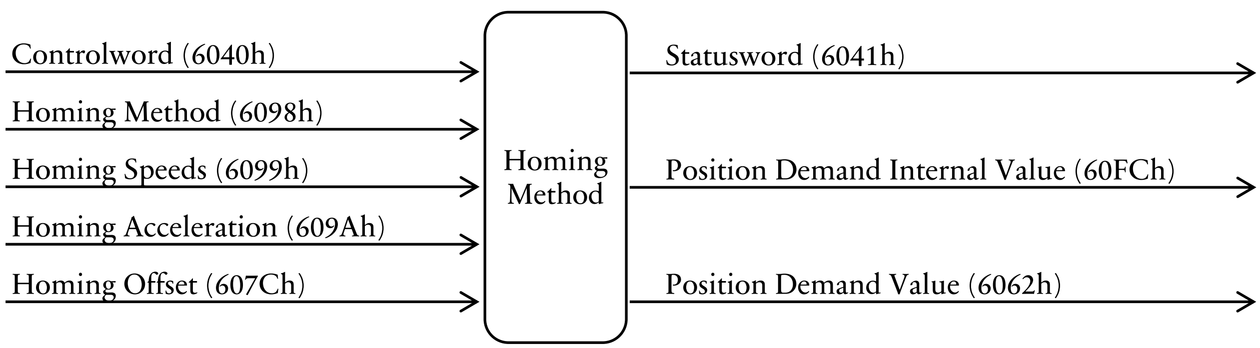


Figure 7 - 1. Homing Mode Function

There are four sources of homing signals available; the negative and positive limit switches, the home switch, and the index pulse from an incremental encoder. Each method described in this section utilizes a unique combination of the homing signals to achieve the home attain status.

In the diagrams of the homing methods shown below, the encoder count increases as the axis' position moves to the right (positive direction) and the encoder count decreases as the axis’ position moves to the left (negative direction). In other words, the left is the minimum position, and the right is the maximum position. In case that a limit switch, at the end of travel, is activated by the axis, the axis will move in the opposite direction to leave the position.

There are two homing speeds in a typical homing cycle. The faster speed is used to find the home switch, which is denoted by thick black line in the diagrams, and the slower speed is used to find the index pulse, which is denoted by thin black line in the diagrams. The direction of movement is also indicated by an arrowhead. The red dot with the number indicates the homing attained position. Figure 7 - 2 highlights the key components utilized in the homing diagrams Figure 7 - 3 through Figure 7 - 32.

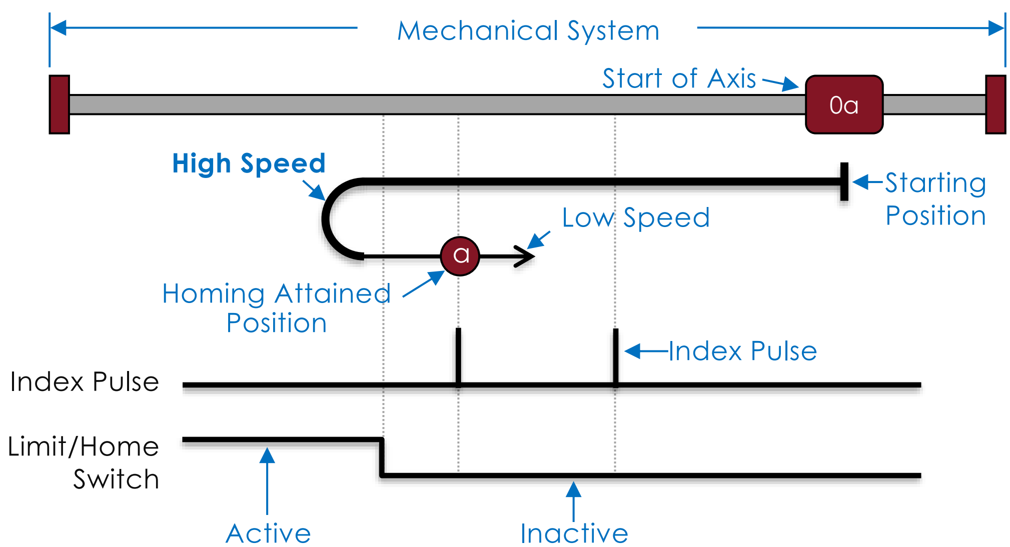


Figure 7 - 2. Key Components for the Homing Diagrams

7.3. General Definitions

This section describes the different homing methods.

- **Homing methods 1-14** utilize home and limit switches in combination with the index pulse.
- **Homing methods 17-30** have identical routines as those to methods 1-14 except that they do not utilize the index pulse and instead are dependent on the edge of the home or limit switch.
- **Homing methods 33 and 34** home to the nearest index pulse.
- **Homing method 35** homes to the current position

7.3.1. Method 1: Homing on Negative Limit Switch and Index Pulse

Axis 1a moves in the negative direction until the **negative limit switch** (NLS) is active. The axis will then move in the positive direction until the first index pulse is detected after the NLS goes inactive. After this event, the drive sets *home attained*.

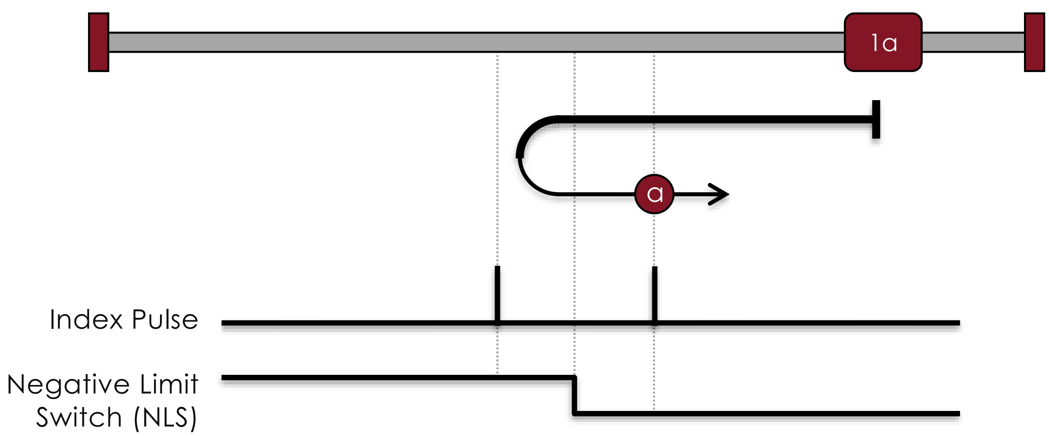


Figure 7 - 3. Homing Method 1

7.3.2. Method 2: Homing on Positive Limit Switch and Index Pulse

Axis 2a moves in the positive direction until the **positive limit switch** (PLS) is active. The axis will then move in the negative direction until the first index pulse is detected after the PLS goes inactive. After this event, the drive sets *home attained*.

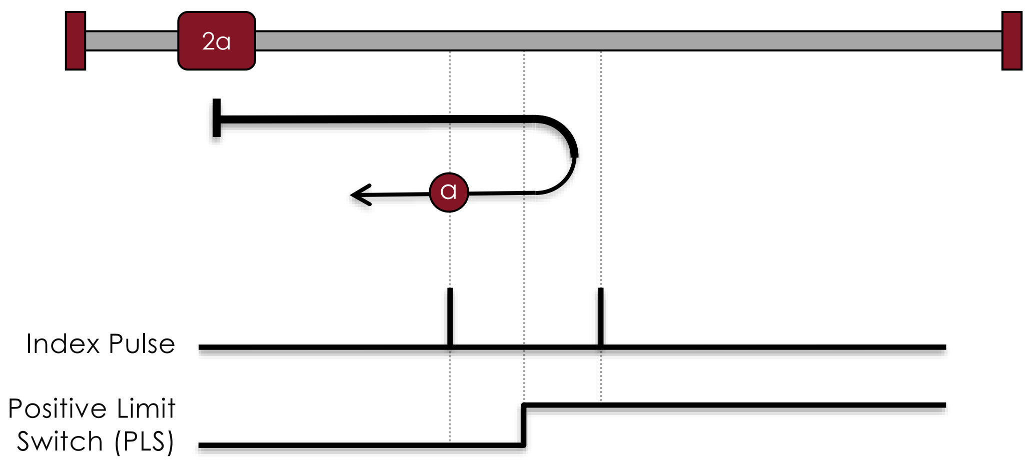


Figure 7 - 4. Homing Method 2

7.3.3. Method 3 and 4: Homing on Positive Home Switch and Index Pulse

Using these methods as shown in Figure 7 - 5 and Figure 7 - 6 the initial direction of movement is dependent on the state of the home switch. The home attain position should be at the index pulse to the left or right of the point where the home switch changes states. If the initial position is situated so that the direction of movement needs to be reversed during homing, the point at which it reverses should be after the home switch changes states.

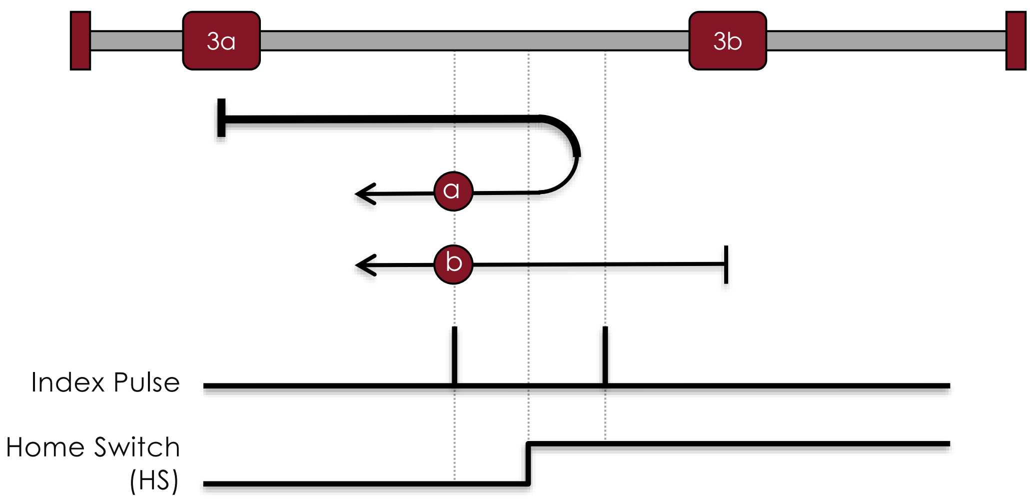


Figure 7 - 5. Homing Method 3

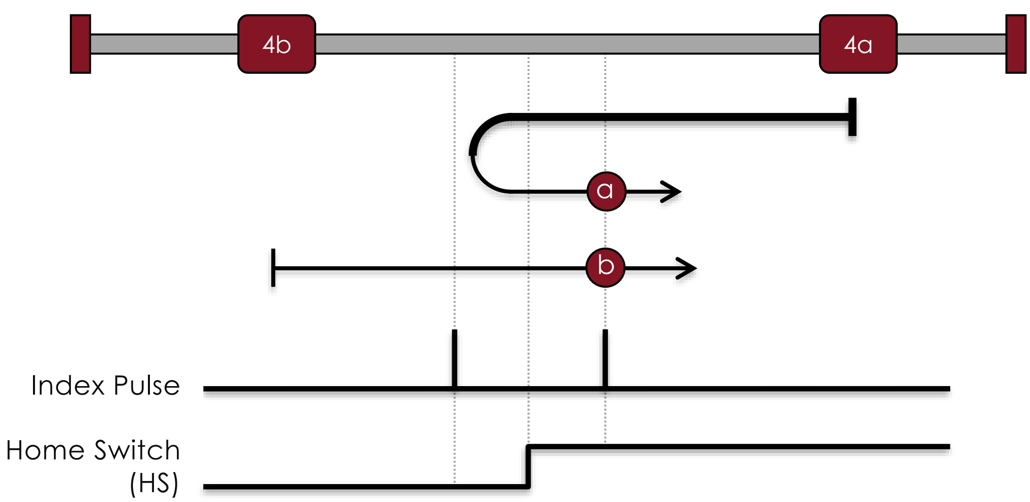


Figure 7 - 6. Homing Method 4

7.3.4. Method 5 and 6: Homing on Negative Home Switch and Index Pulse

Using these methods as shown in Figure 7 - 7 and Figure 7 - 8 the initial direction of movement is dependent on the state of the home switch. The home attain position should be at the index pulse to the left or right of the point where the home switch changes states. If the initial position is situated so that the direction of movement needs to be reversed during homing, the point at which it reverses should be after the home switch changes states.

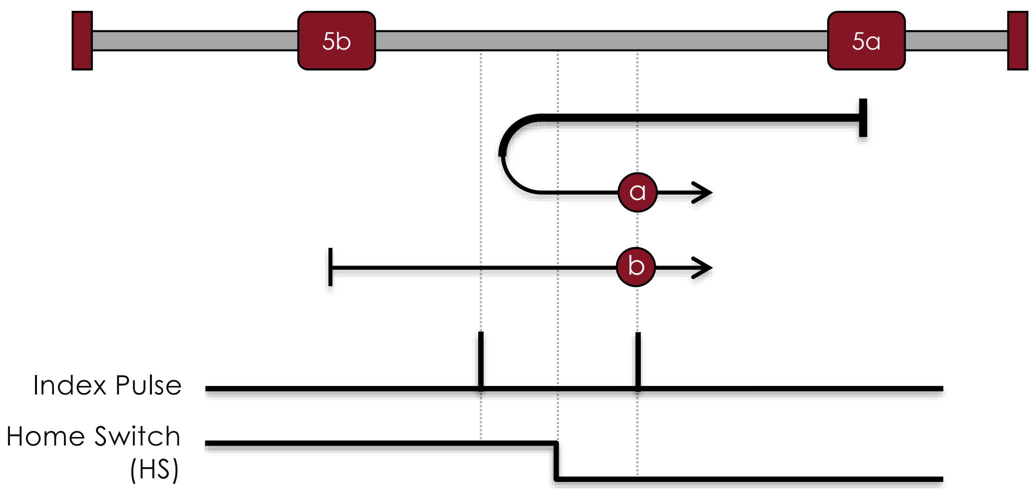


Figure 7 - 7. Homing Method 5

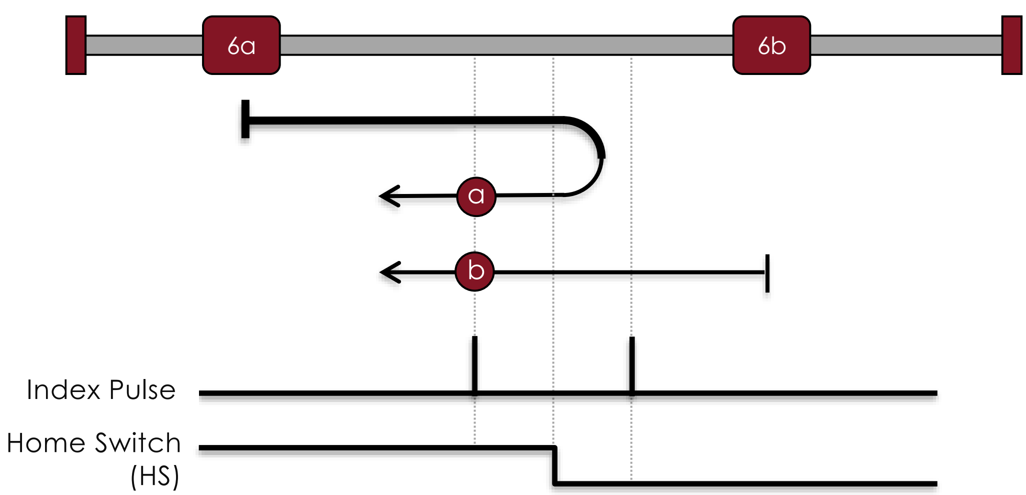


Figure 7 - 8. Homing Method 6

7.3.5. Method 7 to 14: Homing on Home Switch and Index Pulse

These methods use a home switch, which is active only a portion of the travel. The home switch has a 'momentary' action as the axis position goes pass it. Using methods 7 to 10, Figure 7 - 9 through Figure 7 - 12 respectively, the initial direction of movement is to the right. Using methods 11 to 14, Figure 7 - 13 through Figure 7 - 16 respectively, the initial direction of movement is to the left except if the home switch is active at the start of the motion. In such case, the initial direction of motion depends on the edge of interest. The home position is at the index pulse on the rising or falling edges of the home switch. If the initial direction of movement leads away from the home switch, the drive should reverse on encountering the relevant limit switch.

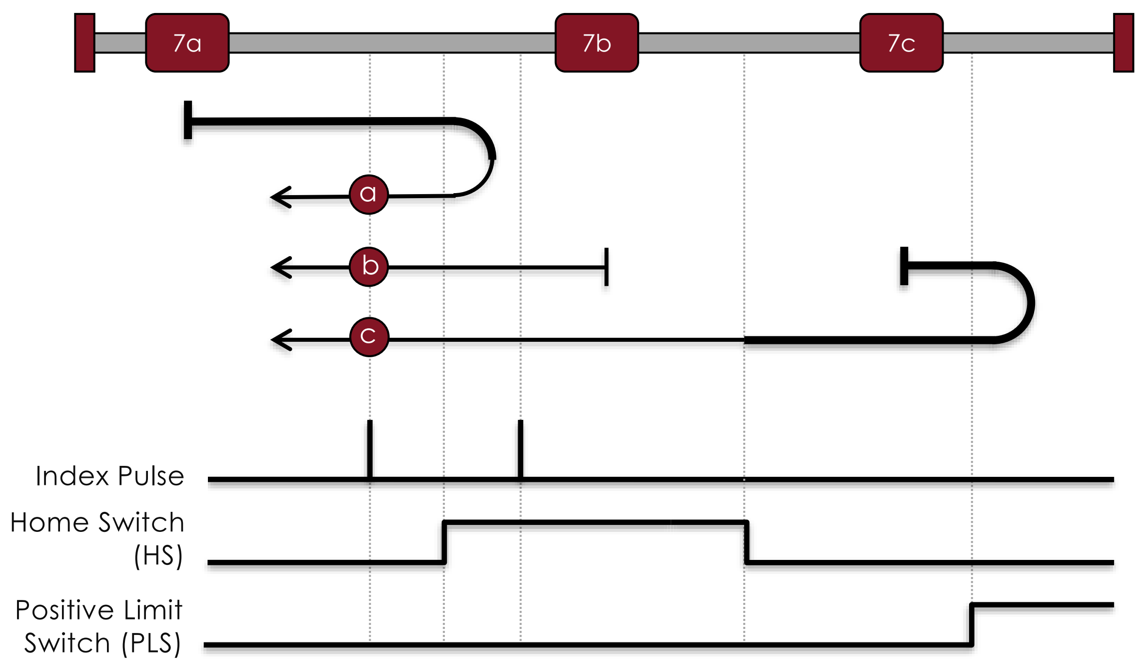


Figure 7 - 9. Homing Method 7

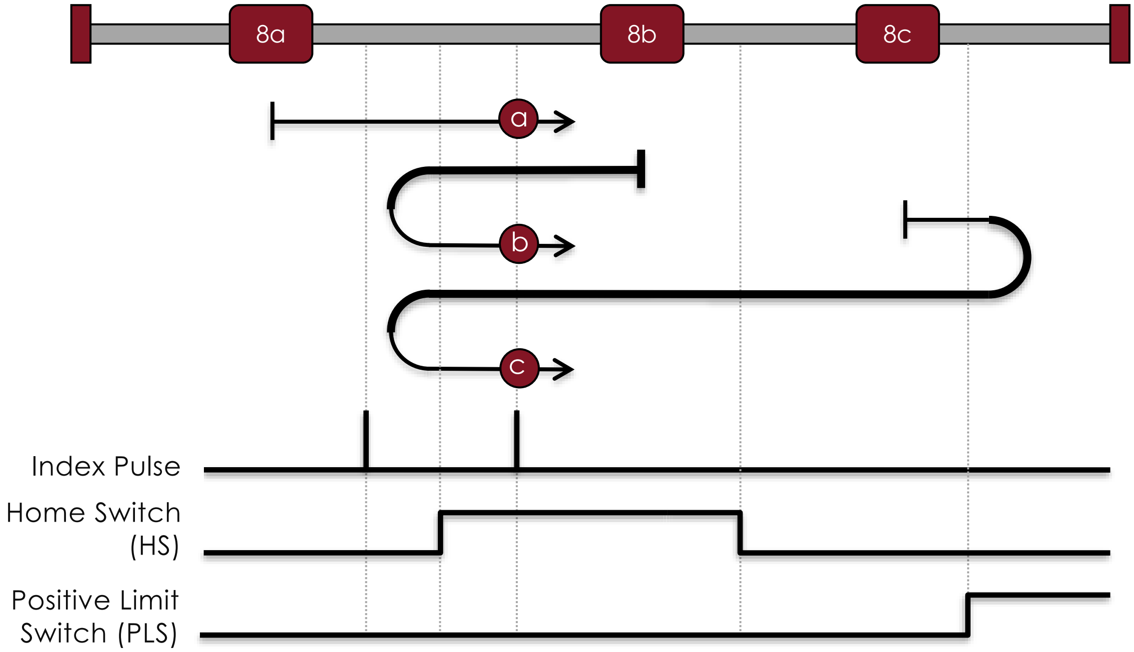


Figure 7 - 10. Homing Method 8

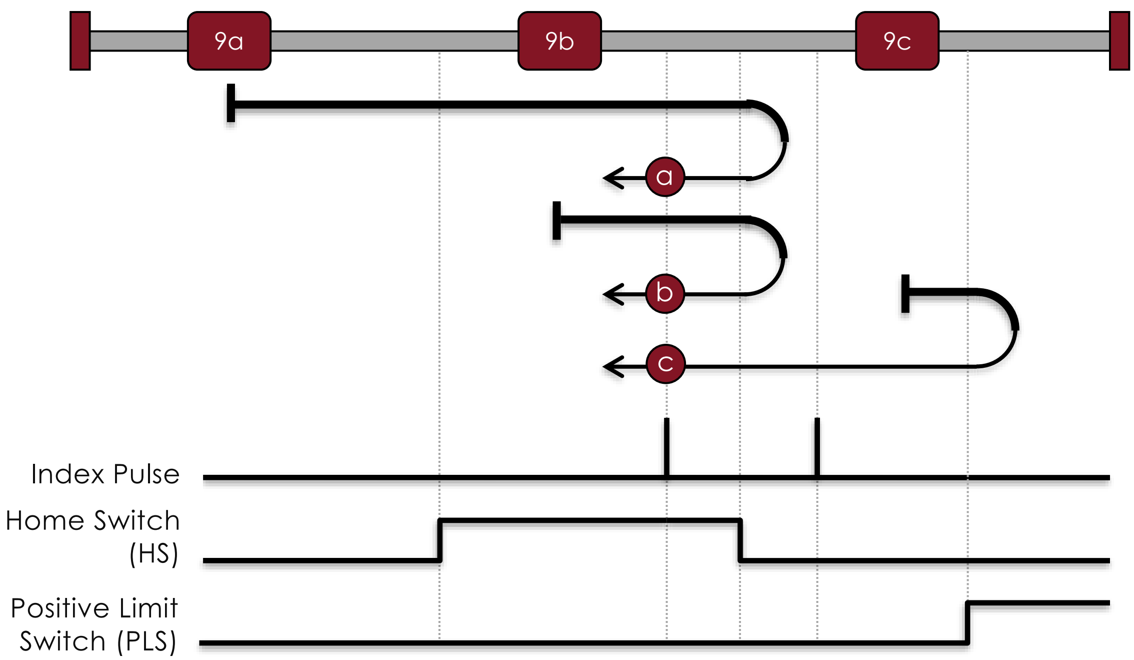


Figure 7 - 11. Homing Method 9

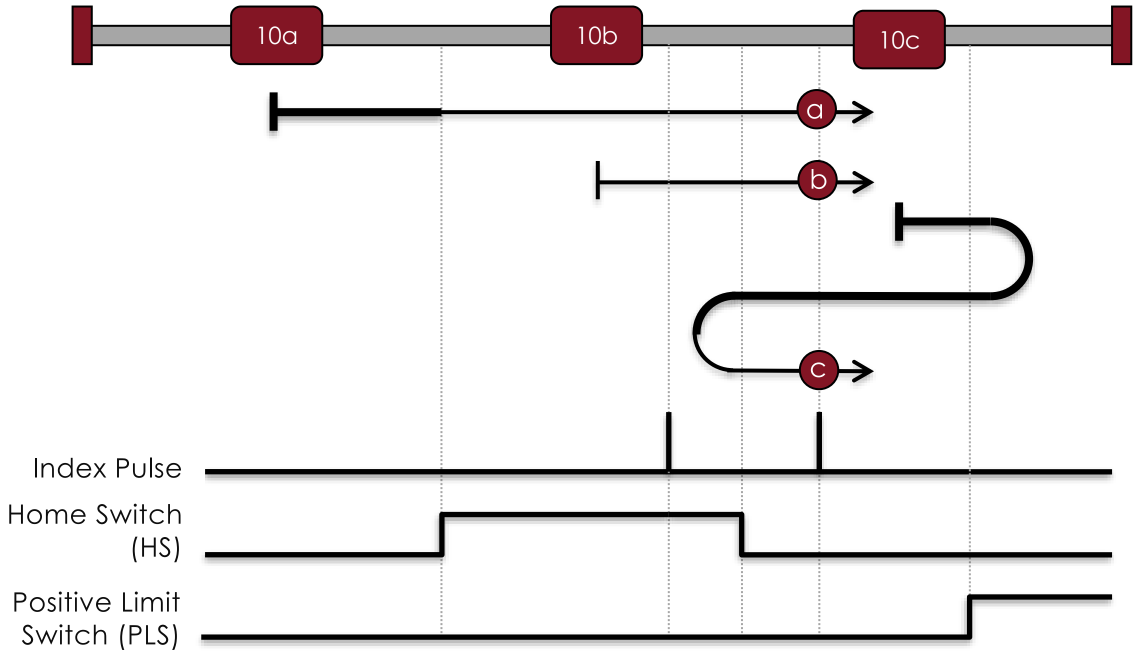


Figure 7 - 12. Homing Method 10

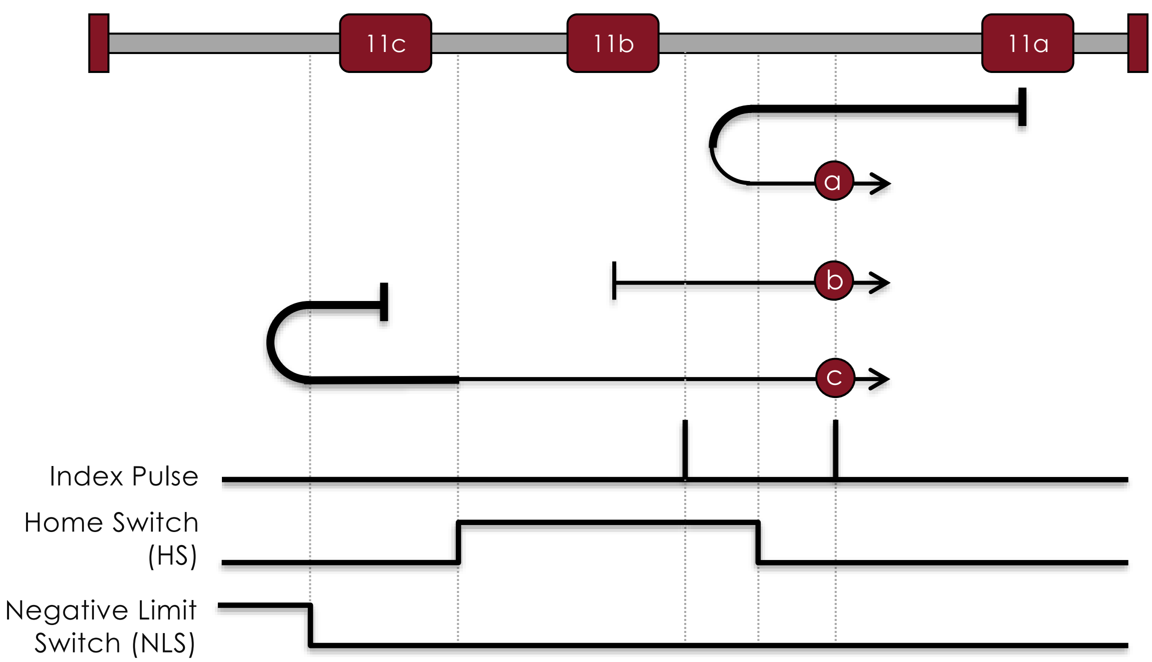


Figure 7 - 13. Homing Method 11

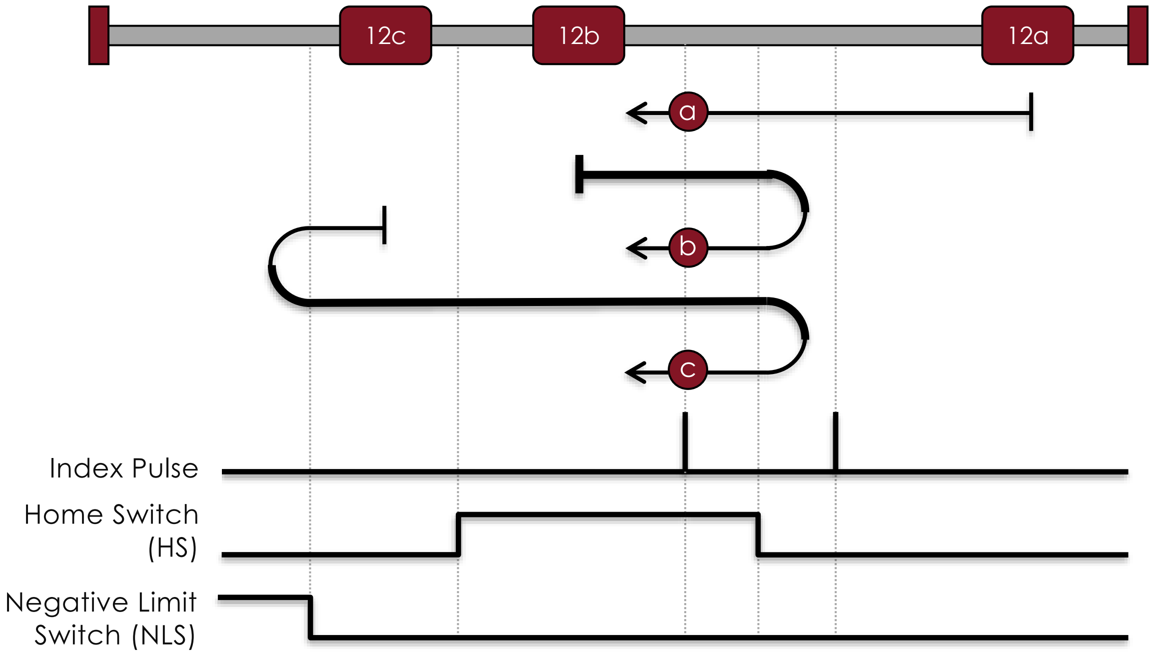


Figure 7 - 14. Homing Method 12

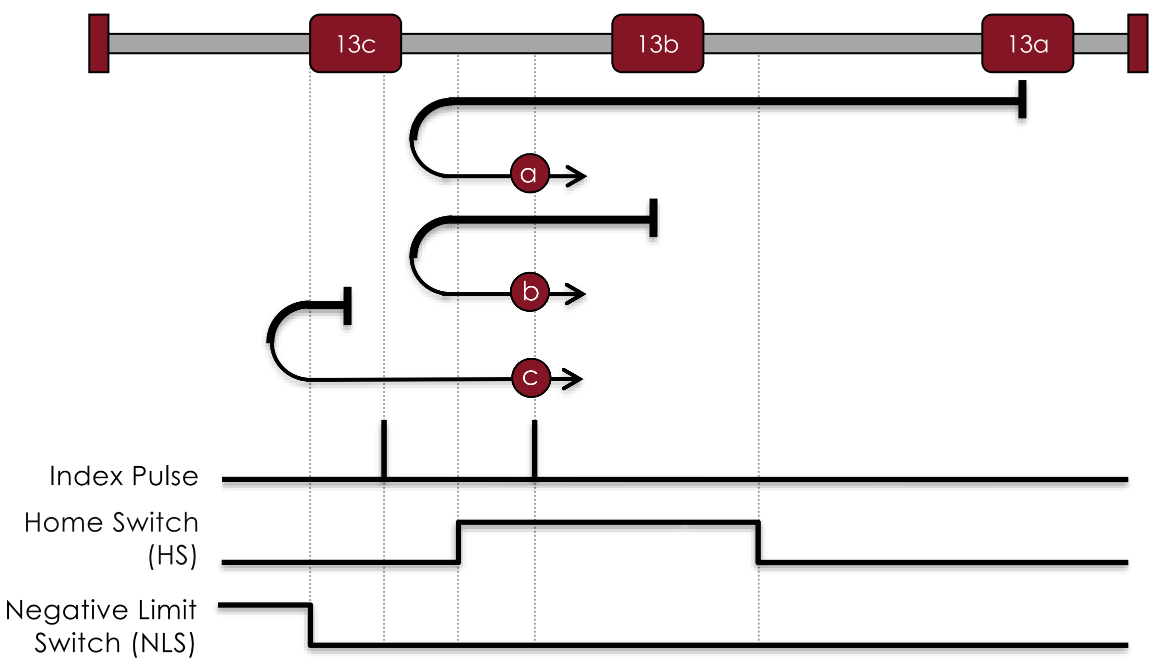


Figure 7 - 15. Homing Method 13

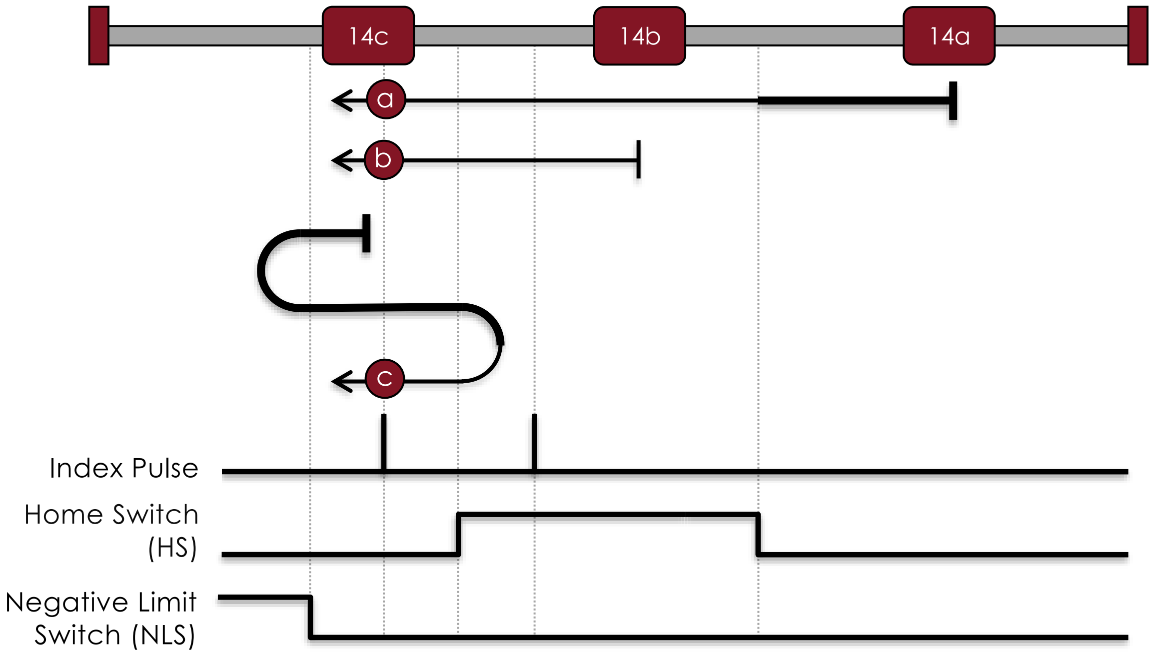


Figure 7 - 16. Homing Method 14

7.3.6. Method 15 and 16: Reserved

7.3.7. Method 17 to 30: Homing without Index Pulse

These methods are similar to methods 1 through 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 17 (Figure 7 - 17) and 18 (Figure 7 - 18) are similar to methods 1 (Figure 7 - 3) and 2 (Figure 7 - 4) respectively and so on.

Axis 17a moves in the negative direction until the negative limit switch (NLS) is active. The axis will then move in the positive direction until the NLS goes inactive. After this event, the drive sets *home attained*.

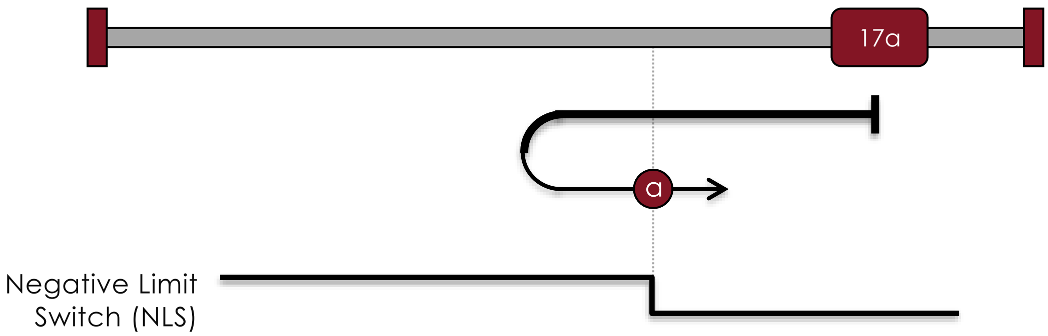


Figure 7 - 17. Homing Method 17

Axis 18a moves in the positive direction until the **positive limit switch** (PLS) is active. The axis will then move in the negative direction until the PLS goes inactive. After this event, the drive sets *home attained*.

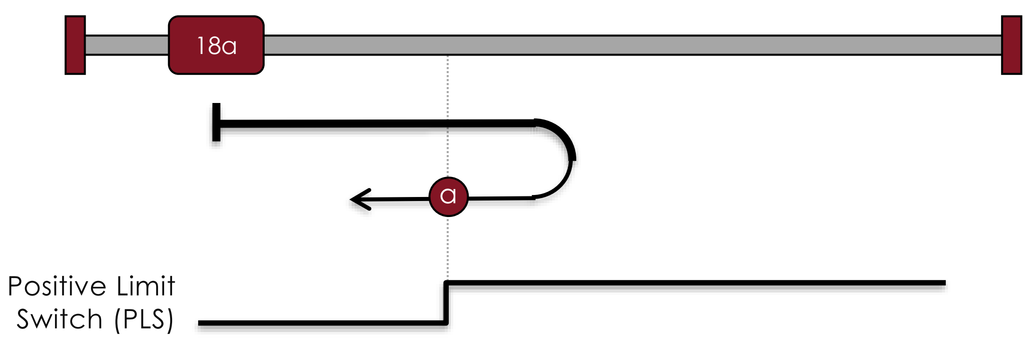


Figure 7 - 18. Homing Method 18

Using these methods as shown in Figure 7 - 19 and Figure 7 - 20 the initial direction of movement is dependent on the state of the home switch. The home attain position should be where the home switch changes states. If the initial position is situated so that the direction of movement needs to be reversed during homing, the point at which it reverses should be after the home switch changes states.

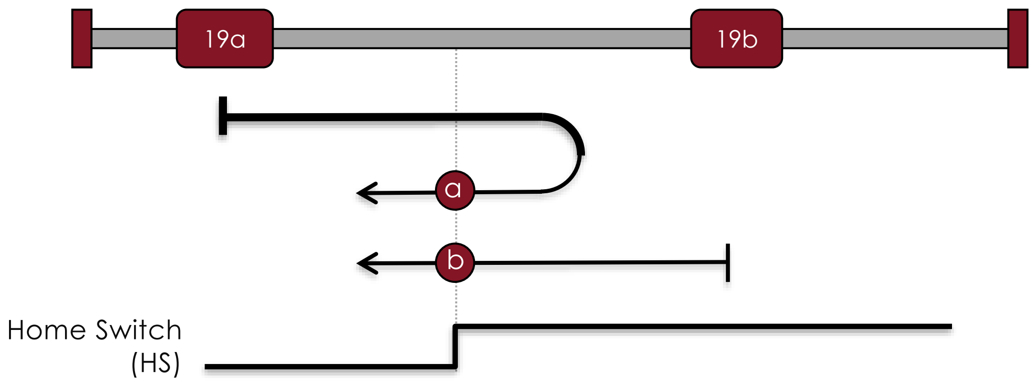


Figure 7 - 19. Homing Method 19

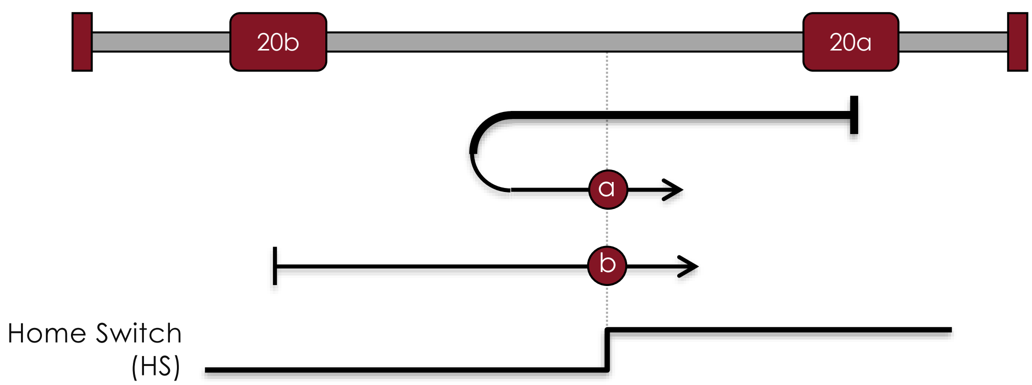


Figure 7 - 20. Homing Method 20

Using these methods as shown in Figure 7 - 21 and Figure 7 - 22 the initial direction of movement is dependent on the state of the home switch. The home attain position should be where the home switch changes states. If the initial position is situated so that the direction of movement needs to be reversed during homing, the point at which it reverses should be after the home switch changes states.

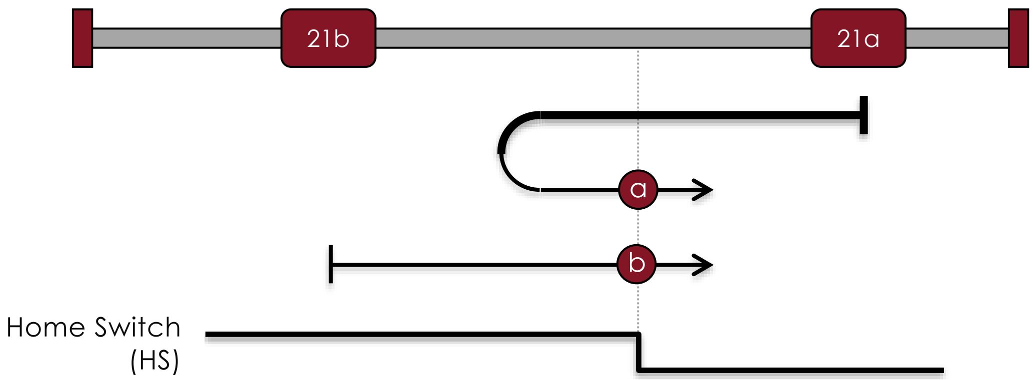


Figure 7 - 21. Homing Method 21

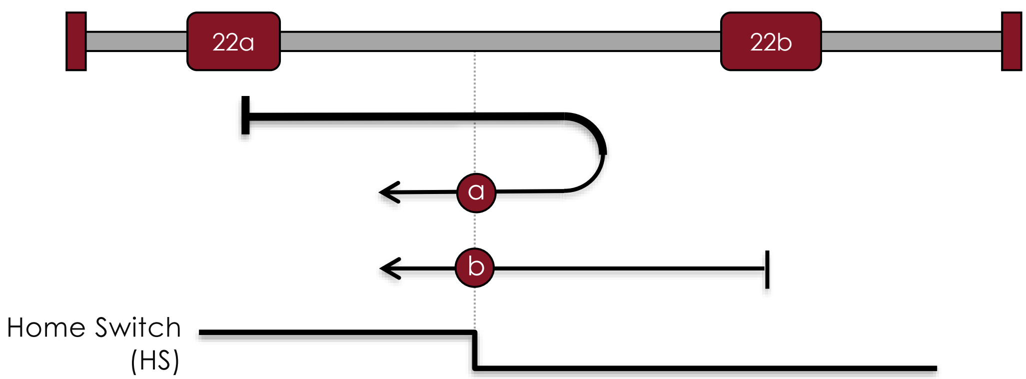


Figure 7 - 22. Homing Method 22

These methods use a home switch, which is active only a portion of the travel. The home switch has a 'momentary' action as the axis position goes pass it. Using methods 23 to 26, Figure 7 - 23 through Figure 7 - 26 respectively, the initial direction of movement is to the right. Using methods 27 to 30, Figure 7 - 27 through Figure 7 - 30 respectively, the initial direction of movement is to the left except if the home switch is active at the start of the motion. In such case, the initial direction of motion depends on the edge of interest. The home position is the rising or falling edges of the home switch. If the initial direction of movement leads away from the home switch, the drive should reverse on encountering the relevant limit switch.

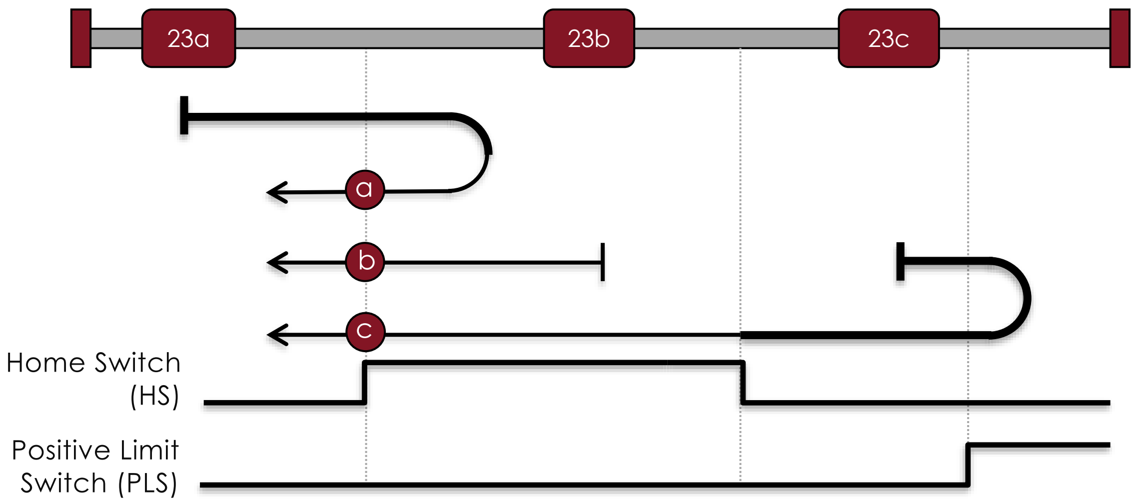


Figure 7 - 23. Homing Method 23

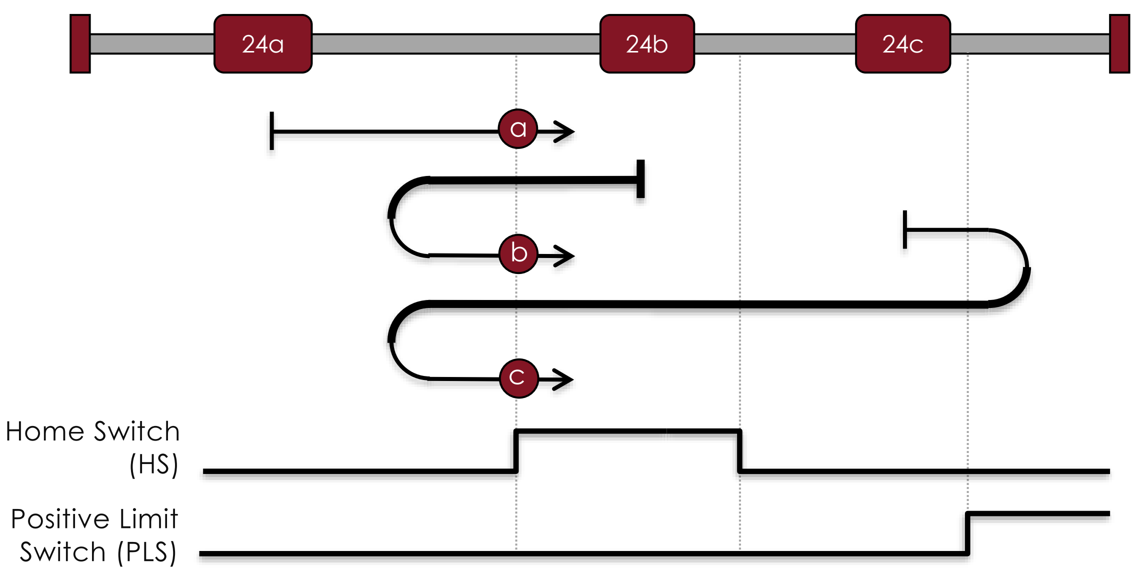


Figure 7 - 24. Homing Method 24

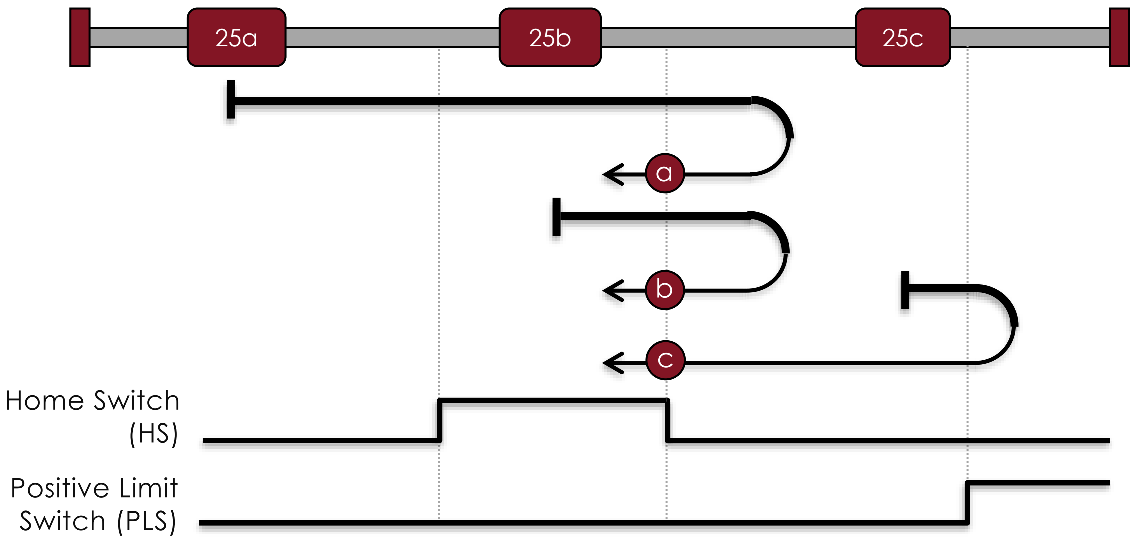


Figure 7 - 25. Homing Method 25

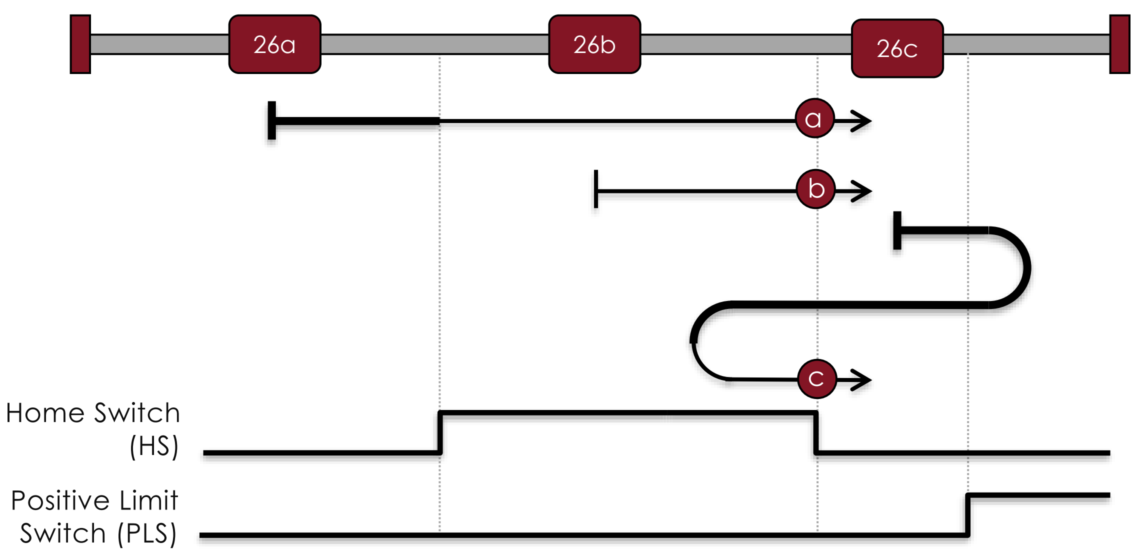


Figure 7 - 26. Homing Method 26

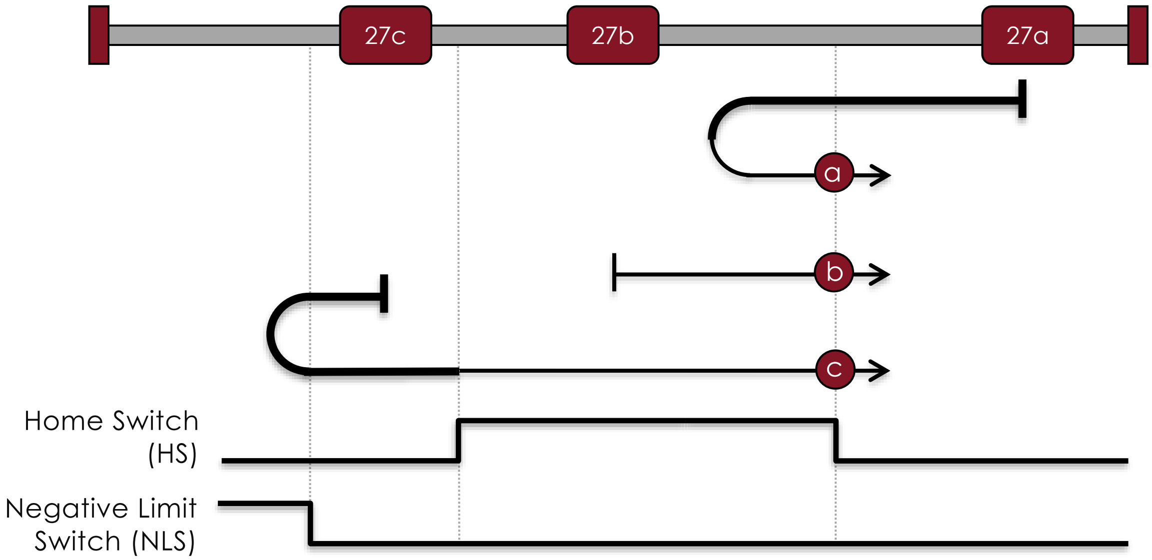


Figure 7 - 27. Homing Method 27

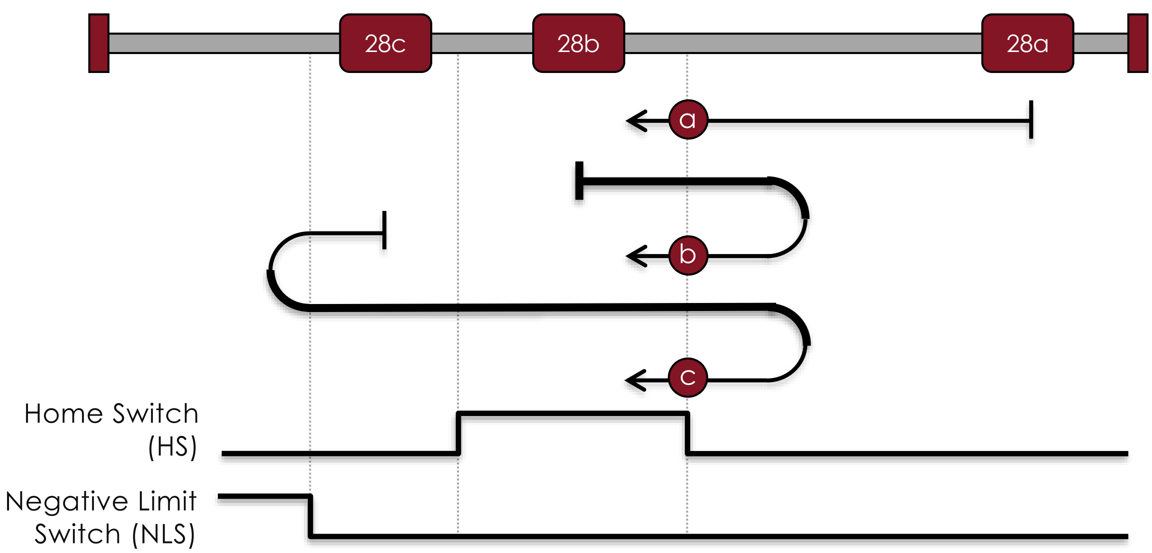


Figure 7 - 28. Homing Method 28

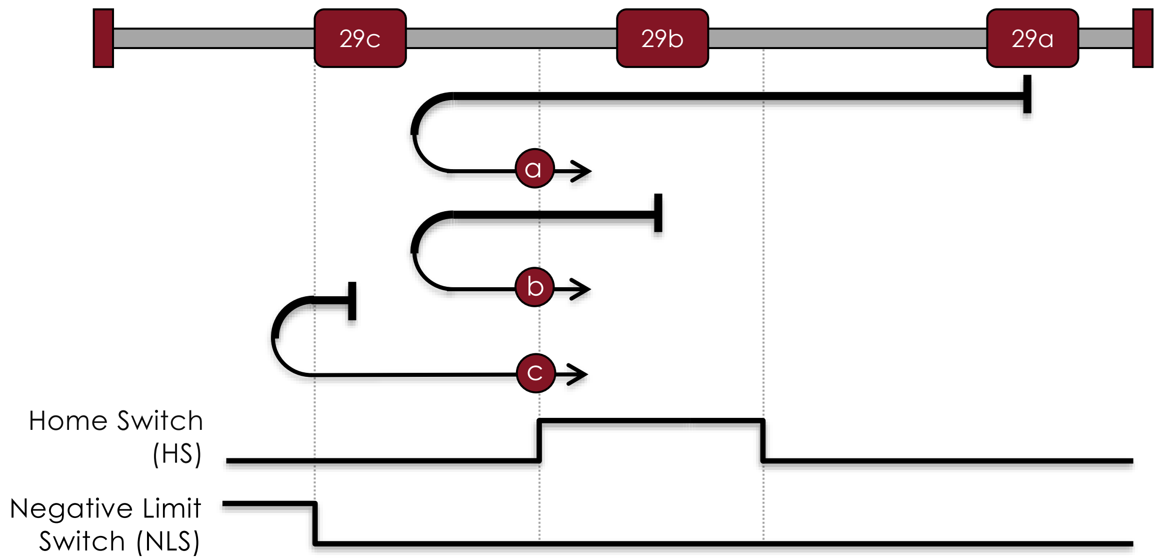


Figure 7 - 29. Homing Method 29

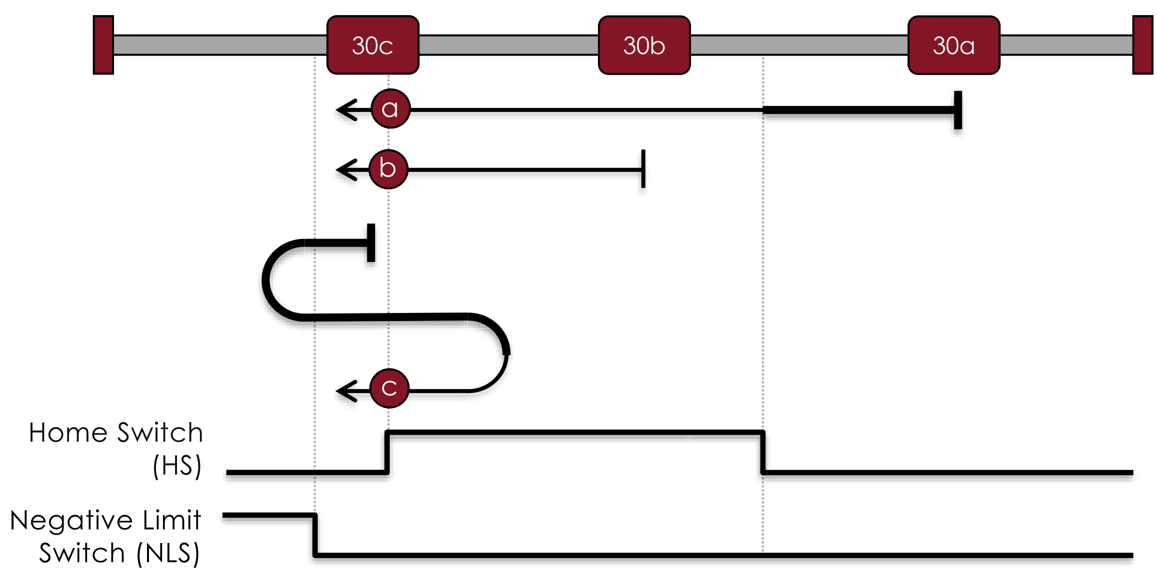


Figure 7 - 30. Homing Method 30

7.3.8. Method 31 and 32: Reserved

7.3.9. Method 33 and 34: Homing on Index Pulse

Using this method, the direction of homing is negative or positive respectively. The home position should be at the index pulse found in the appropriate direction as shown in Figure 7 - 31 and Figure 7 - 32.

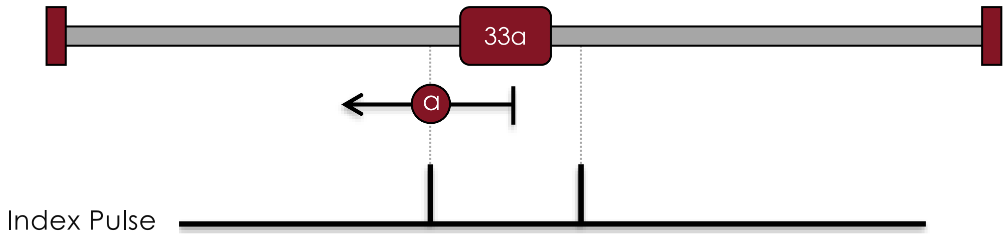


Figure 7 - 31. Homing Method 33

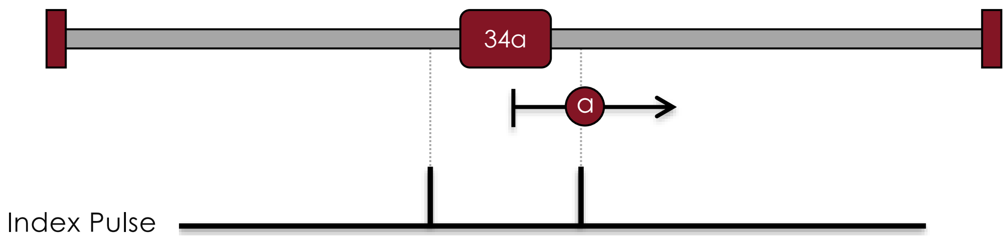


Figure 7 - 32. Homing Method 34

7.3.10. Method 35: Homing on the Current Position

In this method, the current position is taken to be the home position. This method does not require the drive to be in *operational enabled state.*

7.4. Use of Controlword and Statusword

The homing mode uses bits of the controlword and the statusword for mode-specific purposes. Figure 7 - 33 shows the structure of the controlword. Table 7 - 1 defines the values for bit 4 and 8 of the controlword.

15 9	8	7	6 5	4	3 0
(See 6.3.1)	Halt	(See 6.3.1)	Reserved (0)	Homing Operation Start	(See 6.3.1)
MSB					LSB

Figure 7 - 33. Controlword for Homing Mode

Table 7 - 1. Definition of Controlword bit 4 and bit 8

Bit	Value	Definition
4	0	Do not start homing procedure
	1	Start or continue homing procedures
8	0	Enable bit 4
	1	Stop axis according to halt option code (605Dh)

Figure 7 - 34 shows the structure of the statusword. Table 7 - 2 defines the values for bit 10, bit 12, and bit 13.

15 14	13	12	11	10	9 0
(See 6.3.2)	Homing Error	Homing Attained	(See 6.3.2)	Target Reached	(See 6.3.2)
MSB					LSB

Figure 7 - 34. Statusword for Homing Mode

Table 7 - 2. Definition of Statusword bit 10, bit 12, and bit 13

Bit 13	Bit 12	Bit 10	Definition
0	0	0	Homing procedure is in progress
0	0	1	Homing procedure is interrupted or not started
0	1	0	Homing is attained, but target is not reached
0	1	1	Homing procedure is completed successfully
1	0	0	Homing error occurred; velocity is not 0
1	0	1	Homing error occurred; velocity is 0
1	1	X	Reserved

7.5. Detailed object definitions

7.5.1. Object 607Ch: Home offset

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
607Ch	00h	Home Offset	0	No	Integer 32-bit	RW	Yes

Home offset is the amount, measured in position units, that the axis moves from the home position, attained during homing, to reach the zero position. The zero position is to be determined by the user depending on the application and set using the home offset. Once the home offsets are set and the zero position is determined, all absolute moves will then be relative to the zero position. If object is not implemented, the home offset will default to zero. This is illustrated by Figure 7 - 35.

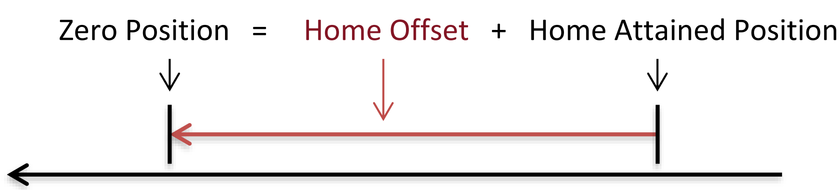


Figure 7 - 35. Setting Zero Position Utilizing Home Offset

7.5.2. Object 6098h: Homing Method

This object indicates the homing method that should be used.

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
6098h	00h	Homing Method	1	No	Integer 8-bit	RW	Yes

Table 7 - 3. 6098h Value Range Definition

Value	Definition
-128 to -1	Reserved
0	No homing method assigned
1 – 14	Homing combined with index pulse
15 – 16	Reserved
17 – 30	Homing without index pulse
31 – 32	Reserved
33 – 34	Homing on index pulse
35 – 127	Reserved

7.5.3. Object 6099h: Homing Speeds

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
6099h	00h	Number of Elements	02h	No	Unsigned 8-bit	C	No
6099h	01h	Speed during search for switch	2000*	No	Unsigned 32-bit	RW	Yes
6099h	02h	Speed during search for zero	1000*	No	Unsigned 32-bit	RW	Yes
*The units are in counts/second.							

This object defines the speeds use for homing. The value is in counts/second.

7.5.4. Object 609Ah: Homing Acceleration

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
609Ah	00h	Homing Acceleration	2000	No	Unsigned 32-bit	RW	Yes
The units are in counts/s^2.							

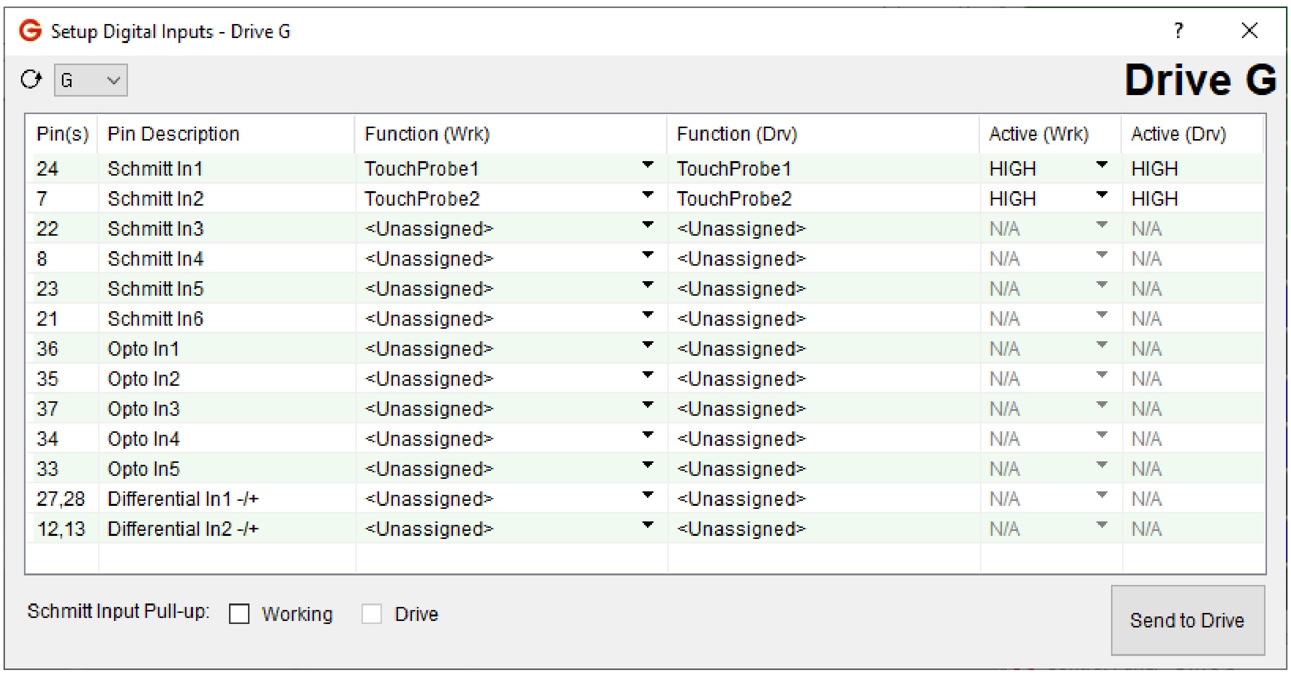
This object defines the acceleration used during the homing process. Deceleration is performed at the same value. The value is counts/second^2.

8. Touch Probe Functionality

8.1. Overview

The Touch Probe function is based on IEC 61800-7-201[8] provides the user can capture the position information when Touch Probe position condition is reached. The position can be latched at positive or negative edge or at pulse signal of encoder.

The two touch probes can be assigned as Digital inputs by the MotionMaestro. Open **Setup** > **Setup Digital Inputs** and then assign the touch probe 1 and 2 as Figure 8 - 1. Any input pins are available to assign for Touch Probe.


Setup Digital Inputs - Drive G

Drive G

Pin(s)	Pin Description	Function (Wrk)	Function (Drv)	Active (Wrk)	Active (Drv)
24	Schmitt In1	TouchProbe1	TouchProbe1	HIGH	HIGH
7	Schmitt In2	TouchProbe2	TouchProbe2	HIGH	HIGH
22	Schmitt In3	<Unassigned>	<Unassigned>	N/A	N/A
8	Schmitt In4	<Unassigned>	<Unassigned>	N/A	N/A
23	Schmitt In5	<Unassigned>	<Unassigned>	N/A	N/A
21	Schmitt In6	<Unassigned>	<Unassigned>	N/A	N/A
36	Opto In1	<Unassigned>	<Unassigned>	N/A	N/A
35	Opto In2	<Unassigned>	<Unassigned>	N/A	N/A
37	Opto In3	<Unassigned>	<Unassigned>	N/A	N/A
34	Opto In4	<Unassigned>	<Unassigned>	N/A	N/A
33	Opto In5	<Unassigned>	<Unassigned>	N/A	N/A
27,28	Differential In1 -/+	<Unassigned>	<Unassigned>	N/A	N/A
12,13	Differential In2 -/+	<Unassigned>	<Unassigned>	N/A	N/A

Schmitt Input Pull-up: Working Drive

Send to Drive

Figure 8 - 1. Example of Assigning Touch Probe 1 and 2

8.1.1. Object 60B8h: Touch Probe Function

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60B8h	00h	Touch Probe Function	0h	Yes	Unsigned 16-bit	RW	Yes

This object indicates the configured function of the touch probe. Table 8 - 1 specifies the value definition.

Table 8 - 1. Value Definition of 60B8h

Bit	Value	Definition
0	0	Disable touch probe 1
	1	Enable touch probe 1
1	0	Single Event Trigger
	1	Continuous Trigger
3, 2	00b	Trigger with touch probe 1 input
	01b	Trigger with zero pulse of encoder
	10b	Reserved
	11b	Reserved
4	0	Disable sampling at positive edge of touch probe 1
	1	Enable sampling at positive edge of touch probe 1
5	0	Disable sampling at negative edge of touch probe 1
	1	Enable sampling at negative edge of touch probe 1
6, 7	-	Reserved
8	0	Disable touch probe 2
	1	Enable touch probe 2
9	0	Single Event Trigger
	1	Continuous Trigger
11, 10	00b	Trigger with touch probe 2 input
	01b	Trigger with zero pulse of encoder
	10b	Reserved
	11b	Reserved
12	0	Disable sampling at positive edge of touch probe 2
	1	Enable sampling at positive edge of touch probe 2
13	0	Disable sampling at negative edge of touch probe 2
	1	Enable sampling at negative edge of touch probe 2
14, 15	-	Reserved

8.1.2. Object 60B9h: Touch Probe Status

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60B9h	00h	Touch Probe Status	0h	Yes	Unsigned 16-bit	RO	No

This object provides the status of the touch probe. Table 8 - 2 specifies the value definition.

Table 8 - 2. Value Definition of 60B9h

Bit	Value	Definition
0	0	Touch probe 1 is disabled
	1	Touch probe 1 is enabled
1	0	No touch probe 1 positive edge value captured
	1	Touch probe 1 positive edge position captured
2	0	No touch probe 1 negative edge value captured
	1	Touch probe 1 negative edge position captured
3 to 5	0	Reserved
6	0	No encoder zero pulse position captured
	1	Encoder zero pulse position captured
7	-	Reserved
8	0	Touch probe 2 is disabled
	1	Touch probe 2 is enabled
9	0	No touch probe 2 positive edge value captured
	1	Touch probe 2 positive edge position captured
10	0	No touch probe 2 negative edge value captured
	1	Touch probe 2 negative edge position captured
11 to 13	0	Reserved
14	0	No encoder zero pulse position captured
	1	Encoder zero pulse position captured
15	-	Reserved
NOTE Bit 1 and bit 2 are set to 0b when touch probe 1 is switched off (object 60B8h bit 0 is 0b). Bit 9 and 10 are set to 0b when touch probe 2 is switched off (object 60B8h bit 8 is 0b).		

8.1.3. Object 60BAh: Touch Probe 1 Positive Edge / Encoder Zero Pulse Position

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60BAh	00h	Touch Probe 1 Positive Edge / Encoder Zero Pulse Position	0h	Yes	Signed 32-bit	RO	No
The units are in counts.							

Table 8 - 3. Value Definition of 60BAh

60B8h	60BAh
xxxx xxxx xxx1 00x1b	Touch Probe 1 Positive Edge Captured
xxxx xxxx xxx1 01x1b	Encoder Zero Pulse Position Captured

8.1.4. Object 60BBh: Touch Probe 1 Negative Edge

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60BBh	00h	Touch Probe 1 Negative Edge	0h	Yes	Signed 32-bit	RO	No
The units are in counts.							

This object provides the position value of the touch probe 1 at negative edge. The units for the value are in counts.

8.1.5. Object 60BCh: Touch Probe 2 Positive Edge / Encoder Zero Pulse Position

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60BCh	00h	Touch Probe 2 Positive Edge / Encoder Zero Pulse Position	0h	Yes	Signed 32-bit	RO	No
The units are in counts.							

Table 8 - 4. Value Definition of 60BCh

60B8h	60BCh
xx1x 00x1 xxxx xxxxb	Touch Probe 2 Positive Edge Captured
xx1x 01x1 xxxx xxxxb	Encoder Zero Pulse Position Captured

8.1.6. Object 60BDh: Touch Probe 2 Negative Edge

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60BDh	00h	Touch Probe 2 Negative Edge	0h	Yes	Signed 32-bit	RO	No
The units are in counts.							

This object provides the position value of the touch probe 2 at negative edge. The units for the value are in counts.

8.2. Touch Probe Time Stamp Latch

The objects 60D1h, 60D2h, 60D3h and 60D4h provide the corresponding time stamp for touch probe 1, 2 or encoder zero pulse capture.

8.2.1. Object 60D1h: Touch Probe 1 Positive Edge Time Stamp / Encoder Zero Pulse Time Stamp

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60D1h	00h	Touch Probe 1 Positive Edge Time Stamp / Encoder Zero Pulse Time Stamp	0h	Yes	Unsigned 32-bit	RO	No
The units are in micro-seconds.							

Table 8 - 5. Value Definition of 60D1h

60B8h	60D1h
xxxx xxxx xxx1 00x1b	Touch Probe 1 Positive Edge Time Stamp Captured
xxxx xxxx xxx1 01x1b	Encoder Zero Pulse Time Stamp Captured

8.2.2. Object 60D2h: Touch Probe 1 Negative Edge Time Stamp

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60D2h	00h	Touch Probe 1 Negative Edge Time Stamp	0h	Yes	Unsigned 32-bit	RO	No
The units are in micro-seconds.							

This object provides the time stamp value of the touch probe 1 at negative edge. The units for the value are in micro-seconds.

8.2.3. Object 60D3h: Touch Probe 2 Positive Edge Time Stamp / Encoder Zero Pulse Position Time Stamp

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60D3h	00h	Touch Probe 2 Positive Edge Time Stamp / Encoder Zero Pulse Position Time Stamp	0h	Yes	Unsigned 32-bit	RO	No
The units are in micro-seconds.							

Table 8 - 6. Value Definition of 60D3h

60B8h	60D3h
xx1x 00x1 xxxx xxxxb	Touch Probe 2 Positive Edge Time Stamp Captured
xx1x 01x1 xxxx xxxxb	Encoder Zero Pulse Time Stamp Captured

8.2.4. Object 60D4h: Touch Probe 2 Negative Edge Time Stamp

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60D4h	00h	Touch Probe 2 Negative Edge Time Stamp	0h	Yes	Unsigned 32-bit	RO	No
The units are in micro-seconds.							

This object provides the time stamp value of the touch probe 2 at negative edge. The units for the value are in micro-seconds.

8.3. Touch Probe Edge Counter for Continuous Mode

For continuous touch probe mode (60B8h, bit 1 = 1b or 60B8h, bit 9 = 1b) a counter per touch probe channel is incremented on each touch probe event. Thus, the control device may check how many touch probe events happen between the control cycles. Per touch probe and per edge a counter object (objects 60D5h, 60D6h, 60D7h and 60D8h) is defined.

8.3.1. Object 60D5h: Touch Probe 1 Positive Edge Counter / Encoder Zero Pulse Counter

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60D5h	00h	Touch Probe 1 Positive Edge Counter / Encoder Zero Pulse Counter	0h	Yes	Unsigned 16-bit	RO	No

Table 8 - 7. Value Definition of 60D5h

60B8h	60D5h
xxxx xxxx xxx1 0001b	Touch Probe 1 Positive Edge Counter Captured for single event measuring
xxxx xxxx xxx1 0101b	Encoder Zero Pulse Counter Captured for single event measuring
xxxx xxxx xxx1 0011b	Touch Probe 1 Positive Edge Counter Captured for continuous measuring
xxxx xxxx xxx1 0111b	Encoder Zero Pulse Counter Captured for continuous measuring

8.3.2. Object 60D6h: Touch Probe 1 Negative Edge Counter

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60D6h	00h	Touch Probe 1 Negative Edge Counter	0h	Yes	Unsigned 16-bit	RO	No

Table 8 - 8. Value Definition of 60D6h

60B8h	60D6h
xxxx xxxx xxx1 0001b	Touch Probe 1 Negative Edge Counter Captured for single event measuring
xxxx xxxx xxx1 0011b	Touch Probe 1 Negative Edge Counter Captured for continuous measuring

8.3.3. Object 60D7h: Touch Probe 2 Positive Edge Counter / Encoder Zero Pulse Counter

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60D7h	00h	Touch Probe 2 Positive Edge Counter / Encoder Zero Pulse Counter	0h	Yes	Unsigned 16-bit	RO	No

Table 8 - 9. Value Definition of 60D7h

60B8h	60D7h
xxx1 0001 xxxx xxxxb	Touch Probe 2 Positive Edge Counter Captured for single event measuring
xxx1 0101 xxxx xxxxb	Encoder Zero Pulse Counter Captured for single event measuring
xxx1 0011 xxxx xxxxb	Touch Probe 2 Positive Edge Counter Captured for continuous measuring
xxx1 0111 xxxx xxxxb	Encoder Zero Pulse Counter Captured for continuous measuring

8.3.4. Object 60D8h: Touch Probe 2 Negative Edge Counter

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60D8h	00h	Touch Probe 2 Negative Edge Counter	0h	Yes	Unsigned 16-bit	RO	No

Table 8 - 10. Value Definition of 60D8h

60B8h	60D8h
xxx1 0001 xxxx xxxxb	Touch Probe 2 Negative Edge Counter Captured for single event measuring
xxx1 0011 xxxx xxxxb	Touch Probe 2 Negative Edge Counter Captured for continuous measuring

8.4. Timing Diagram for Touch Probe Example

Figure 8 - 2 [8] shows a timing diagram for an example touch probe configuration and the corresponding behavior. Table 8 - 11 [8] explains the timing diagram.

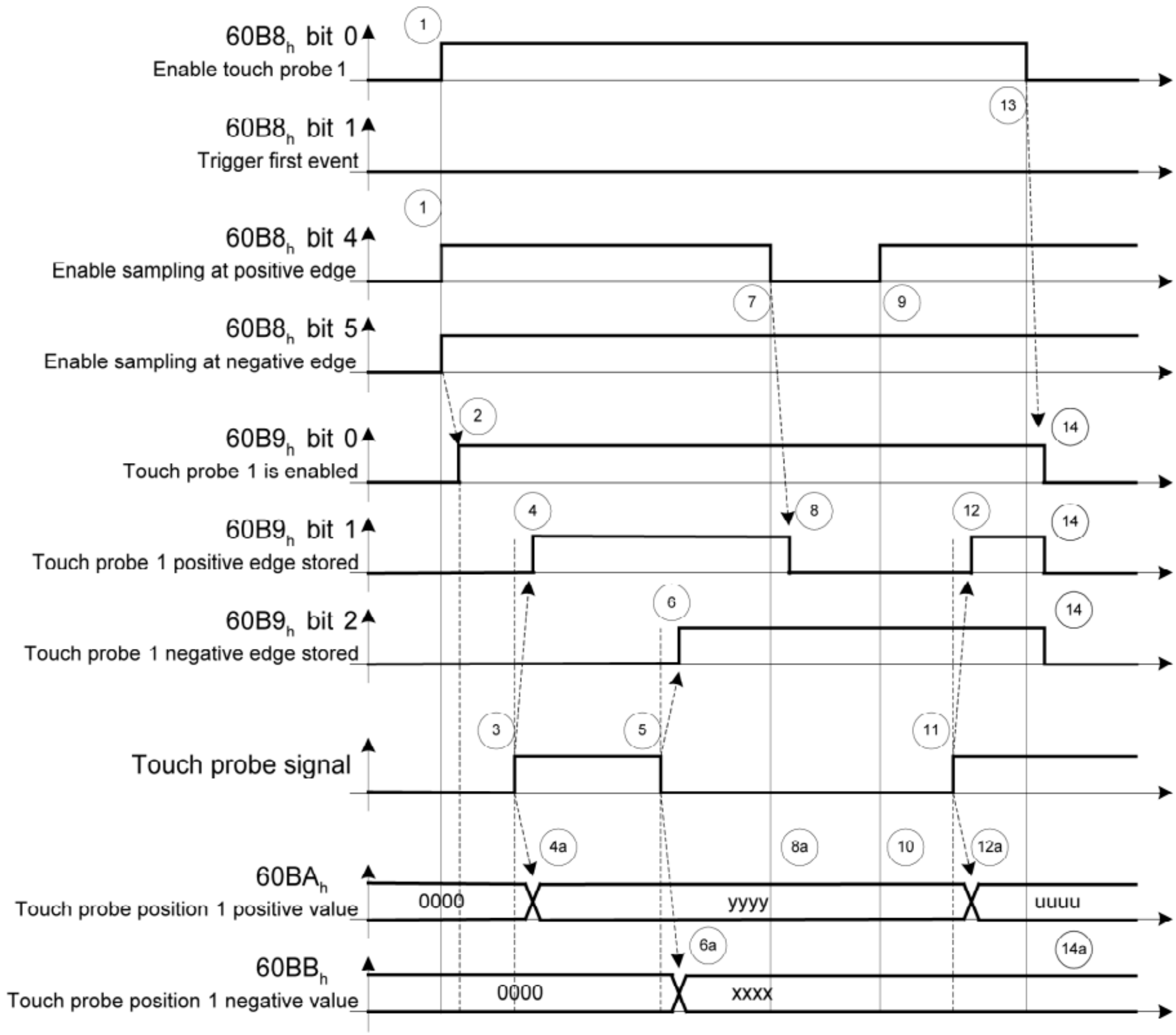


Figure 8 - 2. Timing Diagram for Touch Probe Example

Table 8 - 11. Explanation of the Timing Diagram

Number	Touch Probe Behavior
(1)	60B8h, bit 0 = 1b Enable touch probe 1 60B8h, bit 1, 4, 5 Configure and enable touch probe 1 positive and negative edge
(2)	→ 60B9h, bit 0 = 1b Status "Touch probe 1 enabled" is set
(3)	External touch probe signal has positive edge
(4)	→ 60B9h, bit 1 = 1b Status "Touch probe 1 positive edge stored" is set
(4a)	→ 60Bah Touch probe position 1 positive value is stored
(5)	External touch probe signal has negative edge
(6)	→ 60B9h, bit 2 = 1b Status "Touch probe 1 negative edge stored" is set
(6a)	→ 60BBh Touch probe position 1 negative value is stored
(7)	60B8h, bit 4 = 0b Sample positive edge is disabled
(8)	→ 60B9h, bit 0 = 0b Status "Touch probe 1 positive edge stored" is reset
(8a)	→ 60BAh Touch probe position 1 positive value is not changed
(9)	60B8h, bit 4 = 1b Sample positive edge is enabled
(10)	→ 60BAh Touch probe position 1 positive value is not changed
(11)	External touch probe signal has positive edge
(12)	→ 60B9h, bit 1 = 1b Status "Touch probe 1 positive edge stored" is set
(12a)	→ 60BAh Touch probe position 1 positive value is stored
(13)	60B8h, bit 0 = 0b Touch probe 1 is disabled
(14)	→ 60B9h, bit 0, 1, 2 = 0b Status bits are reset
(14a)	→ 60BAh, 60BBh Touch probe position 1 positive/negative value are not changed

9. Cyclic Synchronous Position Mode

9.1. Overview

The overall structure for this mode is shown in Figure 9 - 1. With CSP mode, the position profile generator is located in the control. In cyclic synchronous manner, it provides a target position to the drive, which performs position control, velocity control, and torque/current control. A velocity and torque offsets are provided by the control system in order to allow for velocity and/or torque feedforward. The drive provides actual values for position, velocity, and torque to the control.

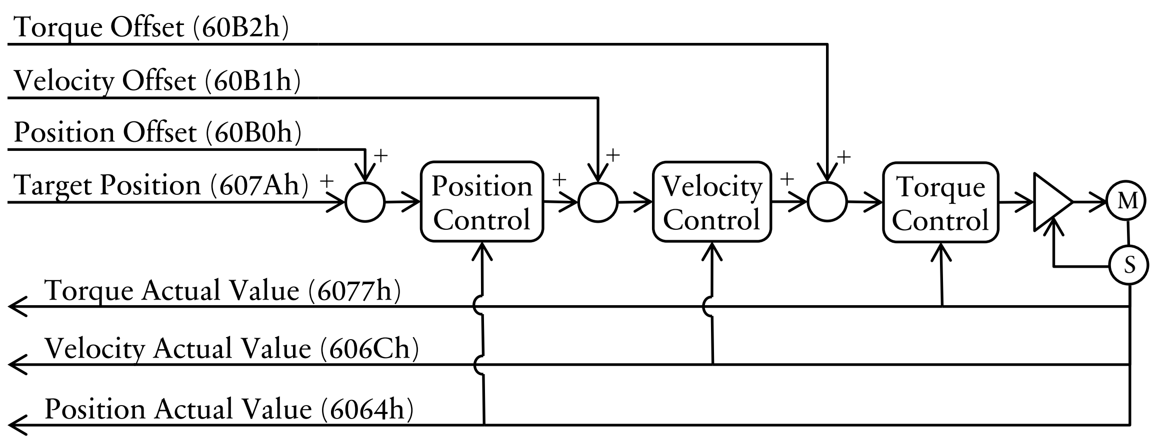
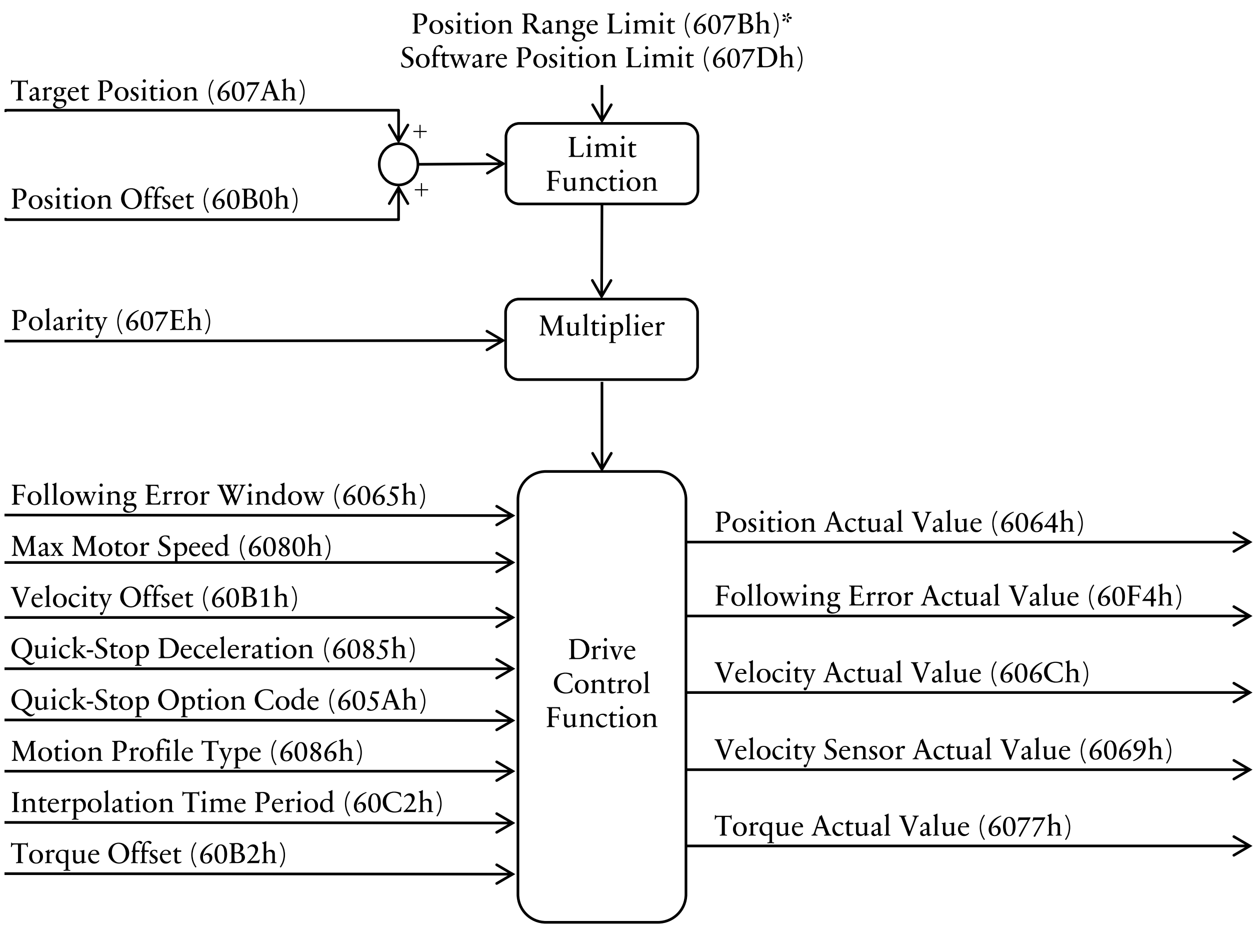


Figure 9 - 1. Cyclic Synchronous Position Mode Overview

9.2. Functional Description

Figure 9 - 2: Cyclic Synchronous Position Control Function shows the inputs and outputs of the drive control function. The target position is an absolute value.



*Object is in the process of being written. Please consult Glentek if needed.

Figure 9 - 2. Cyclic Synchronous Position Control Function

9.3. Use of Controlword and Statusword

The cyclic synchronous position mode uses Controlword (See Figure 6 - 2). Figure 9 - 3 shows CSP mode Statusword. Table 9 - 1 defines the values of bit 12 and 13 of the Statusword.

15 14	13	12	11	10	9 0
(See 6.3.2)	Following Error	Target Position Ignored	(See 6.3.2)	Reserved	(See 6.3.2)
MSB					LSB

Figure 9 - 3. Statusword for Profile Cyclic Synchronous Position Mode

Table 9 - 1. Definition of Statusword bit 12 and bit 13

Bit	Value	Definition
12	0	Target position ignored
	1	Target position is used as input to the position control loop
13	0	No following error
	1	Following error

9.4. Detailed Object Definitions

9.4.1. Object 6064h: Position Actual Value

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
6064h	00h	Position Actual Value	-	Yes	Integer 32-bit	RO	No
The units are in Counts.							

This object shows the actual position of motor, units are encoder counts.

9.4.2. Object 6065h: Following Error Window

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
6065h	00h	Following Error Window	-	No	Unsigned 32-bit	RW	Yes

This object defines range for tolerated position following error. In the case actual position value is out of the following error window, a following error occurs (Bit 13 of Statusword set to 1). The default value is determined by setting 'Position following error' on MotionMaestro.

9.4.3. Object 6069h: Velocity Sensor Actual Value

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
6069h	00h	Velocity Sensor Actual Value	-	Yes	Integer 32-bit	RO	No
The units are in counts/second.							

This object provides the value read from a velocity sensor. The value is given in counts per second.

9.4.4. Object 607Ah: Target Position

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
607Ah	00h	Target Position	0h	Yes	Integer 32-bit	RW	No

This object defines the commanded position that the drive should move to in position profile mode in units of encoder counts. This object is interpreted as absolute or relative depending on "abs/rel" flag (Bit 6) of Object 6040h: Controlword is set.

9.4.5. Object 607Dh: Software Position Limit

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
607Dh	00h	Software Position Limit	02h	No	Unsigned 8-bit	RO	No
607Dh	01h	Min Position Limit	2^{31}	No	Integer 32-bit	RW	Yes
607Dh	02h	Max Position Limit	$2^{31}-1$	No	Integer 32-bit	RW	Yes

This object defines the maximal and minimal position limits for Object 6064h: Position Actual Value in units of encoder counts. These limits are active only after the drive has been referenced (Homing successful).

9.4.6. Object 607Eh: Polarity

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
607Eh	00h	Polarity	0h	No	Unsigned 8-bit	RW	Yes

This object indicates if the position demand value shall be multiplied by 1 of by -1. The polarity flag has no influence on homing mode. The Position Polarity bit is used for profile position and cyclic synchronous position mode. The velocity polarity bit is used for profile velocity and cyclic synchronous velocity mode. Polarity in a position or velocity profile is chosen according to Table 9 - 2.

7	6	5 … 0
Position Polarity	Velocity Polarity	Reserved (0)
MSB		LSB

Figure 9 - 4. Object 607Eh Value Definition

Table 9 - 2. Definition of Bits 6 and 7 of 607Eh

Bit	Value	Definition
6	0	Multiply by 1
	1	Multiply by -1
7	0	Multiply by 1
	1	Multiply by -1

9.4.7. Object 6080h: Max Motor Speed

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
6080h	00h	Max Motor Speed	–	No	Unsigned 32-bit	RW	Yes
The units are in counts/s.							

This object indicates the configured maximal allowed speed for the motor in either direction. The default value is determined by setting 'Overspeed threshold' on MotionMaestro.

9.4.8. Object 6085h: Quick Stop Deceleration

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
6085h	00h	Quick Stop Deceleration	0h	No	Unsigned 32-bit	RW	Yes
The units are in counts/s^2.							

9.4.9. Object 6086h: Motion Profile Type

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
6086h	00h	Motion Profile Type	0h	No	Integer 16-bit	RW	Yes

9.4.10. Object 60B0h: Position Offset

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60B0h	00h	Position Offset	0h	Yes	Integer 32-bit	RW	No
The units are in counts.							

This object defines the offset of the target position. The offset is given in counts. The value is absolute and it's independent of how often it is transmitted over the communication system. Transmitting the value twice does not mean a double value. The object is an offset to the target position, it is used to control the drive with relative values with regards to the target position.

9.4.11. Object 60B1h: Velocity Offset

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60B1h	00h	Velocity Offset	0h	Yes	Integer 32-bit	RW	No
The units are in counts/s.							

This object defines the offset for the velocity. The offset is given in counts/second. In cyclic synchronous position mode, this object contains the input value for velocity feedforward. The value is absolute, and it is independent of how often it is transmitted over the communication system. Transmitting the value twice does not mean a double value. The object is an offset to the target velocity, it is used to control the drive with relative values with regards to the target velocity.

9.4.12. Object 60B2h: Torque Offset

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60B2h	00h	Torque Offset	0h	Yes	Integer 16-bit	RW	No
The units are in drive rated torque/1000.							

This object defines the offset for the torque. The offset is given in per thousand drive rated torque. For example, the value 1000(=03E8h) means 100.0% of drive rated torque. In cyclic synchronous position mode and cyclic synchronous velocity mode, the torque offset is the input value for torque feedforward. The value is absolute, and it is independent of how often it is transmitted over the communication system. Transmitting the value twice does not mean a double value.

9.4.13. Object 60F4h: Following Error Actual Value

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60F4h	00h	Following Error Actual Value	-	Yes	Integer 32-bit	RO	No
The units are in counts.							

This object reports the difference: Position Demand Value – Position Actual Value.

10. Cyclic Synchronous Velocity Mode

10.1. Overview

The CSV mode structure is shown in Figure 10 - 1. With this mode, the position profile generator is located in the control. In cyclic synchronous manner, it provides a target velocity to the drive, which performs velocity control, and torque control. Velocity and torque offset values are provided by the control system to provide a second source for velocity and/or torque feedforward. The drive provides actual values for position, velocity, and torque to the control.

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

- Demand value input
- Velocity capture using position sensor or velocity sensor
- Velocity control function with appropriate input and output signals
- Limitation of torque demand

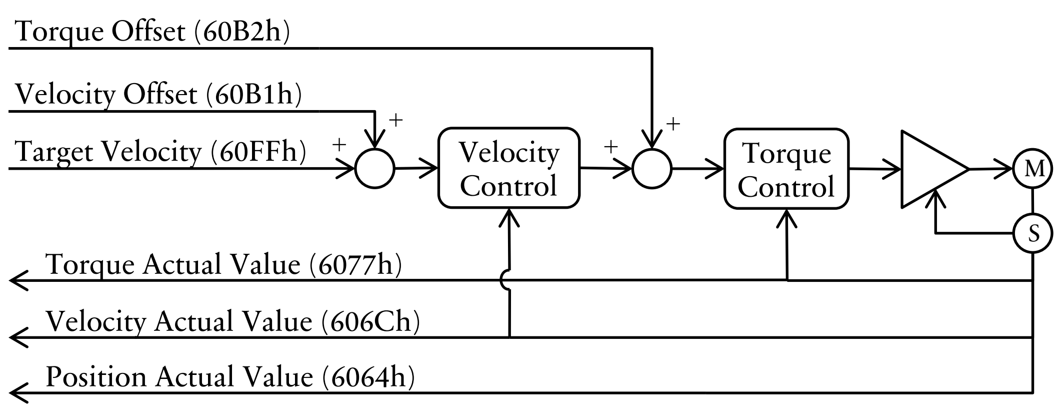


Figure 10 - 1. Cyclic Synchronous Velocity Mode Overview

10.2. Functional Description

Figure 10 - 2 shows the inputs and outputs of the drive control function.

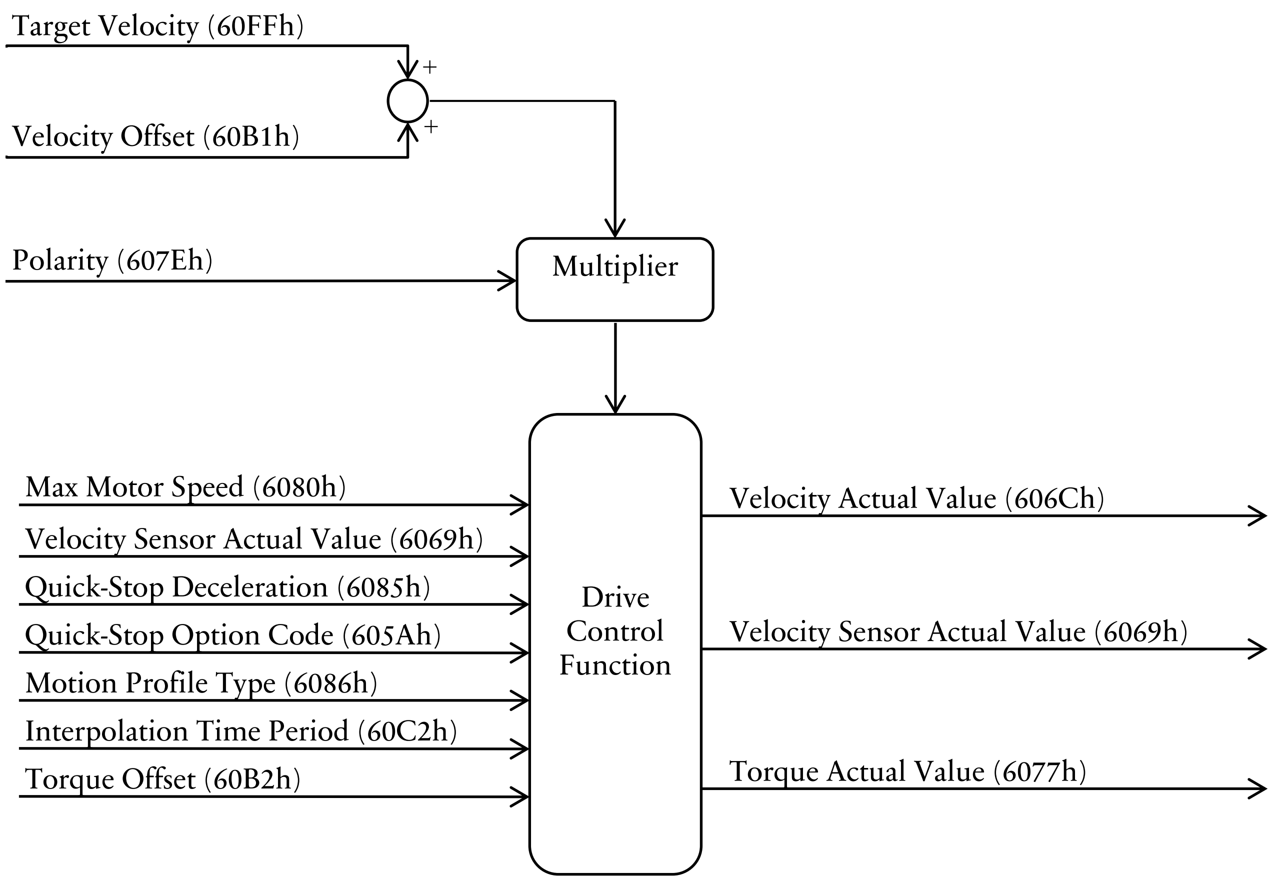


Figure 10 - 2. Cyclic Synchronous Velocity Control Function

10.3. Use of Controlword and Statusword

The cyclic synchronous velocity mode uses Controlword (See Figure 6 - 2). Figure 10 - 3 shows the CSV mode Statusword. Table 10 - 1 defines the values of bit 12 of the Statusword.

15 14	13	12	11	10	9 0
(See 6.3.2)	Reserved	Target Velocity Ignored	(See 6.3.2)	Reserved	(See 6.3.2)
MSB					LSB

Figure 10 - 3. Statusword for Profile Cyclic Synchronous Velocity Mode

Table 10 - 1. Definition of Statusword bit 12

Bit	Value	Definition
12	0	Target velocity ignored
	1	Target velocity is used as input to the velocity control loop

10.4. Detailed Object Definitions

10.4.1. Object 606Ch: Velocity Actual Value

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
606Ch	00h	Velocity Actual Value	-	Yes	Integer 32-bit	RO	No
The units are in counts/s.							

This object provides the actual velocity value derived from the velocity sensor or the position sensor. The value is given in counts/second.

10.4.2. Object 60FFh: Target Velocity

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60FFh	00h	Target Velocity	0h	Yes	Integer 32-bit	RW	No
The units are in counts/s.							

This object is used as target velocity input. The value is given in counts/second.

11. Cyclic Synchronous Torque Mode

11.1. Overview

The CST mode structure is shown in Figure 11 - 1. With this mode, the position profile generator is located in the control. In cyclic synchronous manner, it provides a target torque to the drive, which performs torque control. Torque offset value is provided by the control system to provide a second source for torque feedforward. The drive provides actual values for position, velocity, and torque to the control.

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

- Demand value input
- Torque capture
- Torque control function with appropriate input and output signals
- Limitation of torque demand

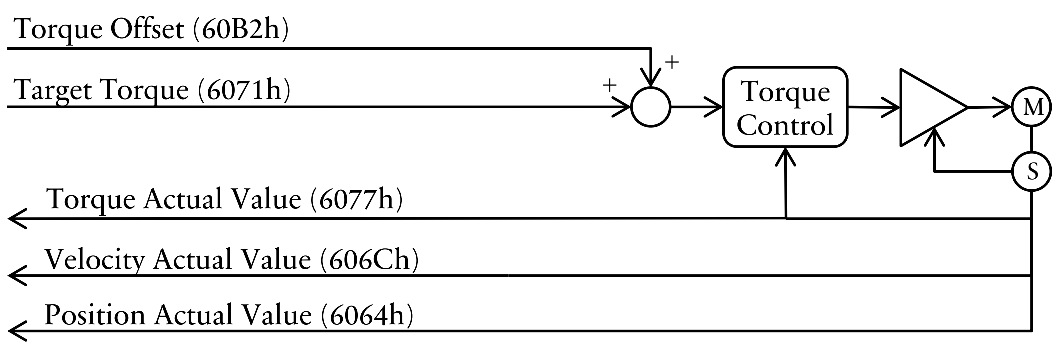


Figure 11 - 1. Cyclic Synchronous Torque Mode Overview

11.2. Functional Description

Figure 11 - 2 shows the inputs and outputs of the drive control function.

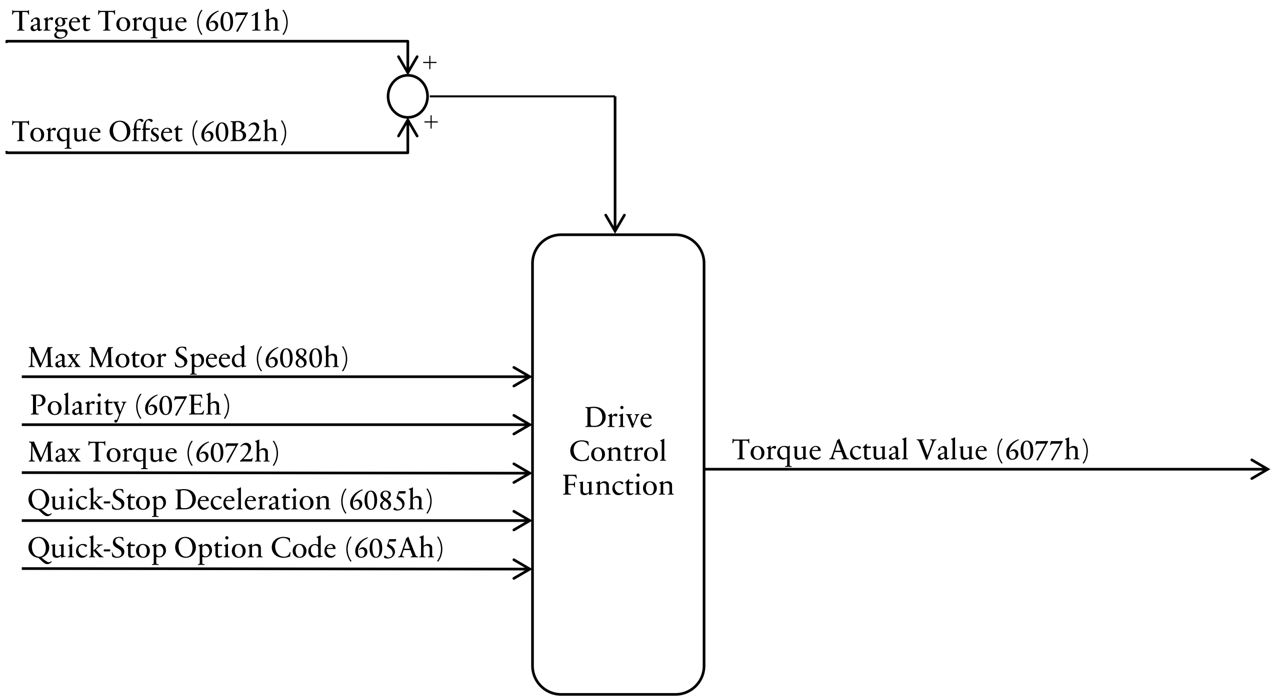


Figure 11 - 2. Cyclic Synchronous Torque Control Function

11.3. Use of Controlword and Statusword

The cyclic synchronous torque mode uses Controlword (See Figure 6 - 2). Figure 11 - 3 shows the CST mode Statusword. Table 11 - 1 defines the values of bit 12 of the Statusword.

15 14	13	12	11	10	9 0
(See 6.3.2)	Reserved	Drive follows the command value	(See 6.3.2)	Reserved	(See 6.3.2)
MSB					LSB

Figure 11 - 3. Statusword for Profile Cyclic Synchronous Torque Mode

Table 11 - 1. Definition of Statusword bit 12

Bit	Value	Definition
12	0	Target torque ignored
	1	Target torque is used as input to the torque control loop

11.4. Detailed Object Definitions

11.4.1. Object 6071h: Target Torque

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
6071h	00h	Target Torque	0h	Yes	Integer 16-bit	RW	No
The units are in drive rated torque/1000.							

The value is given per thousand of drive rated torque. For example, the value 1000(=03E8h) means 100.0% of drive rated torque.

11.4.2. Object 6077h: Torque Actual Value

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
6077h	00h	Torque Actual Value	-	Yes	Integer 16-bit	RO	No
The units are in drive rated torque/1000.							

The value is given per thousand of drive rated torque. For example, the value 1000(=03E8h) means 100.0% of drive rated torque.

11.4.3. Object 6072h: Max Torque

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
6072h	00h	Max Torque	03E8h	Yes	Unsigned 16-bit	RW	Yes
The units are in drive rated torque/1000.							

The default value 1000(=03E8h) means 100.0% of drive rated torque. The default value is determined by setting ‘Current Limit’ on MotionMaestro.

12. Slave Information

The EtherCAT slave information (ESI) file (XML format) is used to configure and communicate the Gamma by the EtherCAT Master. The ESI file describes the EtherCAT slave features. The ESI file (that can be changed after updating) is:

GTK_9GEXX_90700100_Vxxx.xml

The EtherCAT slave information after opening it by XML Notepad is shown in Figure 12 - 1. The Vendor ID is 0x2C0, the Vendor Name is Glentek, the Group Name is Gamma Series, and the Device Name is 9GEXX. 4 x Sync Manager (SM) is used. 2 x SMs are for Mailbox and others are for Process Data. There are 4 x RxPDO and 4 x TxPDO for Process Data exchange. The CoE is used for SDO exchange.

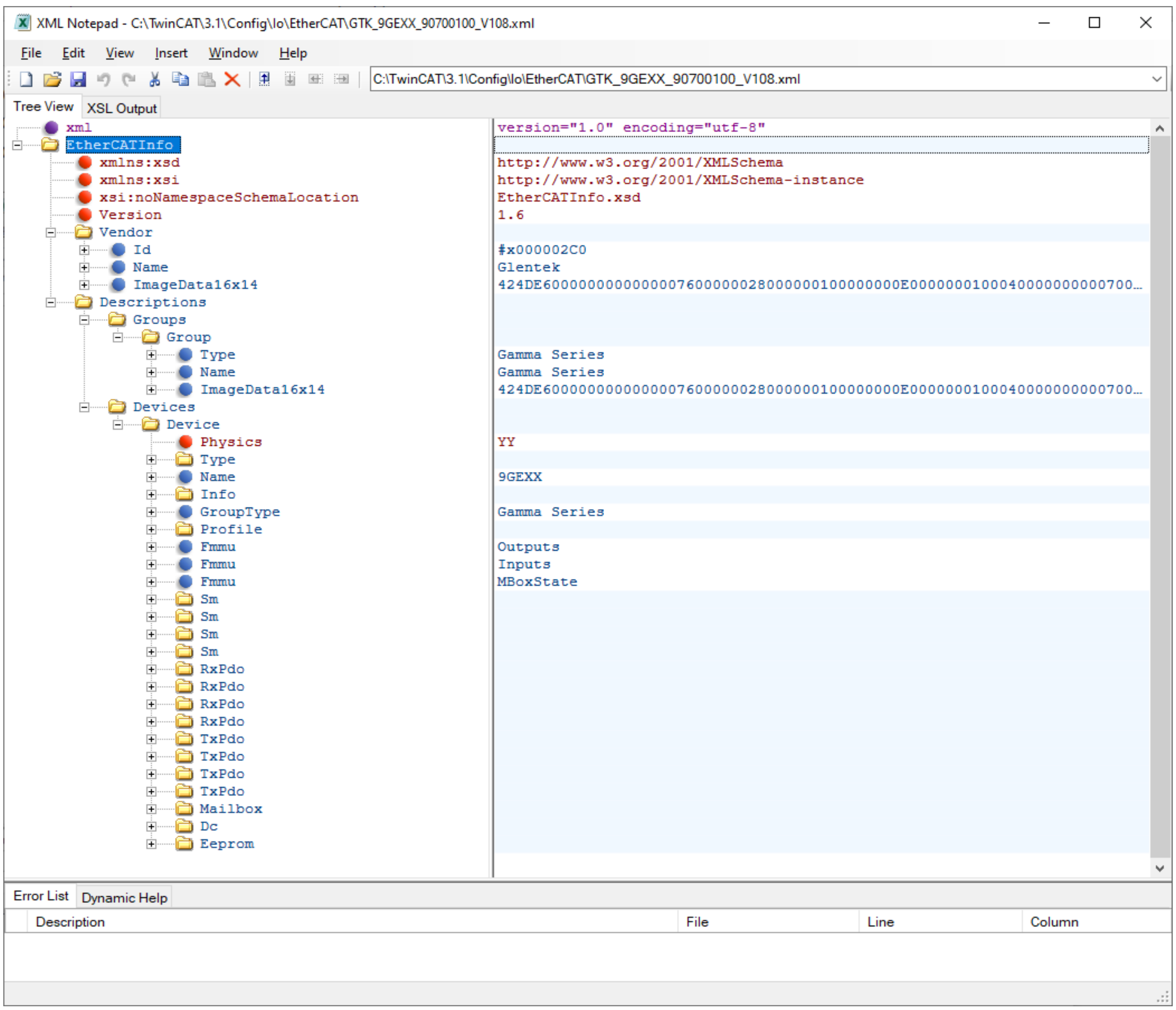


Figure 12 - 1. Open the ESI file

13. Communications

13.1. PDO Mapping

RxPDO Mappings are object 1600h to 1604h. TxPDO Mappings are object 1A00h to 1A04h. 1C12h and 1C13h define the PDO assignment. One of RxPDOs can be assigned at 1C12h and one of TxPDOs can be assigned at 1C13h. Only one PDO can be assigned at a time. The assigned PDOs will be activated for PDO data exchange.

The default PDO Mapping tables is shown in Table 13 - 1. Object 1600h and Object 1A00h are used for dynamic switchable CSP/CSV/CST mode. Object 1601h and Object 1A01h are used for CSP mode. Object 1602h and Object 1A02h are used for CSV mode. Object 1603h and Object 1A03h are used for CST mode. The default RxPDO and TxPDO are 0x1601 and 0x1A01 for CSP mode.

Object 1C12h, 1C13h, all RxPDOs and all TxPDOs can be dynamically mapped during transition from Pre-Operational to Safe-Operational by Master. All objects can use for PDO exchange after mapping. Object 1600h and 1A00h is mappable up to 16 objects and others is mappable up to 8 objects.

To map the content of these objects, the following procedure shall be done [4] .

1) Set Subindex 0 = 0 (the object is disabled)
   Ex1. 1C12:00h = 0, Ex2. 1601:00h = 0

2) Configure the mapping information in the mapping entries Subindex 1 … n
   Ex1. 1C12:01h = 1601h, Ex2. 1601:01h = 60400010h, 1601:02h = 607A0020h
   60400010h ➔ 6040h (object) | 00h (subindex) | 10h (bit length)
   607A0020h ➔ 607Ah (object) | 00h (subindex) | 20h (bit length)

3) Set Subindex 0 = number of used mapping entries (the object is enabled)
   Ex1. 1C12:00h = 1, Ex2. 1601:00h = 2

Table 13 - 1. PDO Mappings

<table>
<tr><th>PDO</th><th>RxPDO0</th><th>TxPDO0</th><th>RxPDO1</th><th>TxPDO1</th><th>RxPDO2</th><th>TxPDO2</th><th>RxPDO3</th><th>TxPDO3</th></tr>
<tr><td>Motion Mode</td><td colspan="2">CSP/CSV/CST</td><td colspan="2">CSP
(default)</td><td colspan="2">CSV</td><td colspan="2">CST</td></tr>
<tr><td>Object</td><td>1600h</td><td>1A00h</td><td>1601h</td><td>1A01h</td><td>1602h</td><td>1A02h</td><td>1603h</td><td>1A03h</td></tr>
<tr><td>SubIndex0</td><td>9</td><td>10</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>3</td></tr>
<tr><td>SubIndex1</td><td>Control Word (6040h) 16 bits</td><td>Status Word (6041h) 16 bits</td><td>Control Word (6040h) 16 bits</td><td>Status Word (6041h) 16 bits</td><td>Control Word (6040h) 16 bits</td><td>Status Word (6041h) 16 bits</td><td>Control Word (6040h) 16 bits</td><td>Status Word (6041h) 16 bits</td></tr>
<tr><td>SubIndex2</td><td>Target Position (607Ah) 32 bits</td><td>Position Actual Value (6064h) 32 bits</td><td>Target Position (607Ah) 32 bits</td><td>Position Actual Value (6064h) 32 bits</td><td>Target Velocity (60FFh) 32 bits</td><td>Position Actual Value (6064h) 32 bits</td><td>Target Torque (6071h) 16 bits</td><td>Position Actual Value (6064h) 32 bits</td></tr>
<tr><td>SubIndex3</td><td>Target Velocity (60FFh) 32 bits</td><td>Velocity Actual Value (606Ch) 32 bits</td><td rowspan="6">Available for Mapping</td><td rowspan="6">Available for Mapping</td><td rowspan="6">Available for Mapping</td><td rowspan="6">Available for Mapping</td><td rowspan="6">Available for Mapping</td><td>Torque Actual Value (6077h) 16 bits</td></tr>
<tr><td>SubIndex4</td><td>Target Torque (6071h) 16 bits</td><td>Torque Actual Value (6077h) 16 bits</td><td rowspan="5">Available for Mapping</td></tr>
<tr><td>SubIndex5</td><td>Mode of Operation (6060h) 8 bits</td><td>Modes of Operation Display (6061h) 8 bits</td></tr>
<tr><td>SubIndex6</td><td>Padding (8 bits)</td><td>Padding (8 bits)</td></tr>
<tr><td>SubIndex7</td><td>Max Torque (6072h) 16 bits</td><td>Following Error Actual Value (60F4h) 32 bits</td></tr>
<tr><td>SubIndex8</td><td>Touch Probe Function (60B8h) 16 bits</td><td>Touch Probe Status (60B9h) 16 bits</td></tr>
<tr><td>SubIndex9</td><td>Digital Outputs #1 (60FE:01h) 32 bits</td><td>Touch Probe Value (60BAh) 32 bits</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td></tr>
<tr><td>SubIndex10</td><td rowspan="2">Available for Mapping</td><td>Digital Inputs (60FDh) 32 bits</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td></tr>
<tr><td>SubIndex11
~
SubIndex16</td><td>Available for Mapping</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td></tr>
</table>

13.2. SDO Service [5]

CoE is supported for SDO data exchange. This is acyclic data exchange with Object Dictionary to configure objects for applications. SDO services are used to read from and write to the online OD of the slave. They are also used to read the OD structure. The master sends an SDO request, the slave sends either an SDO Response or an SDO Abort.

Datagram example (Figure 13 - 1): Mailbox communication, CoE SDO Service (via SM0, SM1 at Slave). In this example an SDO Upload Request is written to SM0 Mailbox Out and the response is read from SM1 Mailbox In: A upload response is returned in case of success to upload a data (Case I) and an abort is returned with abort code in case of fail (Case II).

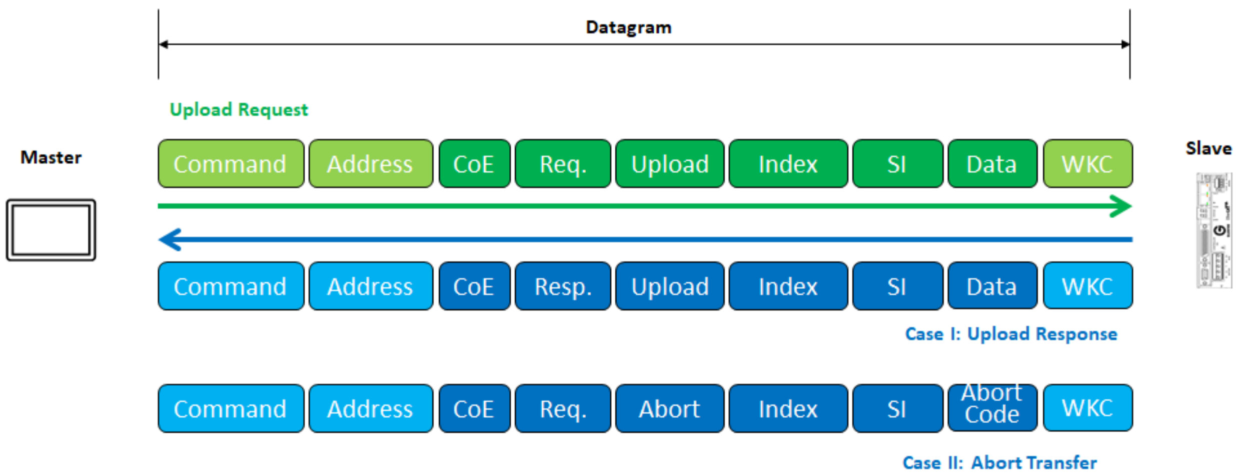


Figure 13 - 1. CoE SDO Service Datagram Example

14. Synchronization Modes

The Gamma drive supports two synchronization mode. One is Freerun mode for non-DC mode and the other is Sync0 synchronization mode for DC mode. The synchronization mode is configured by object 1C32:01h and 1C33:01h.

14.1. Non-DC Mode (Freerun)

In Free Run Mode, the local cycle is started by a local timer interrupt of the application controller. The cycle time may be changed by the Master (optional behavior of the slave) to change the timer. In Free Run Mode, the local cycle runs independent from the communication cycle and/or master cycle [4]. The Gamma application runs independently of the EtherCAT cycle and is triggered by a local timer in the ESC [3]. This mode is run when object 1C32:01h and 1C33:01h set to zero. The sequence of function that is run at the Gamma for Freerun mode is shown in Figure 14 - 1 [6]. The configurations of object 1C32h and 1C33h for Freerun mode are shown in Table 14 - 1 and Table 14 - 2.

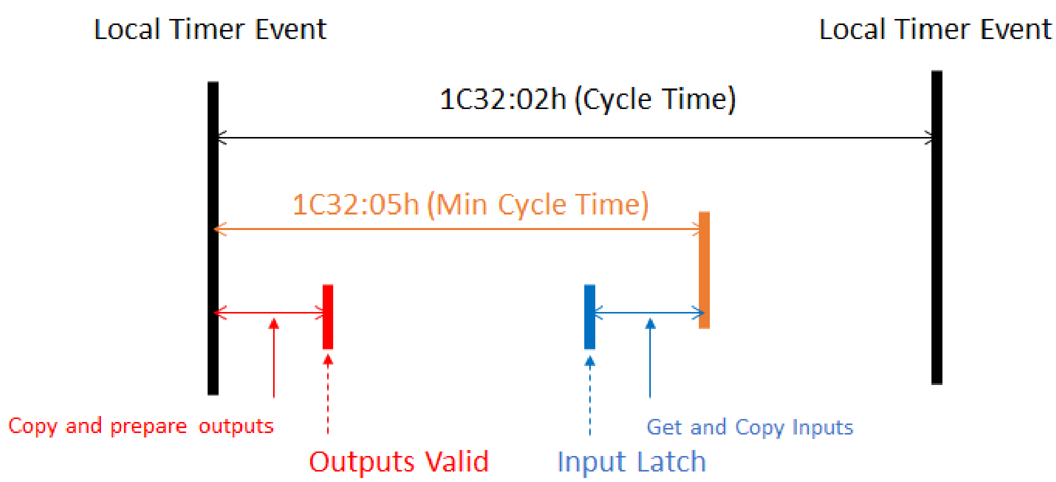


Figure 14 - 1. Freerun Mode

Table 14 - 1. Configurations of object 1C32h for Freerun Mode (See 5.19 Freerun)

Subindex	Description	Access	Use	Value
01h	Synchronization Type	R or RW	M	0
02h	Cycle Time	R or RW	O	Local Cycle Time
04h	Synchronization Types Supported	R	M	Bit 0: Free Run Supported
05h	Minimum Cycle time	R	C	150 [us]

Table 14 - 2. Configurations of object 1C33h for Freerun Mode (See 5.20 Freerun)

Subindex	Description	Access	Use	Value
01h	Synchronization Type	R or RW	M	0
02h	Cycle Time	R or RW	O	Same value as 0x1C32:02
04h	Synchronization Types Supported	R	M	Same value as 0x1C32:04
05h	Minimum Cycle time	R	C	Same value as 0x1C32:05

14.2. DC Mode (Sync0 Synchronization)

The ESCs contain a nanosecond-based timer (DC timer) to provide precise synchronization and time stamping. The Gamma application is triggered with an additional interrupt signal, which is based on the DC time and is produced by the ESC. Every DC timer in the network is aligned to a reference DC clock. The ESC system time is stored in a 64-bit value [3] . In this mode, the Gamma application is started on Sync0 event, so, this mode can reduce a jitter of running of Gamma application. This mode is run when object 1C32:01h and 1C33:01h set to 2. The sequence of function that is run at the Gamma for Sync0 Synchronization mode is shown in Figure 14 - 2 [6] . The configurations of object 1C32h and 1C33h for Sync0 Synchronization mode are shown in Table 14 - 3 and Table 14 - 4. The cycle time should be more than 250 [us]. The initial shift time (1C33:03h) for Input Latch is set to 120 [us] and it can be adjusted by cycle time.

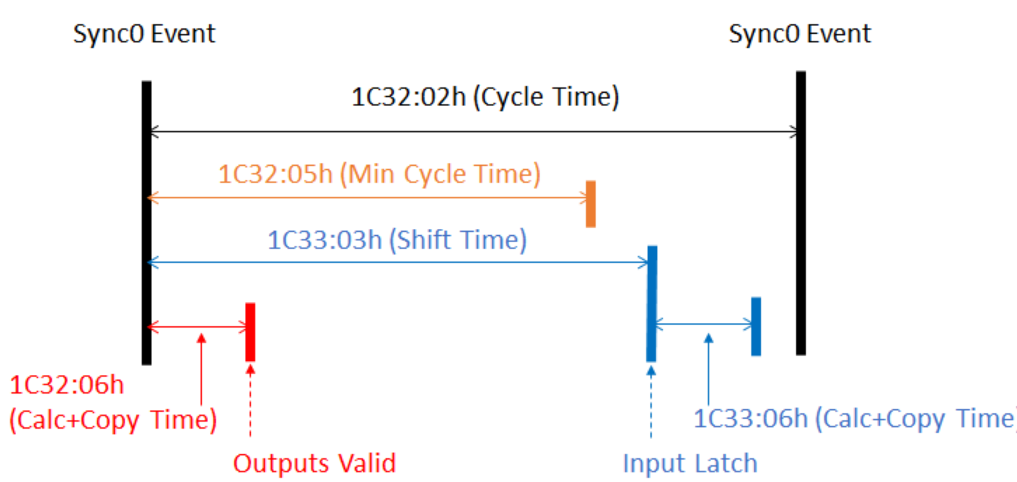


Figure 14 - 2. Sync0 Synchronization Mode

Table 14 - 3. Configurations of object 1C32h for Sync0 Synchronization Mode (See 5.19 Sync0)

Subindex	Description	Access	Use	Value
01h	Synchronization Type	RW	M	2
02h	Cycle Time	R	M	Same value as Sync0 Cycle Time (register 0x09A3:0x09A0)
03h	Shift Time	RW	M	-
04h	Synchronization Types Supported	R	M	Bit 4:2: DC Type supported, 001b = DC Sync0
05h	Minimum Cycle time	R	C	150 [us]
06h	Calc and Copy Time	R	M	30 [us]

Table 14 - 4. Configurations of object 1C33h for Sync0 Synchronization Mode (See 5.20 Sync0)

Subindex	Description	Access	Use	Value
01h	Synchronization Type	RW	M	2
02h	Cycle Time	R	M	Same value as 0x1C32:02
03h	Shift Time	RW	M	120 [us]
04h	Synchronization Types Supported	R	M	Same value as 0x1C32:04
05h	Minimum Cycle time	R	C	Same value as 0x1C32:05
06h	Calc and Copy Time	R	M	30 [us]

15. Connection Example with TwinCAT

15.1. Copying ESI file

The TwinCAT can detect the Gamma Drive as copying the ESI file on TwinCAT\3.1\Config\Io\EtherCAT folder as Figure 16 - 1. The ESI file can be changed for updating.

ESI file: GTK_9GEXX_90700100_Vxxx.xml

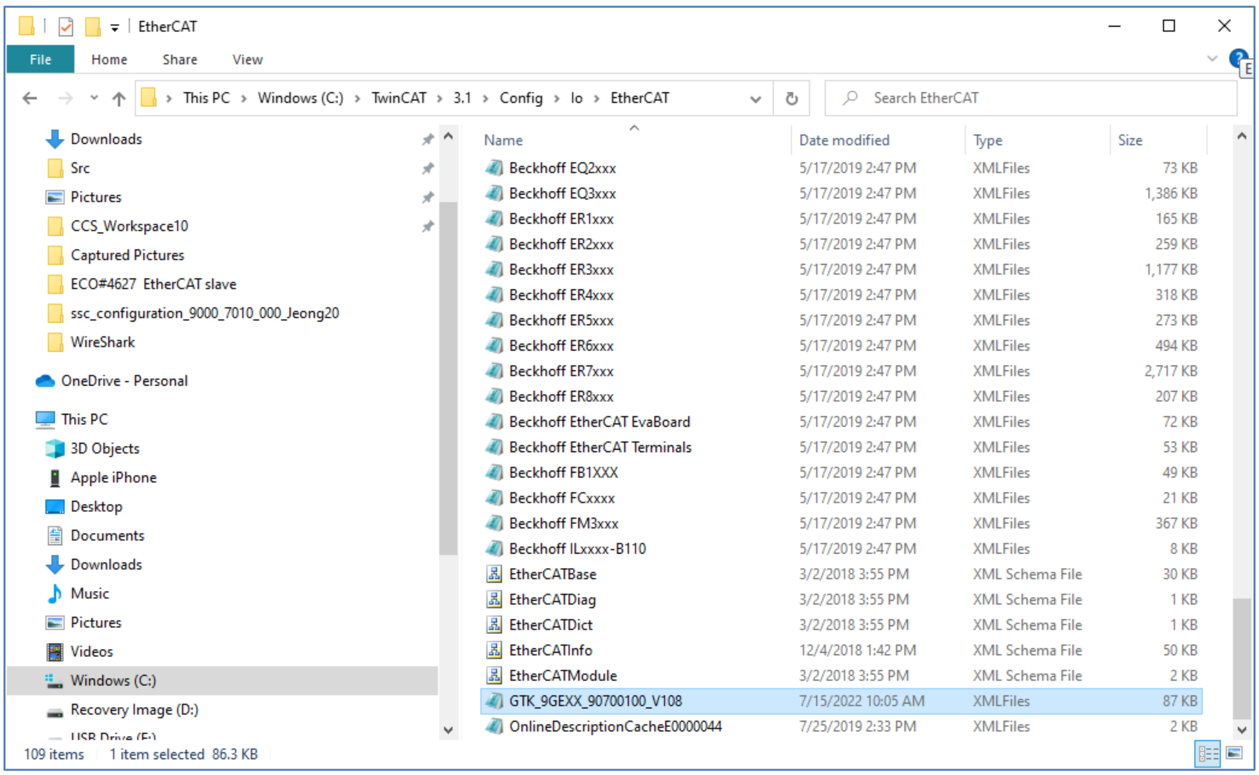


Figure 15 - 1. Copying ESI file for TwinCAT

15.2. Set-up Real-Time

To set-up the real-time of Task at TwinCAT, the Isolated Core is needed as Figure 15 - 2. The Core 3 is set-up for Isolated Core and the Base Time is set-up to 50 [us]. The setting-up of real-time is useful to reduce jitters of Task.

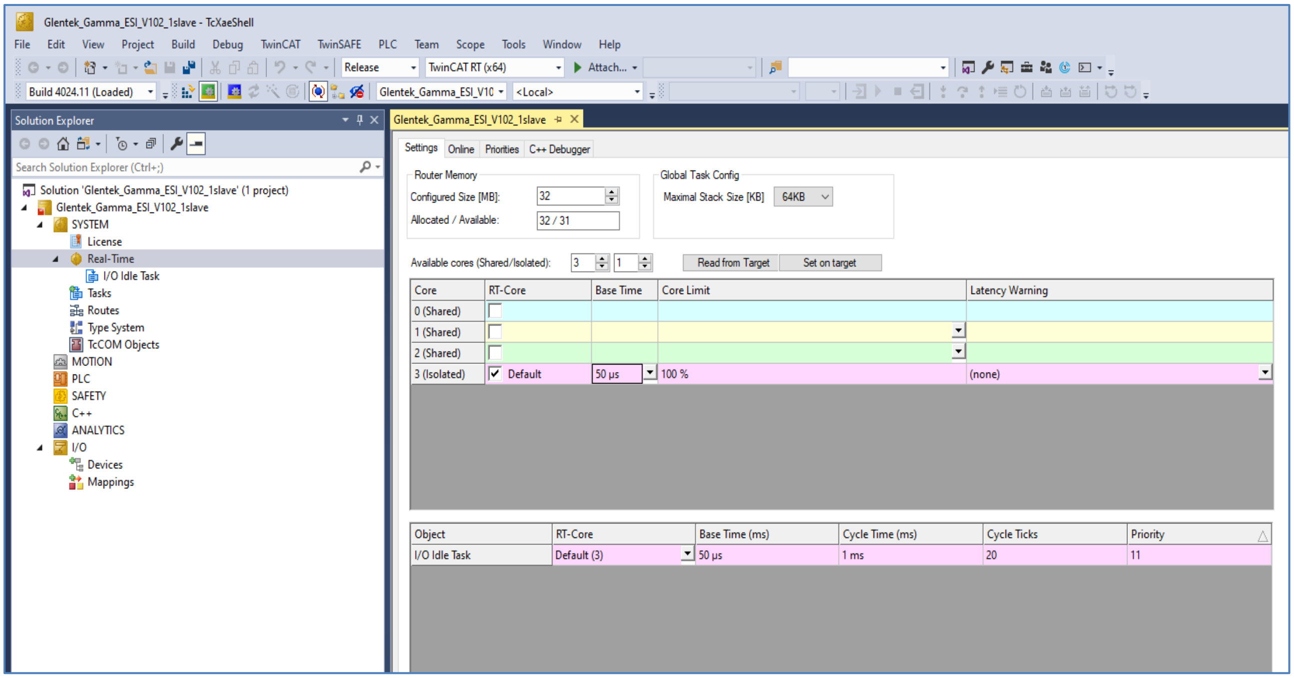


Figure 15 - 2. Set-up Real-Time with the Isolated Core

15.3. Set-up Task

The Cycle ticks is set-up to 0.25 [ms] as Figure 15 - 3. The cycle time in DC mode is the same as this configuration.

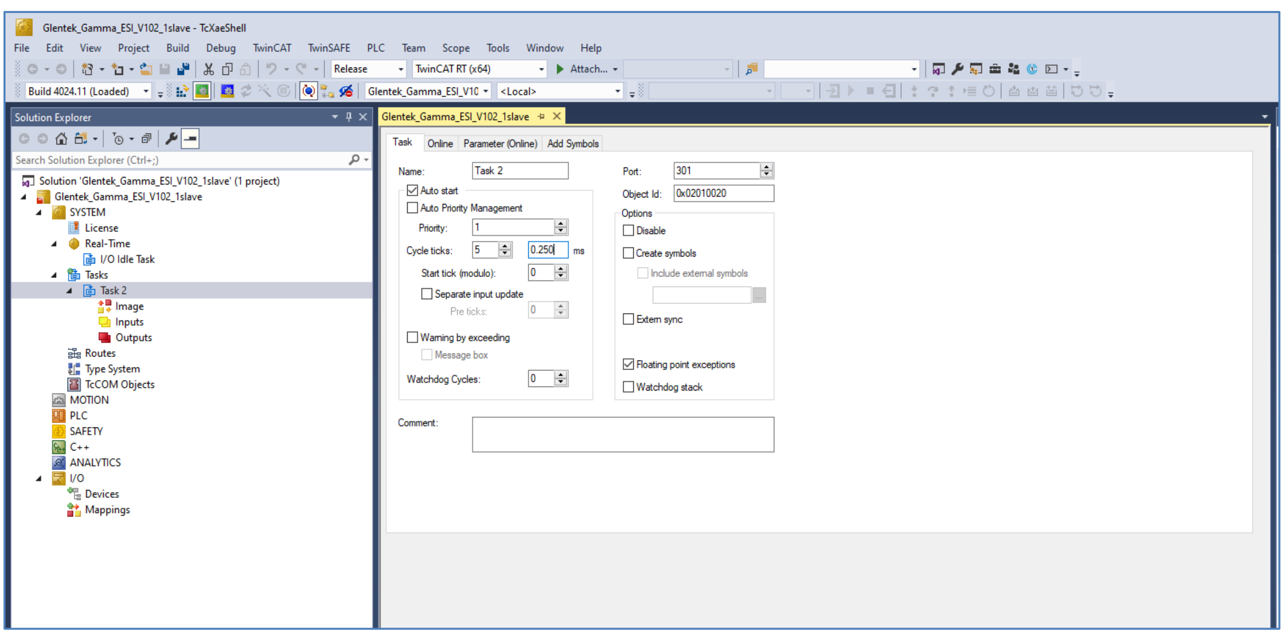


Figure 15 - 3. Set-up Task

15.4. Scanning Gamma Drive

The Gamma Drive (9GXX) is scanned after selecting "Scan" as Figure 15 - 4. The default PDO mapping is CSP mode. Object 1601h is mapped for RxPDO and object 1A01h is mapped for TxPDO.

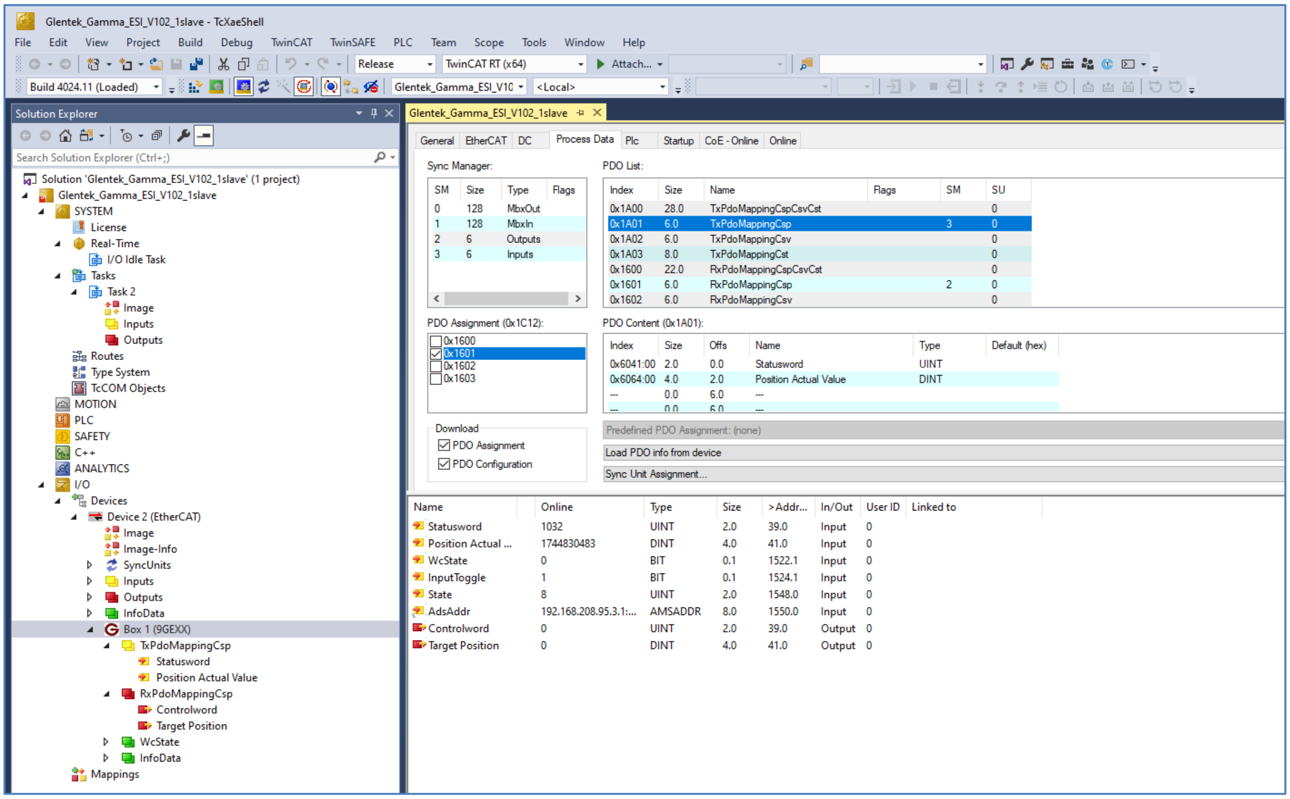


Figure 15 - 4. Scanning Gamma Drive

15.5. Run with Freerun Mode

The Freerun mode can be running by selecting “Freerun” on DC tab and then selecting “Activation” as Figure 15 - 5.

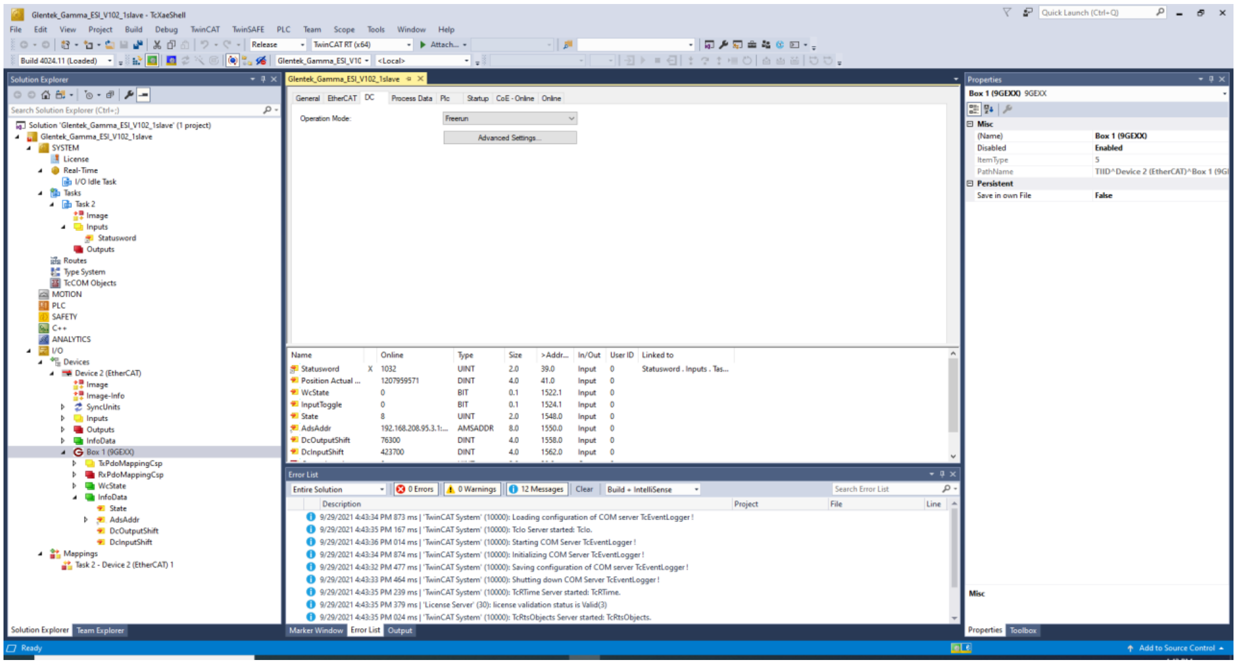


Figure 15 - 5. Run with Freerun Mode

15.6. Run with DC Mode

The DC mode can be running by selecting “DC-Synchron” on DC tab and then selecting “Activate Configuration” as Figure 15 - 6. This mode only supports Sync0 Synchronization.

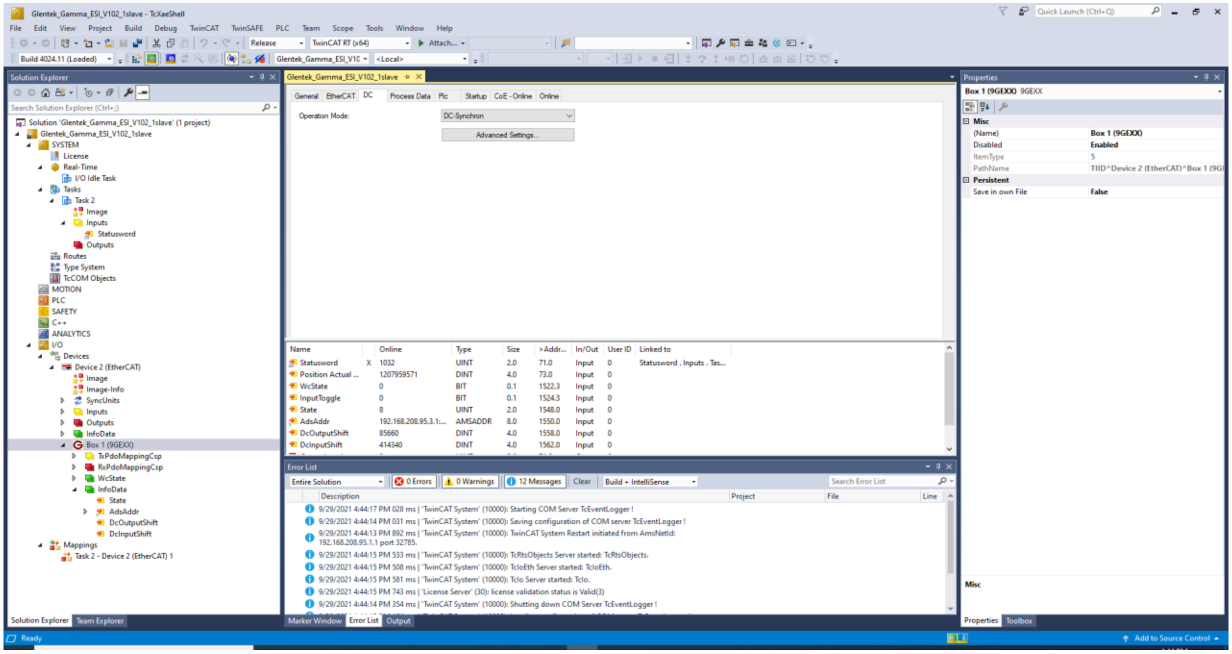


Figure 15 - 6. Run with DC Mode

15.7. PDO Mapping

The PDOs can be mapped during transition from Pre-Op state to Safe-Op state. PDO Contents can added or changed by Add New Item as Figure 15 - 7.

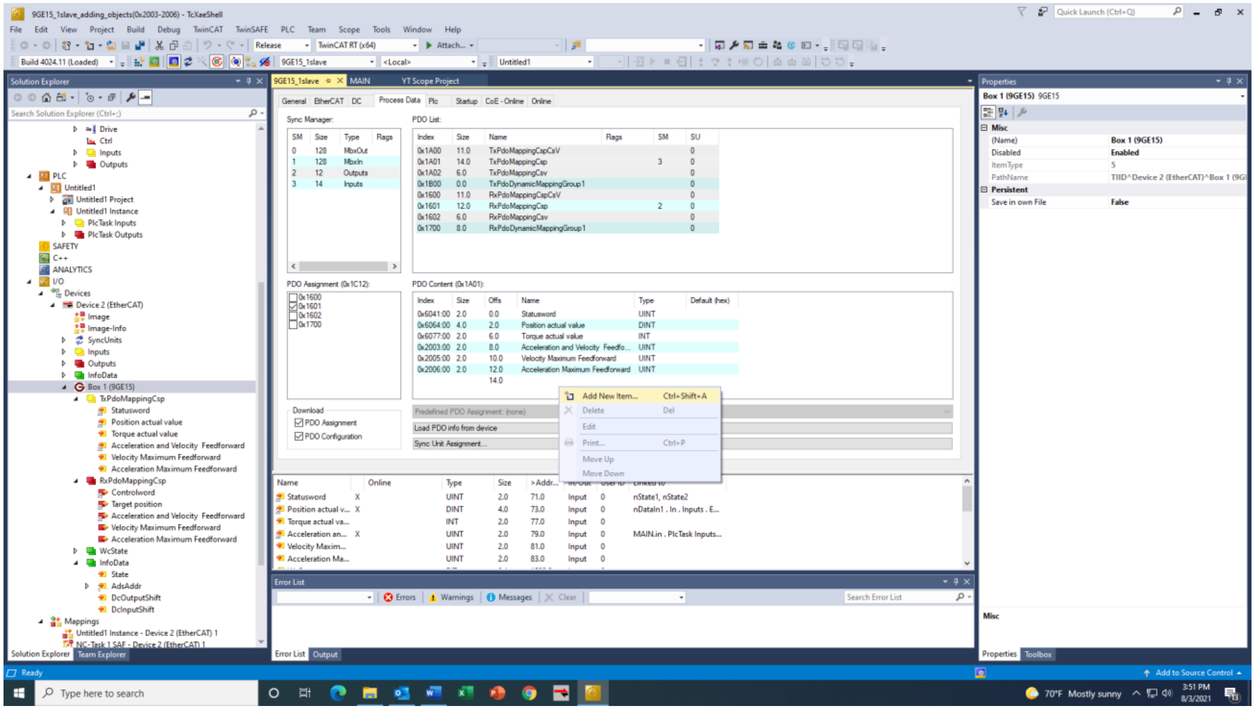


Figure 15 - 7. Add New Item for PDO Mapping

The Edit Pdo Entry window is opened, and available Objects are shown in Figure 15 - 8.

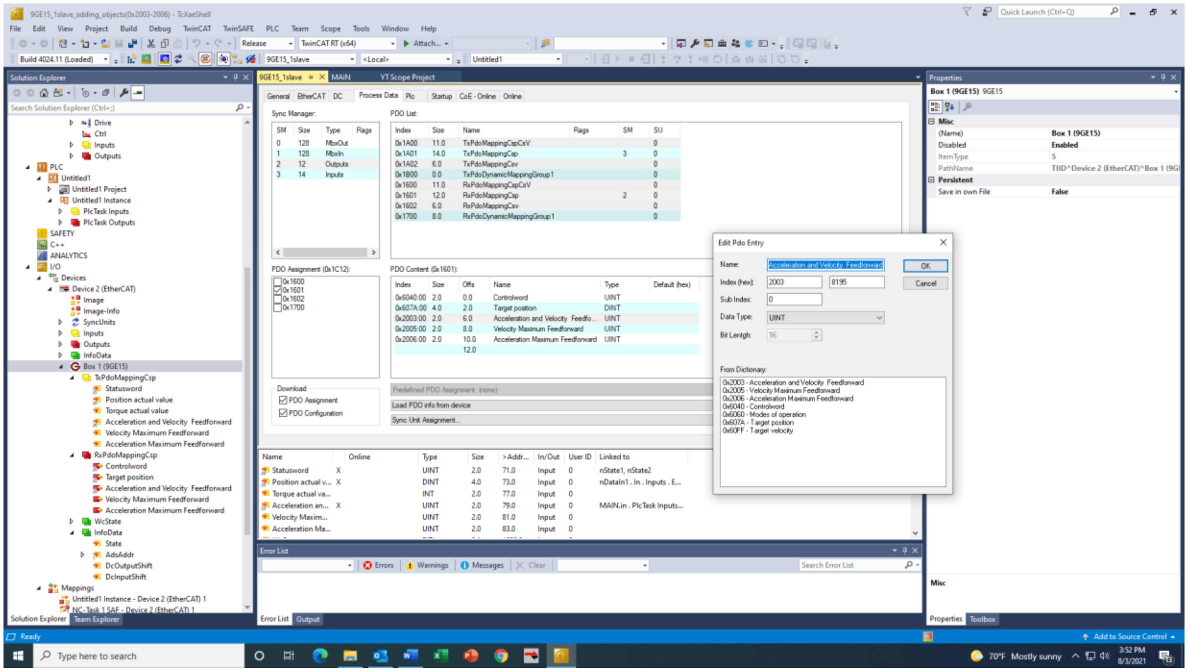


Figure 15 - 8. Edit Pdo Entry for PDO Mapping

The startup of PDO mapping is shown in Figure 15 - 9. The mapping can be running during transition from Pre-Op state to Safe-Op state.

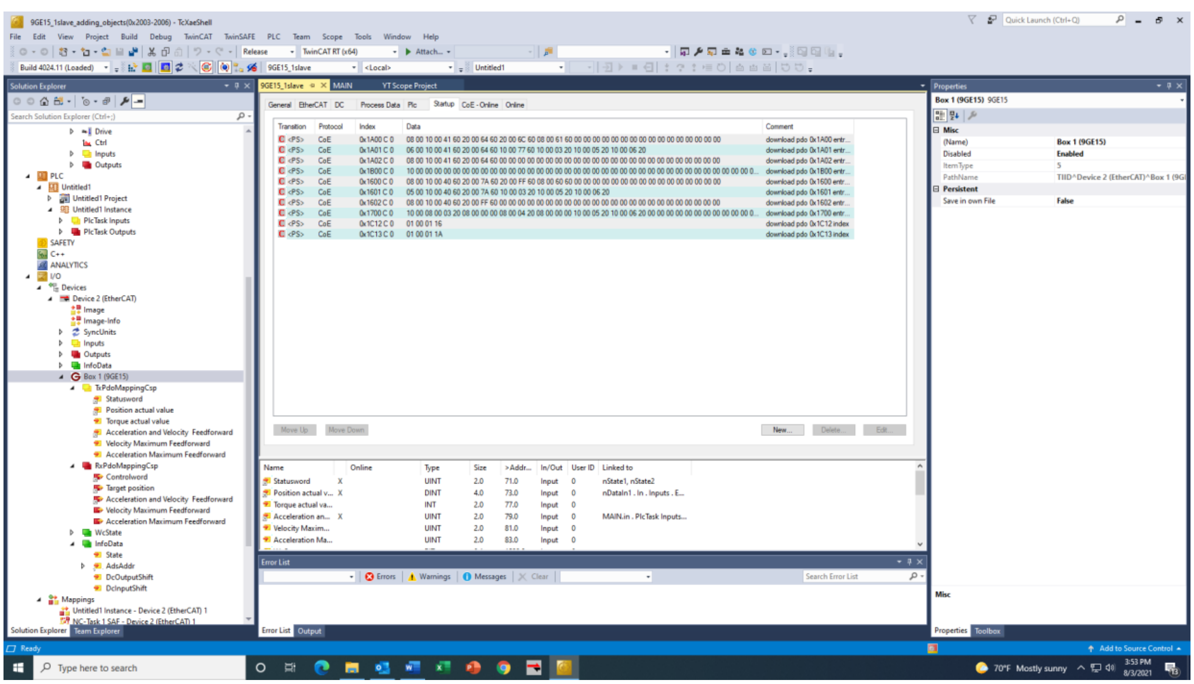


Figure 15 - 9. Startup of PDO Mapping

15.8. Test SDO Input/Output

The CoE-Online is used for testing SDO Input/Output. The values are shown as results of SDO Input. For example, object 6060h can be selecting and output to 8 for CSP mode as Figure 15 - 10.

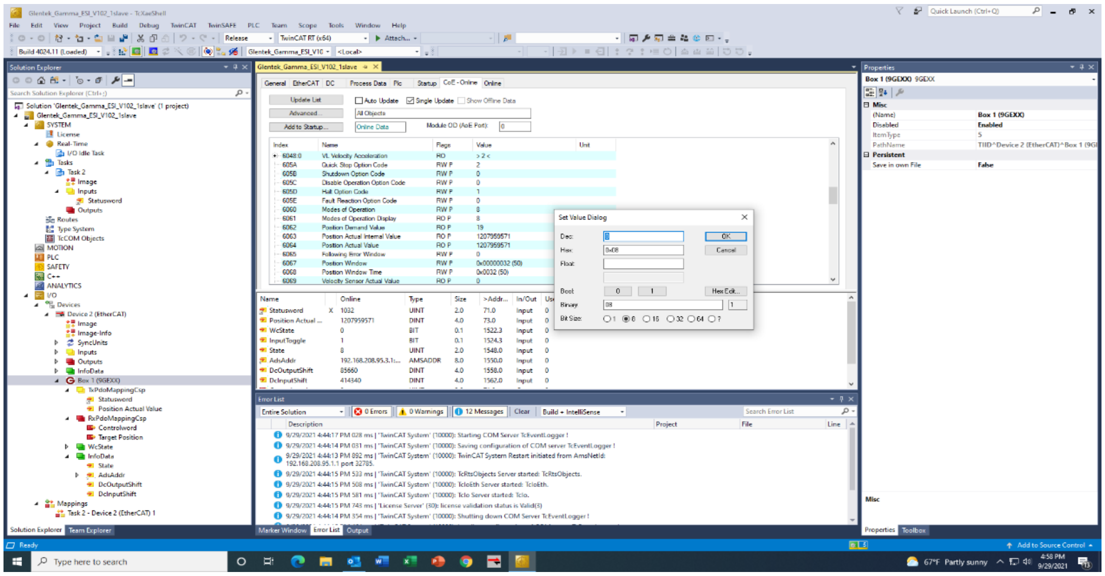


Figure 15 - 10. Testing SDO Output

15.9. Test PDO Input/Output

The objects in the Solution Explorer are used for testing PDO Input/Output. For example, the object Target Position (607Ah) at the RxPdoMappingCsp can be selecting and output values as Figure 15 - 11.

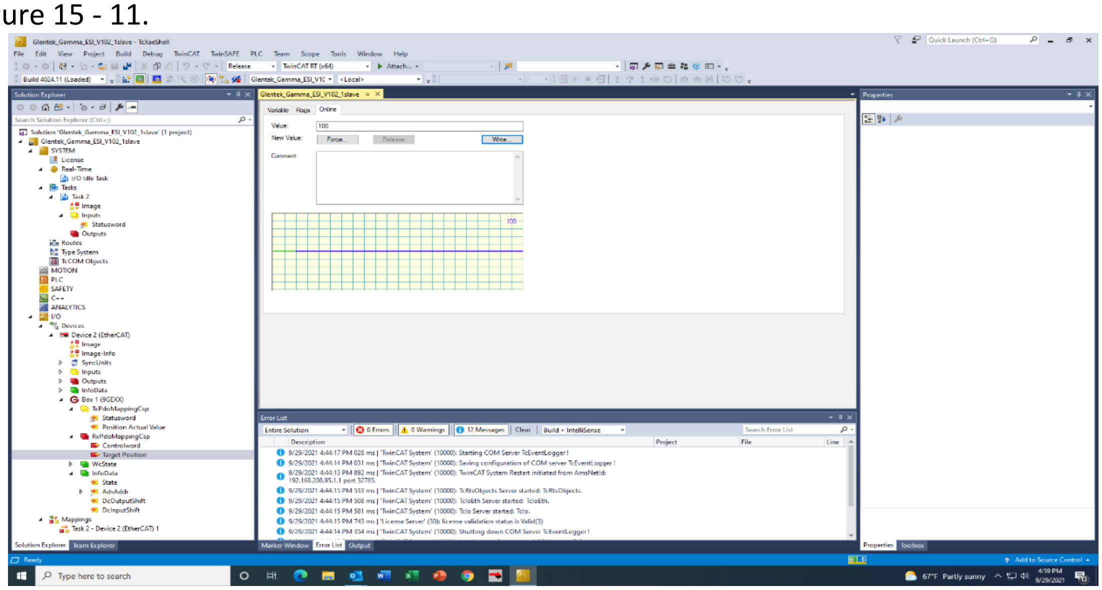


Figure 15 - 11. Testing PDO Output

16. Application Objects

16.1. Object 2000h: Motor Feedback Type

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
2000h	00h	Motor Feedback Type	-	No	Unsigned 16-bit	RO	No

This object reports the current motor feedback type.

Table 16 - 1. Bit Descriptions of 2000h

Bit	Description
0	TTL Encoder
1	Analog Sin/Cos Encoder
2	Resolver
3	Absolute Encoder
4	Tachometer
5	Hall Sensor
6:15	Reserved

16.2. Object 2001h: Faults of Drive

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
2001h	00h	Faults of Drive	-	Yes	Unsigned 16-bit	RO	No

This object reports the current faults of drive.

Table 16 - 2. Bit Descriptions of 2001h

Bit	Faults	Meaning
0	Bus Under volt	The Input Bus Voltage is below the minimum required
1	Commutation Init	Hall angles do not match encoder counter angle. No Halls: Phase finding routine failed
2	Internal Fault	An Internal Fault has occurred
3	Reserved	-
4	EEPROM	EEPROM checksum fault – Contact a Glentek application engineer
5	Hall	The drive has detected an invalid Hall state on the Hall inputs from the motor
6	Motor Over Temp	The drive has detected a motor over temp signal on the *Motor Overtemp* Input or on the I/O Connector
7	Reserved	-
8	RX Overflow	Software overflow
9	Position Following Error	Position Mode Only – Difference between commanded position and actual position is excessive
10	Over Speed	The Motor speed has exceeded the specified threshold
11	Encoder	The drive has detected a fault on one or more of the encoder inputs
12	Drive Over Temp	The drive has exceeded the drive maximum allowed temperature
13	Bus Over Volt	The Input Bus Voltage to drive has exceeded the allowed max voltage
14	LSECB (Low Speed Electronic Circuit Breaker)	The drive output current has exceeded the time + current constraints programmed into the drive
15	HSECB (High Speed Electronic Circuit Breaker)	The drive has detected a short circuit on one or more of the output terminals

16.3. Object 2002h: Status of Drive

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
2002h	00h	Status of Drive	-	Yes	Unsigned 16-bit	RO	No

This object reports the current status of drive.

Table 16 - 3. Bit Descriptions of 2002h

Bit	Status	Meaning
0	STO Active	STO (Safe Torque Off) input activated
1	Drive Fault	One or more of the internal drive faults has occurred
2	Commutation Init	The drive is finding the encoder commutation Tracks/Hall signals and calculating the rotor angle
3	Software Inhibit	The drive is disabled from within MotionMaestro software
4	At Zero	The motor is at zero rpm
5	At Speed	The motor is at a specified rpm
6	Drive Commutated	Commutation Initialization was successful
7	Encoder Index Sensed	The drive has detected the encoder index signal
8	Drive Enabled	The drive is not inhibited and no faults are present
9	Hardware Inhibit	An inhibit signal has been received at one of the I/O terminals
10	Current Fold back	The fold back function has occurred, motor current has been reduced to the configured level
11	Balancing	The drive is in the process of balancing the motor phase currents
12	Stopping	The drive is decelerating the motor
13	External Reset	The reset terminal of the I/O has been activated from an external source
14	Auto Phasing	The drive is in auto phasing
15	Reserved	-

16.4. Object 6062h: Position Demand Value

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
6062h	00h	Position Demand Value	-	Yes	Integer 32-bit	RO	No
The units are in Counts.							

This object shows the motor position which the drive is attempting to move the axis in units of encoder counts.

16.5. Object 6063h: Position Actual Internal Value

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
6063h	00h	Position Actual Internal Value	-	Yes	Integer 32-bit	RO	No
The units are in Counts.							

This object shows the actual motor position calculated by the drive-in units of encoder counts.

16.6. Object 6067h: At Position Window

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
6067h	00h	Position Window	32h	No	Unsigned 32-bit	RW	Yes
The units are in Counts.							

This object defines accept range for target position. When the actual position in the position window, the target position shall be regarded as having been reached. If the drive is situated in the Position Window range, the bit target reached (Bit 10) in Object 6041h: Statusword is set to 1.

16.7. Object 606Bh: Velocity Demand Value

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
606Bh	00h	Velocity Demand Value	-	Yes	Integer 32-bit	RO	No
The units are in counts/s.							

This object provides the output value of the trajectory generator. The value is given in counts/second.

16.8. Object 606Dh: Velocity Window

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
606Dh	00h	Velocity Window	819	No	Unsigned 16-bit	RW	Yes
The units are in percentage of commanded speed.							

This object defines the velocity tracking window. The value is given in percentage of commanded speed, which is used for determining if target reached status word Bit 10.

Value can be found be using the formula below:

$$Value = Percentage \times 32767$$

16.9. Object 606Fh: Velocity Threshold

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
606Fh	00h	Velocity Threshold	-	No	Unsigned 16-bit	RW	Yes
The units are in counts/s.							

This object defines the zero-velocity threshold. The value is given in counts/second. This is used for determining if "zero" speed is reached. When "zero" speed is reached, Statusword Bit 12 is set to 0. The default value is determined by setting 'Zero Speed Window' on MotionMaestro.

Value can be found be using the formula below:

Value = RPM x Line PPR x 4/60

Example for a rotary encoder 1024-line count, 20 rpm:

20 x 1024 x 4/60 = 1365

16.10. Object 607Fh: Max Profile Velocity

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
607Fh	00h	Max Profile Velocity	$2^{31}/2 - 1$	Yes	Unsigned 32-bit	RW	Yes
The units are in counts/s.							

16.11. Object 6083h: Profile Acceleration

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
6083h	00h	Profile Acceleration	0h	No	Unsigned 32-bit	RW	Yes
The units are in counts/s^2.							

16.12. Object 6084h: Profile Deceleration

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
6084h	00h	Profile Deceleration	0h	No	Unsigned 32-bit	RW	Yes
The units are in counts/s^2.							

16.13. Object 60FCh: Position Demand Internal Value

Object	Sub-Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60FCh	00h	Position Demand Internal Value	-	Yes	Integer 32-bit	RO	No
The units are in Counts.							

This object reports the position calculated by the motion profile in units of encoder counts; it takes into the account the acceleration and velocity targets.

16.14. Object 60FDh: Digital Inputs

Object	Sub Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60FDh	00h	Digital Inputs	-	Yes	Unsigned 32-bit	RO	No

The object 60FDh: Digital Inputs can be used after assigning of the programmable digital input by MotionMaestro.

Table 16 - 4. Definition of 60FDh

Bit	Definition
0	Reserved
1	External Inhibit
2	External Reset
3	Positive Limit
4	Negative Limit
5	Fault Input
6	Stop/Go Control
7	E-Stop
8	Aux Encoder Fault
9	Drive Operation Control
10	Profile Generator Trigger Bit
11	Profile Generator Stop Bit
12	Set Home Bit
13	Safe Stop1
14-19	Reserved
20	Touch Probe 1 Input
21	Touch Probe 2 Input
22-31	Reserved

16.15. Object 60FEh: Digital Outputs

Object	Sub Index	Description	Default	PDO Map	Data Type	Access	Save to EEPROM
60FEh	00h	Digital Outputs	2h	No	Unsigned 16-bit	RO	No
60FEh	01h	Digital Outputs 1	0h	Yes	Unsigned 32-bit	RW	No
60FEh	02h	Reserved	0h	Yes	Unsigned 32-bit	RW	No

The object 60FE:01h Digital Outputs 1 can be used after assigning of the programmable digital output by MotionMaestro.

Table 16 - 5. Definition of 60FE:01h Digital Outputs 1

Bit	Definition
0	Digital Out: Bit1
1	Digital Out: Bit2
2	Digital Out: Bit3
3	Digital Out: Bit4
4	Digital Out: Bit5
5	Digital Out: Bit6
6	Digital Out: Bit7
7	Digital Out: Bit8
8-31	Reserved

For example, Open **Setup** > **Setup Digital Outputs** at MotionMaestro and then assign Digital Out:Bit 1 as Figure 16 - 1.

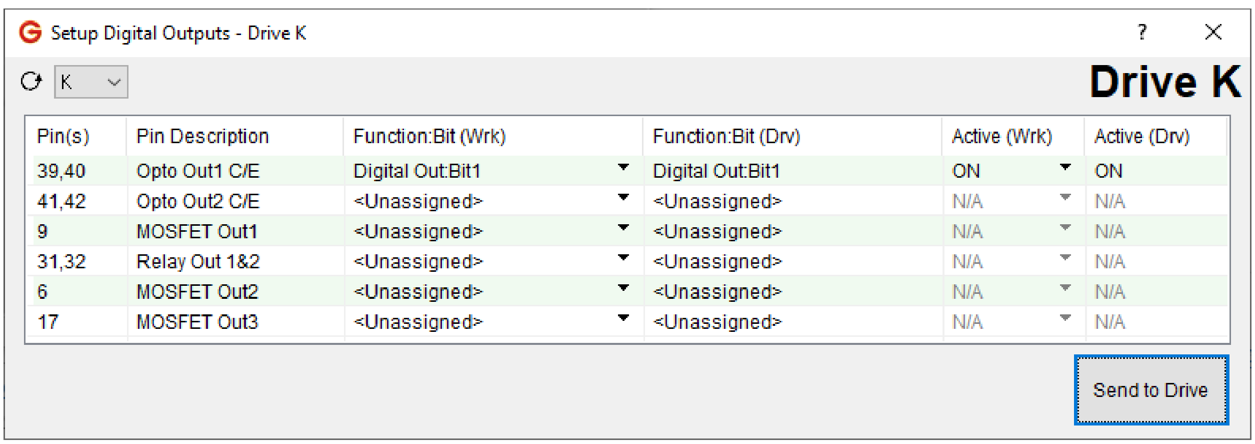


Figure 16 - 1. Example of Assigning Digital Outputs

Appendix A Object Reference

A.1. Communication Objects

A.2. Drive and Motion Control Objects

Appendix B References

[1] https://www.ethercat.org

[2] EtherCAT Overview & ETG Introduction (Brochure)

[3] ETG.2200 V3.1.0 Slave Implementation Guide

[4] ETG. 1020 Protocol Enhancements

[5] EtherCAT Device Protocol Poster

[6] Synchronization modes of an EtherCAT slave (https://infosys.beckhoff.com)

[7] ETG.1020 EtherCAT Protocol Enhancements

[8] International Standard Adjustable Speed Electrical Power Systems – Part 7-201: Generic Interface and Use of Profiles for Power Drive Systems – Profile Type 1 Specification (IEC 61800-7-201 Edition 2.0 2015-11)

Glentek's Gamma EtherCAT Manual

1st Edition

MADE IN

AMERICA

www.glentek.com
208 Standard St.
El Segundo, CA 90245
T: (310) 322-3026
F: (310) 322-7709